



Guia do Desenvolvedor

Amazon CloudFront



Amazon CloudFront: Guia do Desenvolvedor

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens de marcas da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não pertencem à Amazon pertencem a seus respectivos proprietários, que podem ou não ser afiliados, conectados ou patrocinados pela Amazon.

Table of Contents

O que é o Amazon CloudFront?	1
Como configurar o CloudFront para entregar conteúdo	2
Definição de preço	4
Maneiras de usar o CloudFront	4
Acelerar a entrega de conteúdo de site estático	5
Fornecer vídeos de streaming ao vivo ou sob demanda	5
Criptografar campos específicos durante todo o processamento do sistema	5
Personalizar na borda	6
Fornecer conteúdo privado usando personalizações do Lambda@Edge	6
Como o CloudFront entrega conteúdo	7
Como o CloudFront entrega conteúdo aos usuários	7
Como o CloudFront funciona com caches de borda regionais	8
Servidores de borda do CloudFront	11
Use a lista de prefixos gerenciados do CloudFront	11
Como trabalhar com AWS SDKs	12
Recursos técnicos do CloudFront	13
Conceitos básicos	14
Configurar a Conta da AWS	14
Cadastrar-se em uma Conta da AWS	14
Criar um usuário com acesso administrativo	15
Escolha como acessar o CloudFront	16
Conceitos básicos de uma distribuição básica	17
Pré-requisitos	18
Criar bucket	18
Fazer upload de conteúdo	19
Criar distribuição	19
Acessar o conteúdo	20
Limpeza	21
Aprimorar a distribuição básica	21
Conceitos básicos de um site estático seguro	22
Visão geral da solução	22
Implante a solução	23
Configurar distribuições	29
Criar uma distribuição	30

Criar uma distribuição do CloudFront no console	32
Valores que são exibidos	33
Links adicionais	34
Configurações de distribuição	35
Configurações de origem	35
Configurações de comportamento de cache	46
Configurações de distribuição	61
Páginas de erro personalizadas e erro de armazenamento em cache	72
Restrições geográficas	74
Testar uma distribuição	74
Criar links para seus objetos	74
Atualizar uma distribuição	75
Marcar uma distribuição	77
Restrições de tags	78
Adicionar, editar e excluir tags para distribuições	78
Marcação programática	79
Excluir uma distribuição do	79
Usar implantação contínua para testar com segurança as alterações	81
Fluxo de trabalho de implantação contínua do CloudFront	83
Trabalhar com uma distribuição de preparação e uma política de implantação contínua	84
Monitorar uma distribuição de preparação	95
Saiba como funciona a implantação contínua	95
Cotas e outras considerações para implantação contínua	97
Usar várias origens	99
Usar um bucket do Amazon S3	99
Usar um contêiner do MediaStore ou um canal do MediaPackage	111
Usar um Application Load Balancer	112
Usar um URL da função do Lambda	112
Usar o Amazon EC2 (ou outra origem personalizada)	113
Usar grupos de origem do CloudFront	115
Usar URLs personalizados	115
Requisitos para o uso de nomes de domínio alternativos	116
Restrições de uso de nomes de domínio alternativos	118
Adicionar um nome de domínio alternativo	119
Mover um nome de domínio alternativo para uma distribuição diferente	123
Remover um nome de domínio alternativo	130

Usar curingas em nomes de domínio alternativos	131
Usar WebSockets	132
Como o protocolo WebSocket funciona	132
Requisitos de WebSocket	133
Cabeçalhos WebSocket recomendados	133
Armazenamento em cache e disponibilidade	134
Melhorar a proporção de acertos de cache	135
Especificar o tempo durante o qual o CloudFront armazena os objetos em cache	135
Usar o Origin Shield	135
Armazenar em cache com base em parâmetros de string de consulta	136
Armazenar em cache com base nos valores dos cookies	136
Armazenar em cache com base nos cabeçalhos de solicitação	137
Remova o cabeçalho Accept-Encoding quando a compactação não for necessária	139
Disponibilize conteúdo de mídia via HTTP	139
Usar o escudo de origem	139
Casos de uso do Origin Shield	140
Escolher a região da AWS para o Origin Shield	146
Habilitar o Origin Shield	148
Estimar custos do Origin Shield	150
Alta disponibilidade do Origin Shield	151
Como o Origin Shield interage com outros recursos do CloudFront	152
Aumentar disponibilidade com o failover de origem	153
Criar um grupo de origem	155
Controlar tempos limite e tentativas da origem	156
Uso do failover de origem com funções do Lambda@Edge	157
Usar páginas de erro personalizadas com failover de origem	158
Gerenciar expiração de cache	159
Usar cabeçalhos para controlar a duração do cache para objetos individuais	160
Fornecer conteúdo obsoleto (expirado)	161
Especificar por quanto tempo o CloudFront armazena os objetos em cache	163
Adicionar cabeçalhos aos objetos usando o console do Amazon S3	170
Armazenamento em cache e parâmetros de string de consulta	170
Configurações do console e da API para encaminhamento e armazenamento de strings de consulta em cache	173
Otimizar o armazenamento em cache	173
Parâmetros de string de consulta e logs padrão do CloudFront (logs de acesso)	175

Conteúdo em cache com base em cookies	175
Conteúdo em cache com base nos cabeçalhos de solicitação	179
Visão geral de cabeçalhos e distribuições	179
Selecionar os cabeçalhos para basear o armazenamento em cache	181
Configurar o CloudFront para respeitar as configurações do CORS	182
Configurar o armazenamento em cache com base no tipo de dispositivo	183
Configurar o armazenamento em cache com base no idioma do visualizador	183
Configurar o armazenamento em cache com base no local do visualizador	183
Configurar o armazenamento em cache com base no protocolo da solicitação	184
Configurar o armazenamento em cache para arquivos compactados	184
Como o armazenamento em cache com base em cabeçalhos afeta a performance	184
Como a letra e os valores do cabeçalho afetam o armazenamento em cache	184
Cabeçalhos que o CloudFront retorna ao visualizador	185
Controlar a chave de cache com uma política	186
Noções básicas sobre políticas de cache	187
Informações de política	187
Configurações de vida útil (TTL)	187
Configurações da chave de cache	188
Criar políticas de cache	195
Usar políticas de cache gerenciadas	199
Amplify	200
CachingDisabled	201
CachingOptimized	201
CachingOptimizedForUncompressedObjects	202
Elemental-MediaPackage	203
UseOriginCacheControlHeaders	204
UseOriginCacheControlHeaders-QueryStrings	205
Noções básicas sobre a chave de cache	206
Chave de cache padrão	206
Personalizar a chave de cache	208
Controlar as solicitações de origem com uma política	210
Noções básicas sobre políticas de solicitação de origem	211
Informações de política	211
Configurações da solicitação de origem	211
Criar políticas de solicitação de origem	214
Usar políticas de solicitação de origem gerenciadas	219

AllViewer	219
AllViewerAndCloudFrontHeaders-2022-06	220
AllViewerExceptHostHeader	221
CORS-CustomOrigin	222
CORS-S3Origin	222
Elemental-MediaTailor-PersonalizedManifests	223
UserAgentRefererHeaders	223
Adicionar cabeçalhos de solicitação do CloudFront	224
Cabeçalhos para determinar o tipo de dispositivo do visualizador	225
Cabeçalhos para determinar a localização do visualizador	225
Cabeçalhos para determinar a estrutura de cabeçalho do visualizador	227
Outros cabeçalhos do CloudFront	227
Noções básicas sobre como as políticas de solicitação de origem e as políticas de cache funcionam juntas	229
Adicionar ou remover cabeçalhos de resposta com uma política	234
Noções básicas das políticas de cabeçalhos de resposta	235
Detalhes da política (metadados)	235
Cabeçalhos de CORS	236
Cabeçalhos de segurança	240
Cabeçalhos personalizados	242
Remover cabeçalhos	243
Cabeçalho de temporização do servidor	245
Criar políticas de cabeçalhos de resposta	249
Usar políticas de cabeçalhos de resposta gerenciadas	257
CORS-and-SecurityHeadersPolicy	257
CORS-With-Preflight	258
CORS-with-preflight-and-SecurityHeadersPolicy	259
SecurityHeadersPolicy	260
SimpleCORS	261
Comportamento de solicitações e respostas	263
Como o CloudFront processa solicitações HTTP e HTTPS	263
Comportamento de solicitações e respostas para origens do Amazon S3	264
Como o CloudFront processa e encaminha solicitações à sua origem do Amazon S3	264
Como o CloudFront processa as respostas da origem do Amazon S3	271
Comportamento de solicitações e respostas para origens personalizadas	274
Como o CloudFront processa e encaminha solicitações para sua origem personalizada	274

Como o CloudFront processa respostas da sua origem personalizada	293
Comportamento de solicitações e respostas para grupos de origens	298
Adicionar cabeçalhos personalizados às solicitações de origem	299
Casos de uso	299
Configurar o CloudFront para adicionar cabeçalhos personalizados às solicitações de origem	300
Cabeçalhos personalizados que o CloudFront não pode adicionar às solicitações da origem	301
Configurar o CloudFront para encaminhar o cabeçalho Authorization	302
Como o CloudFront processa range GETs	303
Usar solicitações de intervalo para armazenar objetos grandes em cache	304
Como o CloudFront processa códigos de status HTTP 3xx da origem	305
Como o CloudFront processa códigos de status HTTP 4xx e 5xx da origem	305
Como o CloudFront processará erros quando páginas de erro personalizadas estiverem configuradas	307
Como o CloudFront processará erros quando páginas de erro personalizadas não estiverem configuradas	309
Códigos de status HTTP 4xx e 5xx armazenados em cache pelo CloudFront	310
Gerar respostas a erros personalizadas	312
Configurar o comportamento de resposta a erros	313
Criar uma página de erro personalizada para códigos de status HTTP específicos	314
Armazenar objetos e páginas de erro personalizadas em diferentes locais	317
Alterar códigos de resposta exibidos pelo CloudFront	317
Controlar por quanto tempo o CloudFront armazena erros em cache	318
Adicionar, remover ou substituir conteúdo	320
Adicionar e acessar conteúdo	320
Usar o versionamento de arquivos para atualizar ou remover conteúdo existente	321
Atualizar arquivos existentes usando nomes de arquivos com versão	321
Remover conteúdo para que o CloudFront não o distribua	322
Personalizar URLs de arquivos	322
Usar seu próprio nome de domínio (example.com)	323
Usar uma barra (/) no final de URLs	323
Criar URLs assinados para conteúdo restrito	324
Especificar um objeto raiz padrão	324
Como especificar um objeto raiz padrão	324
Como funciona o objeto raiz padrão	326

Como funciona o CloudFront se você não define um objeto raiz	327
Invalidar arquivos para remover conteúdo	328
Escolher entre invalidar arquivos e usar nomes de arquivos com versionamento	328
Determinar quais arquivos invalidar	329
O que você precisa saber ao invalidar arquivos	329
Invalidar arquivos	333
Máximo de solicitações de invalidação simultâneas	337
Pagar pela invalidação de arquivos	337
Fornecer arquivos compactados	338
Configurar o CloudFront para compactar objetos	339
Como a compactação do CloudFront funciona	340
Quando o CloudFront compacta objetos	341
Tipos de arquivos compactados pelo CloudFront	343
Conversão do cabeçalho ETag	344
Usar proteções do AWS WAF	346
Habilitar o AWS WAF para distribuições	347
Habilitar o AWS WAF para uma nova distribuição	347
Usar uma ACL da web existente	348
Habilitar o controle de bots	349
Configurar a proteção por categoria de bot	349
Gerenciar as proteções de segurança do AWS WAF para o CloudFront	350
Pré-requisitos	351
Habilitar logs do AWS WAF	352
Configurar a limitação de taxa	353
Desabilitar as proteções de segurança do AWS WAF	353
Configurar o acesso seguro e restringir o acesso ao conteúdo	355
Usar HTTPS com o CloudFront	355
Exigir HTTPS entre os visualizadores e o CloudFront	357
Exigir HTTPS para uma origem personalizada	359
Exigir HTTPS para uma origem do Amazon S3	362
Protocolos e cifras compatíveis entre visualizadores e o CloudFront	364
Protocolos e criptografias compatíveis entre o CloudFront e a origem	370
Usar nomes de domínio alternativos e HTTPS	372
Escolher como o CloudFront atende a solicitações HTTPS	373
Requisitos para usar certificados SSL/TLS com o CloudFront	377

Cotas no uso de certificados SSL/TLS com o CloudFront (somente HTTPS entre visualizadores e o CloudFront)	382
Configurar nomes de domínio alternativos e HTTPS	384
Determinar o tamanho da chave pública em um certificado RSA SSL/TLS	388
Aumentar as cotas de certificados SSL/TLS	388
Trocar certificados SSL/TLS	389
Reverter um certificado SSL/TLS personalizado para o certificado padrão do CloudFront	391
Alternar de um certificado SSL/TLS personalizado com endereços IP dedicados para SNI ..	392
Restringir conteúdo com URLs e cookies assinados	393
Como fornecer conteúdo privado	393
Restringir o acesso a arquivos	394
Especificar signatários confiáveis	397
Decidir usar URLs ou cookies assinados	407
Usar URLs assinados	408
Usar cookies assinados	430
Comandos do Linux e OpenSSL para criptografia e codificação base64	454
Exemplos de código para signed URLs	454
Restringir o acesso a uma origem da AWS	483
Restringir o acesso a uma origem da AWS Elemental MediaPackage v2	484
Restringir o acesso a uma origem da AWS Elemental MediaStore	491
Restringir o acesso a uma origem do URL de função do AWS Lambda	499
Restringir o acesso a uma origem do Amazon Simple Storage Service	506
Restringir o acesso aos Application Load Balancers	522
Configurar o CloudFront para adicionar um cabeçalho HTTP personalizado às solicitações	523
Configurar um Application Load Balancer para encaminhar apenas solicitações que contenham um cabeçalho específico	525
(Opcional) Melhorar a segurança dessa solução	526
(Opcional) Limite o acesso à origem usando a lista de prefixos gerenciada pela AWS para o CloudFront	528
Restrição geográfica	528
Usar restrições geográficas do CloudFront	528
Usar um serviço de geolocalização de terceiros	530
Use criptografia de nível de campo para ajudar a proteger dados confidenciais	532
Visão geral da criptografia no nível de campo	534
Configurar a criptografia em nível de campo	535

Descriptografar campos de dados na origem	541
Vídeo sob demanda e streaming de vídeo ao vivo	545
Sobre o streaming de vídeo	545
Fornecer vídeo sob demanda	546
Configurar vídeo sob demanda para o Microsoft Smooth Streaming	547
Fornecer streaming de vídeo ao vivo	549
Distribuir vídeo usando o AWS Elemental MediaStore como origem	550
Veicular vídeo ao vivo formatado com o AWS Elemental MediaPackage	551
Usar funções para personalizar na borda	559
Diferenças entre o CloudFront Functions e o Lambda@Edge	560
Personalizar com o CloudFront Functions	563
Tutorial: Criar uma função simples do CloudFront	564
Tutorial: Criar uma função do CloudFront que use valores de chave	567
Escrever código de função	570
Criar funções	652
Testar funções	655
Atualizar funções	660
Publicar as funções	662
Associar funções a distribuições	664
KeyValueStore do CloudFront	667
Personalizar com o Lambda@Edge	690
Como o Lambda@Edge funciona com solicitações e respostas	690
Maneiras de usar o Lambda@Edge	691
Começar a usar o Lambda@Edge	692
Configurar permissões e perfis do IAM	700
Escrever e criar uma função do Lambda@Edge	707
Adicionar acionadores para uma função do Lambda@Edge	712
Testar e depurar	720
Excluir funções e réplicas	728
Estrutura de eventos	729
Trabalho com solicitações e respostas	746
Exemplos de funções	752
Restrições das funções de borda	791
Restrições de todas as funções de borda	791
Restrições do CloudFront Functions	798
Restrições ao Lambda@Edge	800

Relatórios, métricas e logs	806
Relatórios de uso e faturamento da AWS para o CloudFront	806
Visualizar o relatório de faturamento da AWS para o CloudFront	807
Visualizar o relatório de uso da AWS para o CloudFront	808
Interpretar os relatórios de uso e faturamento da AWS para o CloudFront	810
Visualizar relatórios do console do CloudFront	816
Visualizar relatórios de estatísticas de cache do CloudFront	817
Visualizar relatórios de objetos comuns do CloudFront	824
Visualizar relatórios dos principais indicadores do CloudFront	830
Visualizar relatórios de uso do CloudFront	833
Visualizar relatórios de visualizadores do CloudFront	842
Monitorar métricas do CloudFront com o Amazon CloudWatch	853
Visualizar métricas do CloudFront e de funções de borda	854
Criar alarmes	862
Baixar dados de métricas	863
Obter métricas usando a API do CloudWatch	866
Registro em log do CloudFront e de funções de borda	872
Registrar solicitações em log	873
Registrar em log funções de borda	873
Registrar em log as atividades do serviço	874
Usar logs padrão (logs de acesso)	874
Usar logs em tempo real	894
Logs de funções de borda	916
Logs do AWS CloudTrail	919
Monitorar as alterações de configuração com o AWS Config	932
Configurar o AWS Config com o CloudFront	933
Visualizar o histórico de configuração do CloudFront	934
Avaliar as configurações do CloudFront com o AWS Config Rules	935
Segurança	936
Proteção de dados	937
Criptografia em trânsito	938
Criptografia em repouso	939
Restringir o acesso ao conteúdo	939
Identity and Access Management	940
Público	941
Autenticando com identidades	942

Gerenciando acesso usando políticas	945
Como o Amazon CloudFront funciona com o IAM	948
Exemplos de políticas baseadas em identidade	955
Políticas gerenciadas pela AWS	965
Solução de problemas de identidade e acesso do CloudFront	971
Logging e monitoramento	973
Validação de conformidade	974
Melhores práticas de conformidade do CloudFront	975
Resiliência	977
Failover da origem do CloudFront	977
Segurança da infraestrutura	977
Solução de problemas	979
Como solucionar problemas de distribuição	979
O CloudFront retorna um erro Access Denied	979
O CloudFront retorna um erro InvalidViewerCertificate quando tento adicionar um nome de domínio alternativo	982
O CloudFront retorna um erro de registro DNS configurado incorretamente quando tento adicionar um novo CNAME	984
Não consigo visualizar os arquivos na minha distribuição	984
Mensagem de erro: Certificate: <certificate-id> Is Being Used by CloudFront (O certificado: <certificate-id> está sendo usado pelo CloudFront)	986
Solucionar problemas de códigos de status de resposta de erros	986
Código de status HTTP 400 (solicitação inválida)	987
Código de status HTTP 401 (Não autorizado)	989
Código de status HTTP 403 (Permissão negada)	989
Código de status HTTP 404 (Não encontrado)	992
Código de status HTTP 405 (Método não permitido)	993
Código de status HTTP 412 (Falha na pré-condição)	993
Código de status HTTP 500 (Erro interno do servidor)	993
Código de status HTTP 502 (Gateway inválido)	994
Código de status HTTP 503 (Serviço indisponível)	999
Código de status HTTP 504 (tempo limite do gateway)	1002
Testes de carga do CloudFront	1007
Cotas	1009
Cotas gerais	1010
Cotas gerais para distribuições	1010

Cotas gerais para políticas	1013
Cotas no CloudFront Functions	1015
Cotas em armazenamentos de chave-valor	1015
Cotas do Lambda@Edge	1016
Cotas para certificados SSL	1017
Cotas para invalidações	1018
Cotas em grupos de chave	1019
Cotas para conexões do WebSocket	1019
Cotas para criptografia no nível de campo	1020
Cotas para cookies (configurações de cache herdadas)	1021
Cotas em cadeias de consulta (configurações de cache herdadas)	1021
Cotas para cabeçalhos	1022
Informações relacionadas	1023
Exemplos de código	1024
Conceitos básicos	1025
Ações	1026
Cenários	1085
Excluir recursos de assinatura	1086
Assinar URLs e cookies	1088
Exemplos do CloudFront Functions	1091
Adicionar cabeçalhos de segurança HTTP	1092
Adicionar um cabeçalho de CORS	1094
Adicionar um cabeçalho de controle de cache	1095
Adicionar um cabeçalho IP de cliente verdadeiro	1095
Adicionar um cabeçalho de origem	1096
Adicionar index.html a URLs de solicitação	1097
Normalizar parâmetros de string de consulta	1098
Redirecionar para um novo URL	1099
Reescrever um URI de solicitação	1100
Usar pares de chave-valor	1103
Validar um token simples	1104
Histórico do documento	1109

O que é o Amazon CloudFront?

O Amazon CloudFront é um serviço da web que acelera a distribuição do conteúdo estático e dinâmico da web, como arquivos .html, .css, .js e arquivos de imagem, para os usuários. O CloudFront distribui o conteúdo por meio de uma rede global de datacenters denominados pontos de presença. Quando um usuário solicita um conteúdo que você está disponibilizando com o CloudFront, a solicitação é roteada para o ponto de presença que fornece a menor latência (atraso), assim o conteúdo é entregue com a melhor performance possível.

- Se o conteúdo já estiver no ponto de presença com a menor latência, o CloudFront o entregará imediatamente.
- Se o conteúdo não estiver nesse ponto de presença, o CloudFront o recuperará de uma origem definida, como um bucket do Amazon S3, um canal do MediaPackage ou um servidor HTTP (por exemplo, um servidor web), que você identificou como a fonte para a versão definitiva do conteúdo.

Por exemplo, suponha que você esteja exibindo uma imagem de um servidor web tradicional, e não do CloudFront. Por exemplo, você pode fornecer uma imagem, sunsetphoto.png, usando a URL `https://example.com/sunsetphoto.png`.

Seus usuários podem navegar facilmente para esse URL e ver a imagem. Mas provavelmente não sabem que sua solicitação é roteada de uma rede para outra - por meio da coleção complexa de redes interconectadas que compõem a Internet - até que a imagem seja encontrada.

O CloudFront acelera a distribuição do seu conteúdo encaminhando cada solicitação de usuário por meio da rede de estrutura da AWS para o local da borda capaz de veicular melhor seu conteúdo. Normalmente, esse é um servidor de borda do CloudFront que fornece a entrega mais rápida ao visualizador. Usar a rede da AWS reduz drasticamente o número de redes pelas quais as solicitações dos usuários devem passar, melhorando o desempenho. Os usuários obtêm menos latência (o tempo que leva para carregar o primeiro byte do arquivo) e taxas de transferência de dados maiores.

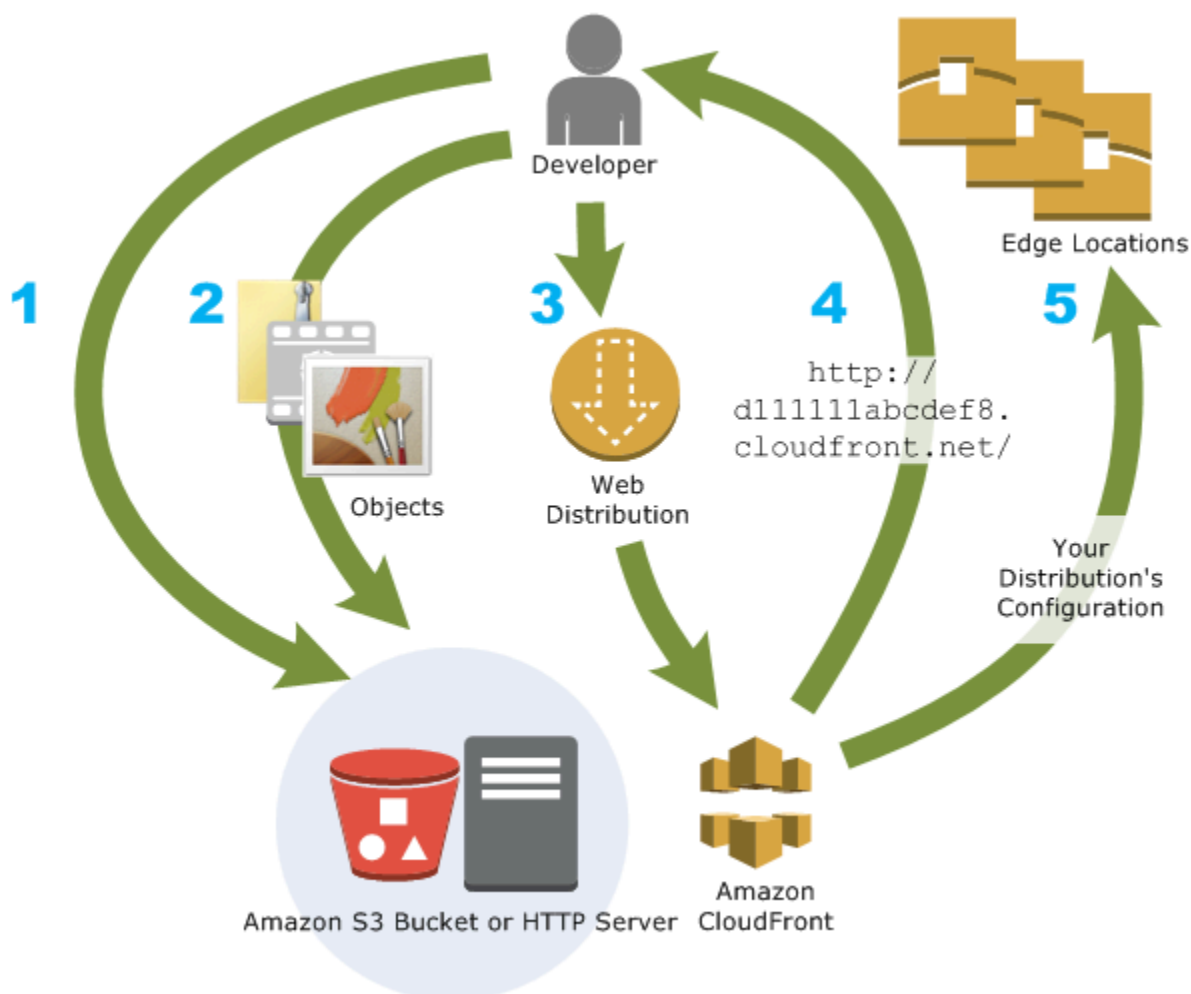
Você também pode obter mais confiabilidade e disponibilidade porque as cópias de seus arquivos (também conhecidos como objetos) agora são mantidos (ou armazenados em cache) em vários pontos de presença em todo o mundo.

Tópicos

- [Como configurar o CloudFront para entregar conteúdo](#)
- [Definição de preço](#)
- [Maneiras de usar o CloudFront](#)
- [Como o CloudFront entrega conteúdo](#)
- [Localizações e intervalos de endereço IP dos servidores de borda do CloudFront](#)
- [Como usar o CloudFront com um SDK da AWS](#)
- [Recursos técnicos do CloudFront](#)

Como configurar o CloudFront para entregar conteúdo

Você cria uma distribuição do CloudFront para informar ao CloudFront de onde você deseja que o conteúdo seja entregue e os detalhes sobre como rastrear e gerenciar a entrega do conteúdo. O CloudFront usa os computadores, servidores de borda, que estão próximos dos visualizadores para entregar esse conteúdo rapidamente quando alguém desejar vê-lo ou usá-lo.



Como configurar o CloudFront para entregar seu conteúdo

1. Especifique os servidores de origem, como um bucket do Amazon S3 ou seu próprio servidor HTTP, dos quais o CloudFront obtém os arquivos que serão distribuídos de pontos de presença do CloudFront no mundo todo.

Um servidor de origem armazena a versão original e definitiva de seus objetos. Se você estiver fornecendo conteúdo por HTTP, o servidor de origem será um bucket do Amazon S3 ou servidor HTTP, como um servidor da web. Seu servidor HTTP pode ser executado em uma instância do Amazon Elastic Compute Cloud (Amazon EC2) ou em um servidor gerenciado por você. Esses servidores também são conhecidos como origens personalizadas.

2. Faça upload dos seus arquivos nos servidores de origem. Seus arquivos, também conhecidos como objetos, geralmente incluem páginas da Web, imagens e arquivos de mídia, mas podem ser qualquer coisa capaz de ser oferecida por HTTP.

Se você estiver usando um bucket do Amazon S3 como servidor de origem, poderá tornar os objetos dele publicamente legíveis, para que qualquer pessoa com os URLs do CloudFront dos objetos possa acessá-los. Você também tem a opção de manter os objetos privados e controlar quem os acessa. Consulte [Veicular conteúdo privado com URLs e cookies assinados](#).

3. Crie uma distribuição do CloudFront, que informa ao CloudFront de quais servidores de origem obter os arquivos quando os usuários os solicitam no site ou na aplicação. Especifique também detalhes, por exemplo, se deseja que o CloudFront registre todas as solicitações e que a distribuição seja ativada quando criada.
4. O CloudFront atribui um nome de domínio à nova distribuição que pode ser visto no console do CloudFront ou que é retornado na resposta a uma solicitação programática, por exemplo, uma solicitação de API. Em vez disso, se preferir, você pode adicionar um nome de domínio alternativo.
5. O CloudFront envia a configuração (mas não o conteúdo) da distribuição a todos os pontos de presença ou POPs, que são conjuntos de servidores em datacenters geograficamente dispersos nos quais o CloudFront armazena cópias dos objetos em cache.

Ao desenvolver o site ou a aplicação, use o nome de domínio fornecido pelo CloudFront para seus URLs. Por exemplo, se o CloudFront retornar `d111111abcdef8.cloudfront.net` como o nome de domínio da distribuição, o URL de `logo.jpg` do bucket do Amazon S3 (ou diretório raiz de um servidor HTTP) será `https://d111111abcdef8.cloudfront.net/logo.jpg`.

Ou é possível configurar o CloudFront para usar seu próprio nome de domínio com a distribuição. Nesse caso, o URL pode ser `https://www.example.com/logo.jpg`.

Opcionalmente, é possível configurar o servidor de origem para adicionar cabeçalhos aos arquivos a fim de indicar o tempo de permanência dos arquivos no cache dos pontos de presença do CloudFront. Por padrão, cada arquivo permanece em um ponto de presença por 24 horas antes de expirar. O tempo mínimo de expiração é 0 segundo. Não há um tempo máximo de expiração. Para ter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Definição de preço

O CloudFront cobra pelas transferências de dados de seus locais da borda, junto com solicitações HTTP ou HTTPS. Os preços variam de acordo com o tipo de uso, a região geográfica e a seleção de atributos.

A transferência de dados da sua origem para o CloudFront é sempre gratuita ao usar origens da AWS como o Amazon Simple Storage Service (Amazon S3), o Elastic Load Balancing ou o Amazon API Gateway. Você só é cobrado pela transferência de dados de saída do CloudFront para o visualizador ao usar as origens da AWS.

Para ter mais informações, consulte a [Definição de preços do CloudFront](#) e as [Perguntas frequentes](#) de faturamento e do pacote Savings.

Maneiras de usar o CloudFront

Usar o CloudFront pode ajudar você a atingir vários objetivos. Esta seção lista apenas alguns deles, juntamente com links para mais informações. Assim, você terá uma ideia das possibilidades.

Tópicos

- [Acelerar a entrega de conteúdo de site estático](#)
- [Fornecer vídeos de streaming ao vivo ou sob demanda](#)
- [Criptografar campos específicos durante todo o processamento do sistema](#)
- [Personalizar na borda](#)
- [Fornecer conteúdo privado usando personalizações do Lambda@Edge](#)

Acelerar a entrega de conteúdo de site estático

O CloudFront pode acelerar a entrega do conteúdo estático (por exemplo, imagens, folhas de estilo, JavaScript e assim por diante) para visualizadores no mundo todo. Ao usar o CloudFront, é possível aproveitar as vantagens da rede de estrutura da AWS e dos servidores de borda do CloudFront para oferecer aos seus visualizadores uma experiência rápida, segura e confiável ao visitar seu site.

Uma abordagem simples para armazenar e fornecer conteúdo estático é usar um bucket do Amazon S3. Usar o S3 com o CloudFront oferece uma série de vantagens, incluindo a opção de usar [controle de acesso à origem](#) para restringir facilmente o acesso ao conteúdo do Amazon S3.

Consulte mais informações sobre como usar o Amazon S3 em conjunto com o CloudFront, incluindo um modelo do AWS CloudFormation para ajudar você a começar rapidamente, em [Conceitos básicos de um site estático seguro](#).

Fornecer vídeos de streaming ao vivo ou sob demanda

O CloudFront oferece várias opções de streaming de mídia para visualizadores globais, tanto de arquivos pré-gravados quanto de eventos ao vivo.

- Para streaming de vídeo sob demanda (VOD), é possível usar o CloudFront para fazer streaming em formatos comuns, como MPEG DASH, Apple HLS, Microsoft Smooth Streaming e CMAF, para qualquer dispositivo.
- Para fazer uma transmissão ao vivo, você pode armazenar em cache fragmentos de mídia no ponto. Assim, várias solicitações para o arquivo manifesto que entrega os fragmentos na ordem correta poderão ser combinadas para reduzir a carga no servidor de origem.

Para mais informações sobre como fornecer conteúdo de streaming com o CloudFront, consulte [Vídeo sob demanda e vídeo de transmissão ao vivo com o CloudFront](#).

Criptografar campos específicos durante todo o processamento do sistema

Ao configurar o HTTPS com o CloudFront, você já terá conexões de ponta a ponta seguras com os servidores de origem. Ao adicionar criptografia no nível do campo, você poderá proteger dados específicos durante todo o processamento do sistema, além da segurança HTTPS, para que apenas determinados aplicativos na sua origem vejam os dados.

Para configurar a criptografia no nível do campo, adicione uma chave pública ao CloudFront e especifique o conjunto de campos que você quer criptografar com a chave. Para obter mais informações, consulte [Use criptografia de nível de campo para ajudar a proteger dados confidenciais](#).

Personalizar na borda

A execução de código sem servidor no ponto abre várias possibilidades para personalizar o conteúdo e a experiência dos espectadores com latência reduzida. Por exemplo, você pode retornar uma mensagem de erro personalizada quando o servidor de origem estiver inativo para manutenção para que os visualizadores não recebam uma mensagem de erro HTTP genérica. Se preferir, você pode usar uma função para ajudar a autorizar usuários e controlar o acesso ao seu conteúdo antes que o CloudFront encaminhe uma solicitação para a origem.

Usar o Lambda@Edge com o CloudFront oferece várias formas de personalizar o conteúdo que o CloudFront entrega. Para saber mais sobre o Lambda@Edge e como criar e implantar funções com o CloudFront, consulte [Personalização na borda com o Lambda@Edge](#). Se você quiser ver exemplos de código personalizáveis para suas próprias soluções, consulte [Funções de exemplo do Lambda@Edge](#).

Fornecer conteúdo privado usando personalizações do Lambda@Edge

Usar o Lambda@Edge pode ajudar você a configurar a distribuição do CloudFront para veiculação de conteúdo privado de sua própria origem personalizada, além de usar URLs assinados ou cookies assinados.

Para fornecer conteúdo privado usando o CloudFront, faça o seguinte:

- Exija que os usuários (visualizadores) acessem o conteúdo usando [URLs assinados ou cookies assinados](#).
- Restrinja o acesso à sua origem para que ela esteja disponível apenas nos servidores voltados para a origem do CloudFront. Para fazer isso, utilize uma das seguintes opções:
 - Para uma origem do Amazon S3, você pode [usar um controle de acesso à origem \(OAC\)](#).
 - Para uma origem personalizada, você pode fazer o seguinte:
 - Se a origem personalizada estiver protegida por um grupo de segurança da Amazon VPC ou pelo AWS Firewall Manager, você pode [usar a lista de prefixos gerenciados do CloudFront](#) a fim de permitir tráfego de entrada para sua origem somente de endereços IP voltados para a origem do CloudFront.

- Use um cabeçalho HTTP personalizado para restringir o acesso somente a solicitações do CloudFront. Para ter mais informações, consulte [the section called “Restringir o acesso a arquivos em origens personalizadas”](#) e [the section called “Adicionar cabeçalhos personalizados às solicitações de origem”](#). Para obter um exemplo que usa um cabeçalho personalizado para restringir o acesso a uma origem do Application Load Balancer, consulte [the section called “Restringir o acesso aos Application Load Balancers”](#).
- Se a origem personalizada exigir lógica de controle de acesso personalizada, você poderá usar o Lambda@Edge para implementar essa lógica, conforme descrito nesta postagem de blog: [Serving Private Content Using Amazon CloudFront & Lambda@Edge](#) (Distribuição de conteúdo privado usando o Amazon CloudFront e o Lambda@Edge).

Como o CloudFront entrega conteúdo

Após a configuração inicial, o CloudFront trabalhará em conjunto com o site ou a aplicação e agilizará a entrega do seu conteúdo. Esta seção explica como o CloudFront veicula seu conteúdo quando os visualizadores o solicitam.

Tópicos

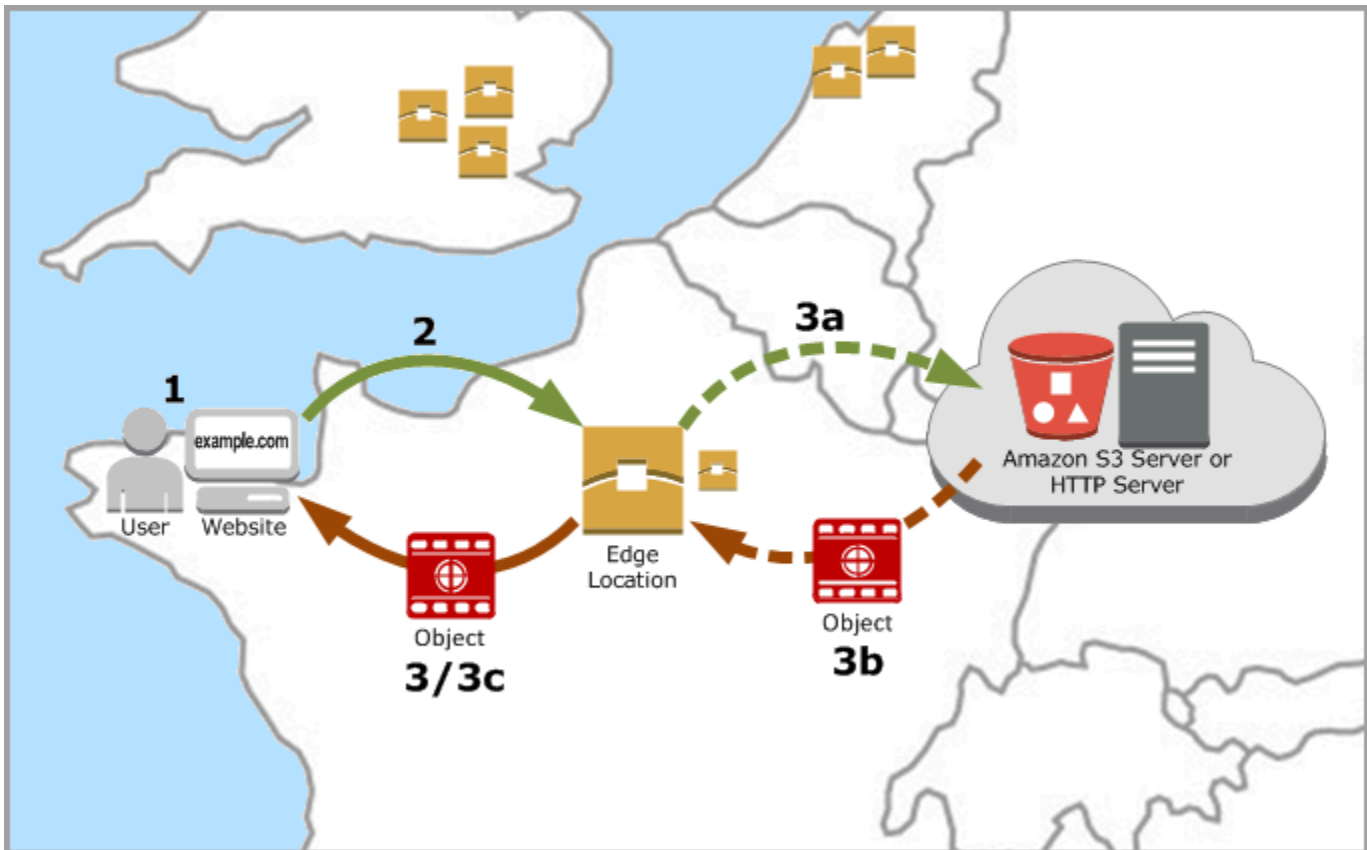
- [Como o CloudFront entrega conteúdo aos usuários](#)
- [Como o CloudFront funciona com caches de borda regionais](#)

Como o CloudFront entrega conteúdo aos usuários

Depois de configurar o CloudFront para entregar seu conteúdo, veja o que ocorre quando os usuários solicitam seus objetos:

1. Um usuário acessa seu site ou sua aplicação e solicita um objeto, como um arquivo de imagem ou um arquivo HTML.
2. O DNS encaminha a solicitação para o POP (local da borda) do CloudFront que melhor atende à solicitação, normalmente o POP do CloudFront mais próximo em termos de latência.
3. O CloudFront procura o objeto solicitado no cache. Se o objeto estiver no cache, o CloudFront o retornará ao usuário. Se o objeto não estiver no cache, o CloudFront fará o seguinte:
 - a. O CloudFront compara a solicitação com as especificações da distribuição e encaminha a solicitação ao servidor de origem para o objeto correspondente. Por exemplo, para o bucket do Amazon S3 ou o servidor HTTP.

- b. O servidor de origem enviará o objeto de volta ao local da borda.
- c. Assim que o primeiro byte chegar da origem, o CloudFront começará a encaminhar o objeto ao usuário. O CloudFront também adicionará o objeto no cache na próxima vez em que alguém o solicitar.



Como o CloudFront funciona com caches de borda regionais

Os pontos de presença (também conhecidos como POPs ou locais de borda) do CloudFront garantem que um conteúdo muito requisitado possa ser veiculado rapidamente aos seus visualizadores. O CloudFront também tem caches de borda regionais que aproximam mais seu conteúdo dos visualizadores, mesmo quando o conteúdo não é procurado o suficiente para permanecer em um POP, para ajudar a melhorar a performance desse conteúdo.

Os caches de ponto regional ajudam com todos os tipos de conteúdo, especialmente aqueles que tendem a se tornar menos populares com o tempo. Exemplos incluem conteúdo gerado pelo usuário, como vídeos, fotos ou arte; ativos de comércio eletrônico, como fotos e vídeos de produtos; e notícias e conteúdo relacionado a eventos, que pode repentinamente mudar de popularidade.

Como funcionam os caches regionais

Caches de borda regionais são locais do CloudFront implantados globalmente, próximos aos visualizadores. Eles estão localizados entre seu servidor de origem e os POPs, locais de borda globais que fornecem conteúdo diretamente aos visualizadores. À medida que os objetos se tornam menos populares, os POPs individuais podem removê-los para dar lugar a conteúdo mais procurado. Os caches de ponto regionais têm um cache maior que um POP individual, portanto, os objetos permanecem no cache por mais tempo no local do cache de ponto regional mais próximo. Isso ajuda a manter mais conteúdo perto dos visualizadores, reduzindo a necessidade de acesso ao servidor de origem pelo CloudFront e aumentando a performance geral para os visualizadores.

Quando um visualizador faz uma solicitação em seu site ou por meio de seu aplicativo, o DNS roteia a solicitação para o POP que melhor atende à solicitação do usuário. Normalmente, essa localização é o local de borda do CloudFront mais próximo em termos de latência. No POP, o CloudFront procura o objeto solicitado no cache. Se o objeto estiver no cache, o CloudFront o retornará ao usuário. Se o objeto não estiver no cache, o POP normalmente acessará o cache de borda regional mais próximo para obtê-lo. Para obter mais informações sobre quando o POP ignora o cache de borda regional e acessa diretamente a origem, consulte a observação a seguir.

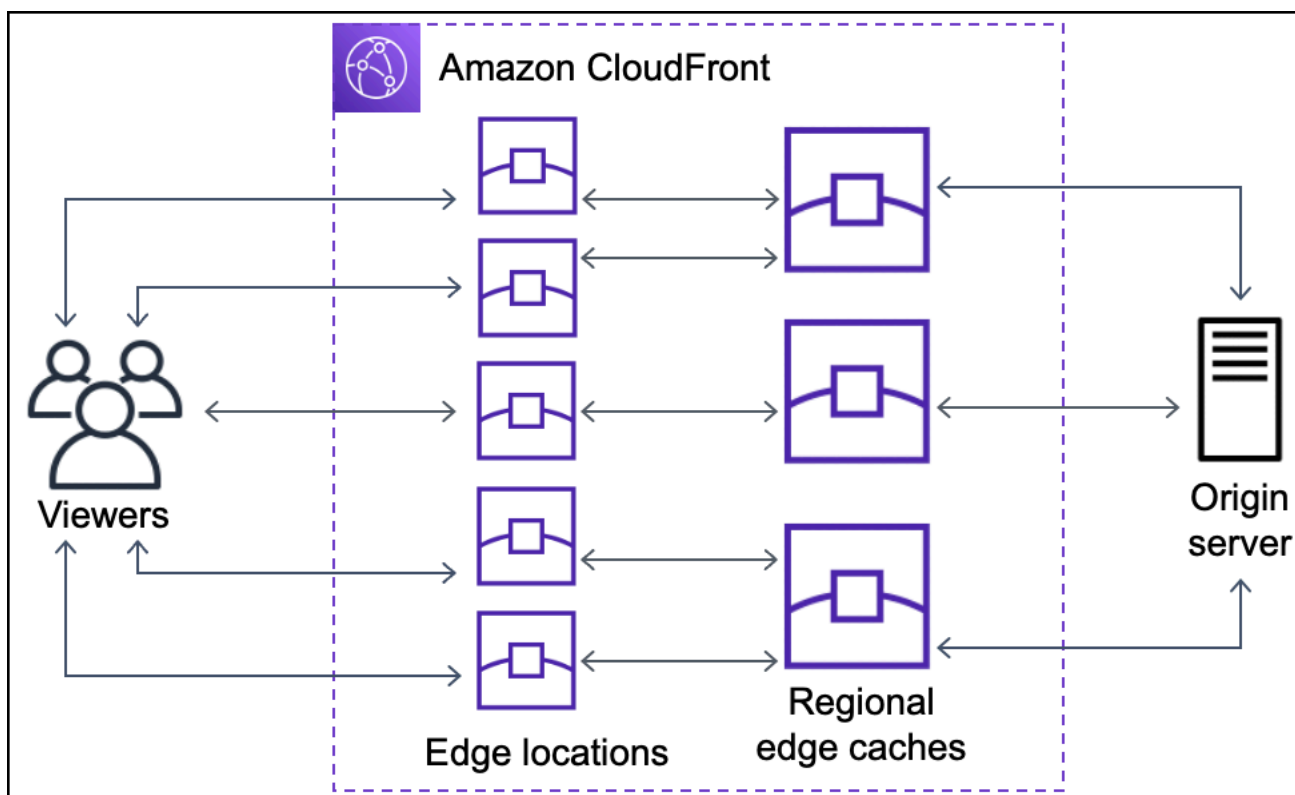
Na localização de cache de borda regional, o CloudFront procura novamente o objeto solicitado em seu cache. Se o objeto estiver no cache, o CloudFront o encaminhará para o POP que o solicitou. Assim que o primeiro byte chega ao cache de borda regional, o CloudFront começa a encaminhar o objeto ao usuário. O CloudFront também adicionará o objeto ao cache no POP na próxima vez em que alguém o solicitar.

Para objetos não armazenados em cache no POP ou no cache de borda regional, o CloudFront compara a solicitação com as especificações nas distribuições e encaminha a solicitação ao servidor de origem. Depois que o servidor de origem envia o objeto à localização do cache de borda regional, ele é encaminhado ao POP, e o CloudFront o encaminha ao usuário. Nesse caso, o CloudFront também adiciona o objeto à localização do cache de borda regional, além do POP, para a próxima vez em que um visualizador o solicitar. Isso garante que todos os POPs de uma região compartilhem um cache local, eliminando várias solicitações aos servidores de origem. O CloudFront também mantém conexões persistentes com os servidores de origem para que os objetos sejam obtidos das origens o mais rapidamente possível.

Note

- Os caches de borda regionais têm paridade de recursos com os POPs. Por exemplo, uma solicitação de invalidação de cache remove um objeto dos caches do POP e dos caches de ponto regionais antes que ele expire. A próxima vez em que um visualizador solicitar o objeto, o CloudFront recorrerá à origem para obter a versão mais recente dele.
- Métodos de proxy HTTP (PUT, POST, PATCH, OPTIONS e DELETE) fluem diretamente para a origem dos POPs e não usam proxy por meio dos caches de borda regionais.
- As solicitações dinâmicas, conforme determinado no momento da solicitação, não fluem por meio de caches de borda regionais, mas vão diretamente para a origem.
- Quando a origem for um bucket do Amazon S3 e o cache de borda regional ideal da solicitação estiver na mesma Região da AWS que o bucket do S3, o POP ignorará o cache de borda regional e irá diretamente para o bucket do S3.

O diagrama a seguir ilustra como as solicitações e respostas fluem por meio de locais de borda do CloudFront e caches de borda regionais.



Localizações e intervalos de endereço IP dos servidores de borda do CloudFront

Para obter uma lista dos locais dos servidores de borda do CloudFront, consulte a página [Rede de borda global do Amazon CloudFront](#).

A Amazon Web Services (AWS) publica seus intervalos de endereços IP atuais em formato JSON. Para visualizar os intervalos atuais, faça download do arquivo [ip-ranges.json](#). Para obter mais informações, consulte [Intervalos de endereços IP da AWS](#) no Referência geral da Amazon Web Services.

Para encontrar os intervalos de endereços IP associados aos servidores de borda do CloudFront, pesquise por ip-ranges.json na seguinte string:

```
"region": "GLOBAL",  
"service": "CLOUDFRONT"
```

Como alternativa, é possível visualizar somente os intervalos de IP do CloudFront <https://d7uri8nf7uskq.cloudfront.net/tools/list-cloudfront-ips>.

Use a lista de prefixos gerenciados do CloudFront

A lista de prefixos gerenciados do CloudFront contém os intervalos de endereços IP de todos os servidores voltados para a origem distribuídos globalmente do CloudFront. Se a origem estiver hospedada na AWS e protegida por um [grupo de segurança](#) da Amazon VPC, você poderá usar a lista de prefixos gerenciados do CloudFront para permitir o tráfego de entrada para a origem somente de servidores voltados para a origem do CloudFront, impedindo que qualquer tráfego que não seja do CloudFront atinja a origem. O CloudFront mantém a lista de prefixos gerenciados para que esteja sempre atualizada com os endereços IP de todos os servidores voltados para a origem global do CloudFront. Com a lista de prefixos gerenciados do CloudFront, você não precisa ler nem manter uma lista de intervalos de endereços IP por conta própria.

Por exemplo, imagine que sua origem seja uma instância do Amazon EC2 na região Europa (Londres) (eu-west-2). Se a instância estiver em uma VPC, você poderá criar uma regra de grupo de segurança que permita acesso HTTPS de entrada usando a lista de prefixos gerenciados do CloudFront. Isso permite que todos os servidores voltados para a origem global do CloudFront alcancem a instância. Se você remover todas as outras regras de entrada do grupo de segurança, impedirá que qualquer tráfego que não seja do CloudFront chegue à instância.

A lista de prefixos gerenciados do CloudFront é nomeada como `com.amazonaws.global.cloudfront.origin-facing`. Para obter mais informações, consulte [Usar uma lista de prefixos gerenciados da AWS](#) no Guia do usuário da Amazon VPC.

Important

A lista de prefixos gerenciados do CloudFront é singular na forma como ela se aplica às cotas da Amazon VPC. Para obter mais informações, consulte [Peso da lista de prefixos gerenciados da AWS](#) no Guia do usuário da Amazon VPC.

Como usar o CloudFront com um SDK da AWS

Os kits de desenvolvimento de software (SDKs) da AWS estão disponíveis para muitas linguagens de programação populares. Cada SDK fornece uma API, exemplos de código e documentação que facilitam a criação de aplicações em seu idioma preferido pelos desenvolvedores.

Documentação do SDK	Exemplos de código
AWS SDK for C++	Exemplos de código do AWS SDK for C++
AWS CLI	Exemplos de código do AWS CLI
AWS SDK for Go	Exemplos de código do AWS SDK for Go
AWS SDK for Java	Exemplos de código do AWS SDK for Java
AWS SDK for JavaScript	Exemplos de código do AWS SDK for JavaScript
AWS SDK para Kotlin	Exemplos de código do AWS SDK para Kotlin
AWS SDK for .NET	Exemplos de código do AWS SDK for .NET
AWS SDK for PHP	Exemplos de código do AWS SDK for PHP
AWS Tools for PowerShell	Tools for PowerShell code examples
AWS SDK for Python (Boto3)	Exemplos de código do AWS SDK for Python (Boto3)

Documentação do SDK	Exemplos de código
AWS SDK for Ruby	Exemplos de código do AWS SDK for Ruby
AWS SDK para Rust	Exemplos de código do AWS SDK para Rust
SDK da AWS para SAP ABAP	Exemplos de código do SDK da AWS para SAP ABAP
AWS SDK for Swift	Exemplos de código do AWS SDK for Swift

Exemplo de disponibilidade

Você não consegue encontrar o que precisa? Solicite um código de exemplo no link Fornecer feedback na parte inferior desta página.

Recursos técnicos do CloudFront

Use os seguintes recursos para obter respostas a perguntas técnicas sobre o CloudFront:

- [AWSre:Post](#): um site de perguntas e respostas da comunidade para desenvolvedores discutirem questões técnicas relacionadas ao CloudFront.
- [AWS Support Center](#): esse site inclui informações sobre seus casos de suporte recentes e sobre os resultados das verificações de integridade do AWS Trusted Advisor. Ele também fornece links para fóruns de discussão, perguntas frequentes técnicas, o painel de integridade do serviço e informações sobre os planos do AWS Support.
- [AWS Premium Support](#): conheça o AWS Premium Support, um canal de suporte de resposta rápida e com atendimento individual, que ajuda você a criar e executar na AWS.
- [AWS IQ](#): receba ajuda de profissionais e especialistas certificados pela AWS.

Conceitos básicos do CloudFront

Os tópicos desta seção mostram como começar a distribuir seu conteúdo com o Amazon CloudFront.

O tópico [Configurar a Conta da AWS](#) descreve os pré-requisitos para os tutoriais a seguir, como criar uma Conta da AWS e criar um usuário com acesso administrativo.

O tutorial básico de distribuição mostra como configurar o controle de acesso de origem (OAC) para enviar solicitações autenticadas a uma origem do Amazon S3.

O tutorial do site estático seguro mostra como criar um site estático seguro para o nome de domínio usando OAC com uma origem do Amazon S3. O tutorial usa um modelo do Amazon CloudFront (CloudFront) para configuração e implantação.

Tópicos

- [Configurar a Conta da AWS](#)
- [Conceitos básicos de uma distribuição básica do CloudFront](#)
- [Conceitos básicos de um site estático seguro](#)

Configurar a Conta da AWS

Este tópico descreve as etapas preliminares, como a criação de uma Conta da AWS, para preparar você para usar o Amazon CloudFront.

Tópicos

- [Cadastrar-se em uma Conta da AWS](#)
- [Criar um usuário com acesso administrativo](#)
- [Escolha como acessar o CloudFront](#)

Cadastrar-se em uma Conta da AWS

Se você ainda não tem Conta da AWS, siga as etapas a seguir para criar um.

Para se cadastrar em uma Conta da AWS

1. Abra <https://portal.aws.amazon.com/billing/signup>.
2. Siga as instruções online.

Parte do procedimento de inscrição envolve receber uma chamada telefônica e inserir um código de verificação no teclado do telefone.

Quando você se cadastra em uma Conta da AWS, um Usuário raiz da conta da AWS é criado. O usuário raiz tem acesso a todos os Serviços da AWS e atributos na conta. Como prática recomendada de segurança, atribua o acesso administrativo a um usuário e use somente o usuário-raiz para executar [tarefas que exigem acesso de usuário-raiz](#).

A AWS envia um e-mail de confirmação depois que o processo de cadastramento é concluído. A qualquer momento, é possível visualizar as atividades da conta atual e gerenciar sua conta acessando <https://aws.amazon.com/> e selecionando Minha conta.

Criar um usuário com acesso administrativo

Depois de se cadastrar em uma Conta da AWS, proteja seu Usuário raiz da conta da AWS, habilite o AWS IAM Identity Center e crie um usuário administrativo para não usar o usuário raiz em tarefas cotidianas.

Proteger seu Usuário raiz da conta da AWS

1. Faça login no [AWS Management Console](#) como o proprietário da conta ao escolher a opção Usuário raiz e inserir o endereço de e-mail da Conta da AWS. Na próxima página, insira sua senha.

Para obter ajuda ao fazer login usando o usuário raiz, consulte [Fazer login como usuário raiz](#) no Guia do usuário do Início de Sessão da AWS.

2. Habilite a autenticação multifator (MFA) para o usuário raiz.

Para obter instruções, consulte [Habilitar um dispositivo MFA virtual para o usuário raiz de sua conta da Conta da AWS \(console\)](#) no Guia do usuário do IAM.

Criar um usuário com acesso administrativo

1. Habilitar o IAM Identity Center.

Para obter instruções, consulte [Habilitar AWS IAM Identity Center](#) no Guia do usuário do AWS IAM Identity Center.

2. No Centro de Identidade do IAM, conceda o acesso administrativo para um usuário.

Para obter um tutorial sobre como usar o Diretório do Centro de Identidade do IAM como fonte de identidade, consulte [Configurar o acesso dos usuários com o Diretório do Centro de Identidade do IAM padrão](#) no Guia do usuário do AWS IAM Identity Center.

Iniciar sessão como o usuário com acesso administrativo

- Para fazer login com seu usuário do Centro de Identidade do IAM, use o URL de login que foi enviado ao seu endereço de e-mail quando você criou o usuário do Centro do Usuário do IAM.

Para obter ajuda com o login utilizando um usuário do Centro de Identidade do IAM, consulte [Fazer login no portal de acesso da AWS](#), no Guia do usuário do Início de Sessão da AWS.

Atribuir acesso a usuários adicionais

1. No Centro de Identidade do IAM, crie um conjunto de permissões que siga as práticas recomendadas de aplicação de permissões com privilégio mínimo.

Para obter instruções, consulte [Create a permission set](#) no Guia do usuário do AWS IAM Identity Center.

2. Atribua usuários a um grupo e, em seguida, atribua o acesso de autenticação única ao grupo.

Para obter instruções, consulte [Add groups](#) no Guia do usuário do AWS IAM Identity Center.

Escolha como acessar o CloudFront

É possível acessar o Amazon CloudFront das seguintes maneiras:

- AWS Management Console: os procedimentos ao longo deste guia explicam como usar o AWS Management Console para realizar tarefas.
- AWS SDKs: se estiver usando uma linguagem de programação para a qual a AWS fornece um SDK, você poderá usar um SDK para acessar o CloudFront. Os SDKs simplificam a autenticação, integram-se com facilidade ao ambiente de desenvolvimento e fornecem acesso aos comandos do CloudFront. Para ter mais informações, consulte [Como usar o CloudFront com um SDK da AWS](#).
- API do CloudFront: se você estiver usando uma linguagem de programação para a qual um SDK não está disponível, consulte a [Referência da API do Amazon CloudFront](#) para obter informações sobre as ações de API e sobre como fazer solicitações de API.

- **AWS CLI:** a AWS Command Line Interface (AWS CLI) é uma ferramenta unificada para gerenciar Serviços da AWS. Para ter informações sobre como instalar e configurar a AWS CLI, consulte [Install or update to the latest version of the AWS CLI](#) no Guia do usuário da AWS Command Line Interface.
- **Ferramentas do Windows PowerShell:** se tiver experiência com o Windows PowerShell, você poderá preferir usar o AWS Tools for Windows PowerShell. Para obter mais informações, consulte [Instalar a AWS Tools for Windows PowerShell](#) no Guia do usuário da AWS Tools for Windows PowerShell.

Conceitos básicos de uma distribuição básica do CloudFront

Os procedimentos nesta seção mostram como usar o CloudFront para definir uma configuração básica que faça o seguinte:

- Cria um bucket para usar como sua origem de distribuição.
- Armazena as versões originais de seus objetos em um bucket do Amazon Simple Storage Service (Amazon S3).
- Usa o controle de acesso à origem (OAC) para enviar solicitações autenticadas à sua origem do Amazon S3. O OAC envia solicitações por meio do CloudFront para impedir que os visualizadores acessem seu bucket do S3 diretamente. Para ter mais informações sobre OAC, consulte [Restringir o acesso a uma origem do Amazon Simple Storage Service](#).
- Usa o nome de domínio do CloudFront em URLs para seus objetos (por exemplo, `https://d111111abcdef8.cloudfront.net/index.html`).
- Armazena seus objetos nos locais da borda do CloudFront pela duração padrão de 24 horas (a duração mínima é de 0 segundo).

É possível personalizar a maioria dessas opções. Para obter informações sobre como personalizar as opções de distribuição do CloudFront, consulte [Criar uma distribuição](#).

Tópicos

- [Pré-requisitos](#)
- [Criar um bucket do Amazon S3](#)
- [Fazer upload do conteúdo no bucket](#)
- [Criar uma distribuição do CloudFront que use uma origem do Amazon S3 com OAC](#)

- [Acessar o conteúdo por meio do CloudFront](#)
- [Limpeza](#)
- [Aprimorar a distribuição básica](#)

Pré-requisitos

Antes de começar, certifique-se de que você concluiu as etapas em [Configurar a Conta da AWS](#).

Criar um bucket do Amazon S3

Um bucket do Amazon S3 é um contêiner destinado a arquivos (objetos) ou pastas. O CloudFront pode distribuir praticamente qualquer tipo de arquivo quando um bucket do S3 é a origem. Por exemplo, o CloudFront pode distribuir texto, imagens e vídeos. Não há máximo para a quantidade de dados que você pode armazenar no Amazon S3.

Para este tutorial, crie um bucket do S3 com os exemplos de arquivos `hello world` fornecidos que você usará para criar uma página da web básica.

Para criar um bucket

1. Faça login no AWS Management Console e abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. Recomendamos que você use nosso exemplo de Hello World para começar. Baixe o arquivo da página da web hello world: [hello-world-html.zip](#). Descompacte e salve a pasta `css` e o arquivo `index` em um local acessível, como a área de trabalho em que você está executando o navegador.
3. Selecione Criar bucket.
4. Insira um Bucket name (Nome de bucket) exclusivo que esteja em conformidade com as [Regras de nomenclatura de buckets de uso geral](#) no Guia do usuário do Amazon Simple Storage Service.
5. Para Região, recomendamos escolher uma Região da AWS que esteja geograficamente perto de você. (Isso reduz a latência e os custos.)
 - Escolher uma região diferente também funciona. Você pode fazer isso para atender aos requisitos regulamentares, por exemplo.
6. Deixe todas as outras configurações no modo padrão e escolha Create bucket (Criar bucket).

Fazer upload do conteúdo no bucket

Depois de criar o bucket do Amazon S3, carregue o conteúdo do arquivo descompactado `hello world` nele. (Você baixou e descompactou esse arquivo.) [Criar um bucket do Amazon S3](#)

Para fazer upload do conteúdo no Amazon S3

1. Na seção Buckets de uso geral, escolha o nome do seu novo bucket.
2. Escolha Carregar.
3. Na página Fazer upload, arraste a pasta `css` e o arquivo `index` na área para soltá-los.
4. Deixe todas as outras configurações no modo padrão e escolha Fazer upload.

Criar uma distribuição do CloudFront que use uma origem do Amazon S3 com OAC

Neste tutorial, você irá criar uma distribuição básica do CloudFront que usa uma origem do Amazon S3 com um controle de acesso à origem (OAC). O OAC ajuda a enviar com segurança solicitações autenticadas à sua origem do Amazon S3. Para ter mais informações sobre OAC, consulte [Restringir o acesso a uma origem do Amazon Simple Storage Service](#).

Como criar uma distribuição do CloudFront com uma origem do Amazon S3 que use OAC

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha Create distribution (Criar distribuição).
3. Em Origin (Origem), Origin domain (Domínio de origem), selecione o bucket do S3 criado para este tutorial.
4. Em Origin (Origem), Origin access (Acesso à origem), selecione Origin access control settings (recommended) (Configurações de controle de acesso à origem [recomendado]).
5. Em Origin access control (Controle de acesso à origem), escolha Create new OAC (Criar novo OAC).
6. No painel Create new OAC (Criar novo OAC), mantenha as configurações padrão e escolha Create (Criar).
7. Em Web Application Firewall (WAF), selecione uma das opções.
8. Em todas as outras seções e configurações, aceite os valores padrão. Para obter mais informações sobre essas opções, consulte [Configurações de distribuição](#).

9. Escolha Create distribution (Criar distribuição).
10. No banner The S3 bucket policy needs to be updated (A política de bucket do S3 precisa ser atualizada), leia a mensagem e escolha Copy policy (Copiar política).
11. No mesmo banner, escolha o link para acessar as permissões do bucket do S3 para atualizar a política. (Isso levará você à página de detalhes do bucket no console do Amazon S3.)
12. Em Bucket policy (Política de bucket), escolha Edit (Editar).
13. No campo Edit statement (Editar instrução), cole a política copiada na etapa 10.
14. Escolha Salvar alterações.
15. Retorne ao console do CloudFront e revise a seção Details (Detalhes) da sua nova distribuição. Quando sua distribuição terminar a implantação, o campo Last modified (Última modificação) muda de Deploying (Implantando) para uma data e hora.
16. Registre o nome de domínio que o CloudFront atribui à sua distribuição. É semelhante ao seguinte: `d111111abcdef8.cloudfront.net`.

Antes de usar a distribuição e o bucket do S3 deste tutorial em um ambiente de produção, certifique-se de configurá-los para atender às suas necessidades específicas. Para ter mais informações sobre como configurar o acesso em um ambiente de produção, consulte [Configurar o acesso seguro e restringir o acesso ao conteúdo](#).

Acessar o conteúdo por meio do CloudFront

Para acessar o conteúdo por meio do CloudFront, combine o nome de domínio da distribuição do CloudFront com o a página principal do conteúdo. (Você registrou seu nome de domínio de distribuição em [Criar uma distribuição do CloudFront que use uma origem do Amazon S3 com OAC](#).)

- O nome de domínio da distribuição pode ser semelhante a: `d111111abcdef8.cloudfront.net`.
- Normalmente, o caminho para a página principal de um site é `/index.html`.

Portanto, o URL para acessar o conteúdo por meio do CloudFront pode ser semelhante a:

```
https://d111111abcdef8.cloudfront.net/index.html
```

Se você seguiu as etapas anteriores e usou a página da web hello world, deverá ver uma página da web com a mensagem Hello world!.

Ao fazer upload de mais conteúdo para esse bucket do S3, você consegue acessar o conteúdo por meio do CloudFront combinando o nome de domínio da distribuição do CloudFront com o caminho para o objeto no bucket do S3. Por exemplo, se você fizer upload de um novo arquivo chamado `new-page.html` na raiz do bucket do S3, o URL será semelhante a:

```
https://d1111111abcdef8.cloudfront.net/new-page.html.
```

Limpeza

Se tiver criado a distribuição e o bucket do S3 apenas como exercício de aprendizado, poderá excluí-los, de maneira que deixe de acumular cobranças. Exclua a distribuição primeiro. Para obter mais informações, consulte os seguintes links:

- [Excluir uma distribuição do](#)
- [Excluir um bucket](#)

Aprimorar a distribuição básica

Este tutorial de conceitos básicos fornece um framework mínimo para a criação de uma distribuição. Recomendamos que você explore os seguintes aprimoramentos:

- Por padrão, os arquivos (objetos) no bucket do Amazon S3 são configurados como privados. Somente a Conta da AWS que criou o bucket tem permissão para ler ou gravar os arquivos. Para permitir que qualquer pessoa acesse os arquivos no bucket do Amazon S3 usando URLs do CloudFront, conceda permissões públicas de leitura aos objetos.
- Você pode usar o atributo de conteúdo privado do CloudFront para restringir o acesso ao conteúdo nos buckets do Amazon S3. Para obter mais informações sobre distribuição de conteúdo privado, consulte [Veicular conteúdo privado com URLs e cookies assinados](#).
- É possível configurar a distribuição do CloudFront para usar um nome de domínio personalizado (por exemplo, `www.example.com` em vez de `d1111111abcdef8.cloudfront.net`). Para ter mais informações, consulte [Usar URLs personalizados](#).
- Este tutorial usa uma origem do Amazon S3 com controle de acesso à origem (OAC). No entanto, não é possível usar o OAC se sua origem for um bucket do S3 configurado como um [endpoint de site](#). Se for esse o caso, você deverá configurar o bucket com o CloudFront como uma origem personalizada. Para ter mais informações, consulte [Usar um bucket do Amazon S3 configurado como um endpoint do site](#). Para ter mais informações sobre OAC, consulte [Restringir o acesso a uma origem do Amazon Simple Storage Service](#).

Conceitos básicos de um site estático seguro

É possível começar a usar o Amazon CloudFront com a solução descrita neste tópico para criar um site estático seguro para seu nome de domínio. Um site estático usa apenas arquivos estáticos, como HTML, CSS, JavaScript, imagens e vídeos, e não precisa de servidores nem de processamento no lado do servidor. Com essa solução, seu site obtém os seguintes benefícios:

- Usa o armazenamento durável do [Amazon Simple Storage Service \(Amazon S3\)](#): essa solução cria um bucket do Amazon S3 para hospedar o conteúdo estático do site. Para atualizar seu site, basta carregar os novos arquivos no bucket do S3.
- É acelerado pela rede de entrega de conteúdo do Amazon CloudFront: essa solução cria uma distribuição do CloudFront que o site forneça baixa latência aos visualizadores. A distribuição é configurada com um [controle de acesso à origem](#) (OAC) para garantir que o site seja acessível somente pelo CloudFront, não diretamente pelo S3.
- É protegido por HTTPS e cabeçalhos de segurança: essa solução cria um certificado SSL/TLS no [AWS Certificate Manager \(ACM\)](#) e o anexa à distribuição do CloudFront. Esse certificado permite que a distribuição veicule o site do seu domínio de forma segura com HTTPS.
- É configurado e implantado com o [AWS CloudFormation](#): esta solução usa um modelo do AWS CloudFormation para configurar todos os componentes para que você possa se concentrar mais no conteúdo do seu site e menos na configuração de componentes.

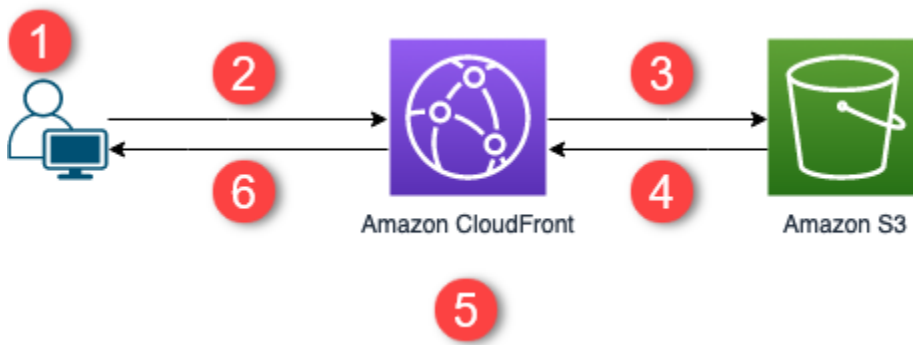
Essa solução é de código aberto no GitHub. Para visualizar o código, enviar uma solicitação pull ou abrir um problema, acesse <https://github.com/aws-samples/amazon-cloudfront-secure-static-site>.

Tópicos

- [Visão geral da solução](#)
- [Implante a solução](#)

Visão geral da solução

O diagrama a seguir mostra uma visão geral de como essa solução de site estático funciona:



1. O visualizador solicita o site em `www.exemplo.com`.
2. Se o objeto solicitado for armazenado em cache, o CloudFront retornará o objeto de seu cache ao visualizador.
3. Se o objeto não estiver no cache do CloudFront, ele solicitará o objeto da origem (um bucket do S3).
4. O S3 retorna o objeto ao CloudFront.
5. O CloudFront armazena o objeto em cache.
6. Os objetos são retornados ao visualizador. As solicitações subsequentes do objeto que chegam ao mesmo ponto de presença do CloudFront são atendidas com o cache do CloudFront.

Implante a solução

Para implantar a solução de site estático seguro, é possível escolher uma das seguintes opções:

- Use o console do AWS CloudFormation para implantar a solução com conteúdo padrão e faça upload do conteúdo do seu site no Amazon S3.
- Clone a solução em seu computador para adicionar o conteúdo do site. Depois, implante a solução com a AWS Command Line Interface (AWS CLI).

i Note

Você deve usar a região Leste dos EUA (Norte da Virgínia) para implantar o modelo do CloudFormation.

Tópicos

- [Pré-requisitos](#)
- [Usar o console do AWS CloudFormation](#)
- [Clonar a solução localmente](#)
- [Encontrar logs de acesso](#)

Pré-requisitos

Para usar essa solução, são necessários os seguintes pré-requisitos:

- Um nome de domínio registrado, como `example.com`, apontado para uma zona hospedada do Amazon Route 53. A zona hospedada deverá estar na mesma Conta da AWS na qual você a solução é implantada. Quando não houver um nome de domínio registrado, é possível [registrar um com o Route 53](#). Quando houver um nome de domínio registrado, mas que não apontado para uma zona hospedada do Route 53, [configure o Route 53 como seu serviço DNS](#).
- Permissões do AWS Identity and Access Management (IAM) para executar modelos do CloudFormation que criam funções do IAM e permissões para criar todos os recursos da AWS na solução. Para obter mais informações, consulte [Controlling access with AWS Identity and Access Management](#) no Manual do usuário do AWS CloudFormation.

Você é responsável pelos custos incorridos durante a utilização da solução. Para ter mais informações sobre custos, consulte [as páginas de preços para cada AWS service \(Serviço da AWS\)](#).

Usar o console do AWS CloudFormation

Como implantar usando o console do CloudFormation

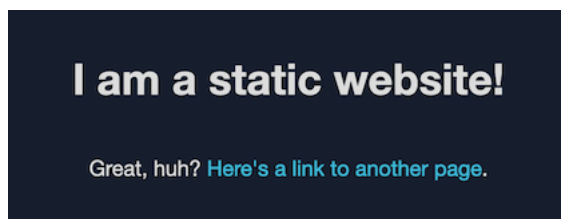
1. [Inicie esta solução no console do AWS CloudFormation](#). Se necessário, faça login na sua Conta da AWS.
2. O assistente Create Stack (Criar pilha) será aberto no console do CloudFormation, com campos pré-preenchidos que especificam o modelo do CloudFormation dessa solução.

Na parte inferior da página, selecione Próximo.

3. Na página Especificar detalhes da pilha, insira valores para os seguintes campos:
 - SubDomain (Subdomínio): insira o subdomínio a ser usado para o site. Por exemplo, se o subdomínio for `www`, o site estará disponível em `www.example.com`. (Substitua `exemplo.com` pelo seu nome de domínio, conforme explicado no marcador a seguir.)

- **DomainName**: insira o nome do domínio, como *example.com*. Esse domínio deve estar apontado para uma zona hospedada do Route 53.
 - **HostedZoneID**: o ID da zona hospedada no Route 53 no seu nome de domínio.
 - **CreateApex**: (opcional) crie um alias para o apex do domínio (exemplo.com) na configuração do CloudFront.
4. Quando terminar, escolha Next (Próximo).
 5. (Opcional) Na página Configure stack options (Configurar opções de pilha), [adicione tags e outras opções de pilha](#).
 6. Quando terminar, escolha Next (Próximo).
 7. Na página Revisar, role até a parte inferior da página e marque as duas caixas na seção Recursos. Esses recursos permitem que o CloudFormation crie um perfil do IAM que permita acesso aos recursos da pilha e nomeie-os dinamicamente.
 8. Selecione Create stack.
 9. Aguarde até que a criação da pilha seja concluída. A pilha criará algumas pilhas aninhadas e poderá levar vários minutos ser concluída. Após a conclusão, o Status mudará para CREATE_COMPLETE.

Quando o status for CREATE_COMPLETE, acesse <https://www.exemplo.com> para visualizar seu site (substitua www.exemplo.com pelo subdomínio e nome de domínio especificado na etapa 3). Será exibido o conteúdo padrão do site:



Como substituir o conteúdo padrão do site por seu próprio conteúdo

1. Abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. Escolha o bucket que tenha o nome que comece com amazon-cloudfront-secure-static-site-s3bucketroot-.

Note

Certifique-se de escolher o bucket com s3bucketroot no nome, não s3bucketlogs. O bucket com s3bucketroot no nome contém o conteúdo do site. Aquele com s3bucketlogs contém somente arquivos de log.

3. Exclua o conteúdo padrão do site e carregue o seu.

Note

Se você visualizou seu site com o conteúdo padrão da solução, é provável que parte do conteúdo padrão esteja armazenada em cache em um ponto de presença do CloudFront. Para garantir que os visualizadores vejam o conteúdo atualizado do site, invalide os arquivos para remover as cópias armazenadas em cache dos pontos de presença do CloudFront. Para ter mais informações, consulte [Invalidar arquivos para remover conteúdo](#).

Clonar a solução localmente

Pré-requisitos

Para adicionar o conteúdo do site antes de implantar essa solução, é necessário empacotar os artefatos da solução localmente, o que requer Node.js e npm. Para obter mais informações, consulte <https://www.npmjs.com/get-npm>.

Como adicionar o conteúdo do site e implantar a solução

1. Clone ou faça download da solução em <https://github.com/aws-samples/amazon-cloudfront-secure-static-site>. Depois que clonar ou fazer download, abra um prompt de comando ou terminal e navegue até a pasta `amazon-cloudfront-secure-static-site`.
2. Execute o seguinte comando para instalar e empacotar os artefatos da solução:

```
make package-static
```

3. Copie o conteúdo do site na pasta `www`, substituindo o conteúdo padrão do site.

4. Execute o seguinte comando da AWS CLI para criar um bucket do Amazon S3 para armazenar os artefatos da solução. Substitua *amzn-s3-demo-bucket-for-artifacts* pelo nome do bucket.

```
aws s3 mb s3://amzn-s3-demo-bucket-for-artifacts --region us-east-1
```

5. Execute o comando da AWS CLI a seguir a fim de empacotar os artefatos da solução como um modelo do CloudFormation. Substitua *amzn-s3-demo-bucket-for-artifacts* pelo nome do bucket que você criou na etapa anterior.

```
aws cloudformation package \  
  --region us-east-1 \  
  --template-file templates/main.yaml \  
  --s3-bucket amzn-s3-demo-bucket-for-artifacts \  
  --output-template-file packaged.template
```

6. Execute o seguinte comando para implantar a solução com o CloudFormation, substituindo os seguintes valores:

- *your-CloudFormation-stack-name*: substitua por um nome para a pilha do CloudFormation.
- *example.com*: substitua pelo nome de seu domínio. Esse domínio deve estar apontado para uma zona hospedada do Route 53 na mesma Conta da AWS.
- *www*: substitua pelo subdomínio a ser usado para o site. Por exemplo, se o subdomínio for *www*, o site estará disponível em *www.example.com*.
- *hosted-zone-ID*: substitua pelo ID da zona hospedada do Route 53 do nome de domínio.

```
aws cloudformation deploy \  
  --region us-east-1 \  
  --stack-name your-CloudFormation-stack-name \  
  --template-file packaged.template \  
  --capabilities CAPABILITY_NAMED_IAM CAPABILITY_AUTO_EXPAND \  
  --parameter-overrides DomainName=example.com SubDomain=www HostedZoneId=hosted-  
zone-ID
```

- (Opcional) Para implantar a pilha com um apex de domínio, execute o comando a seguir.

```
aws --region us-east-1 cloudformation deploy \  

```

```
--stack-name your-CloudFormation-stack-name \  
--template-file packaged.template \  
--capabilities CAPABILITY_NAMED_IAM CAPABILITY_AUTO_EXPAND \  
--parameter-overrides DomainName=example.com SubDomain=www  
HostedZoneId=hosted-zone-ID CreateApex=yes
```

7. Aguarde até que a pilha do CloudFormation conclua a criação. A pilha criará algumas pilhas aninhadas e poderá levar vários minutos ser concluída. Após a conclusão, o Status mudará para CREATE_COMPLETE.

Quando o status for alterado para CREATE_COMPLETE, acesse <https://www.exemplo.com> para visualizar o site (substitua www.exemplo.com pelo subdomínio e nome de domínio especificado na etapa anterior). Será exibido o conteúdo do site.

Encontrar logs de acesso

Essa solução habilita os [logs de acesso](#) para a distribuição do CloudFront. Conclua as etapas a seguir para localizar os logs de acesso da distribuição.

Como localizar os logs de acesso da distribuição

1. Abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. Escolha o bucket que tenha o nome que comece com amazon-cloudfront-secure-static-site-s3bucketlogs-.

Note

Certifique-se de escolher o bucket com s3bucketlogs no nome, não s3bucketroot. O bucket com s3bucketlogs no nome contém arquivos de log. Aquele com s3bucketroot contém o conteúdo do site.

3. A pasta chamada cdn contém os logs de acesso do CloudFront.

Configurar distribuições

Você deve criar uma distribuição do Amazon CloudFront para informar ao CloudFront de onde você deseja que o conteúdo seja entregue e os detalhes sobre como rastrear e gerenciar a entrega do conteúdo.

Escolha entre as seguintes opções de configuração:

- **A origem do conteúdo:** o bucket do Amazon S3, o canal do AWS Elemental MediaPackage, o contêiner do AWS Elemental MediaStore, o balanceador de carga do Elastic Load Balancing ou o servidor HTTP, do qual o CloudFront obtém os arquivos para distribuição. Você pode especificar qualquer combinação de até 25 origens para uma única distribuição.
- **Acesso:** se você deseja que os arquivos fiquem disponíveis para todos ou se preferir restringir o acesso a alguns usuários.
- **Segurança:** se você deseja habilitar a proteção do AWS WAF e exigir que os usuários usem HTTPS para acessar o conteúdo.
- **Chave de cache:** quais possíveis valores você deseja incluir na chave de cache. A chave de cache identifica exclusivamente cada arquivo no cache para uma determinada distribuição.
- **Configurações de solicitação de origem:** se desejar que o CloudFront inclua cabeçalhos HTTP, cookies ou strings de consulta em solicitações enviadas à origem.
- **Restrições geográficas:** se você deseja que o CloudFront impeça o acesso de usuários ao conteúdo em determinados países.
- **Logs:** se você deseja que o CloudFront crie logs padrão ou logs em tempo real que mostrem a atividade do visualizador.

Para ter mais informações, consulte [Referência de configurações da distribuição](#).

Para saber o número máximo atual de distribuições que você pode criar para cada conta da AWS, consulte [Cotas gerais para distribuições](#). Não há número máximo de arquivos que você pode veicular por distribuição.

Você pode usar distribuições para fornecer o seguinte por HTTP ou HTTPS:

- Conteúdo de download estático e dinâmico, como arquivos HTML, CSS, JavaScript e arquivos de imagem, usando HTTP ou HTTPS.

- Vídeos sob demanda em diferentes formatos, como o Apple HTTP Live Streaming (HLS) e o Microsoft Smooth Streaming. Para ter mais informações, consulte [Fornecer vídeo sob demanda com o CloudFront](#).
- Um evento ao vivo, como uma reunião, congresso ou concerto, em tempo real. Para streaming ao vivo, você cria a distribuição automaticamente usando uma pilha do AWS CloudFormation. Para ter mais informações, consulte [Fornecer streaming de vídeo ao vivo com o CloudFront e o AWS Media Services](#).

Os tópicos a seguir fornecem mais detalhes sobre as distribuições do CloudFront e como configurá-las para atender às suas necessidades empresariais. Para obter informações sobre como criar uma distribuição, consulte [Criar uma distribuição](#).

Tópicos

- [Criar uma distribuição](#)
- [Referência de configurações da distribuição](#)
- [Testar uma distribuição](#)
- [Atualizar uma distribuição](#)
- [Marcar uma distribuição](#)
- [Excluir uma distribuição do](#)
- [Usar a implantação contínua do CloudFront para testar com segurança as alterações na configuração da CDN](#)
- [Usar várias origens com distribuições do CloudFront](#)
- [Usar URLs personalizados adicionando nomes de domínio alternativos \(CNAMEs\)](#)
- [Usar WebSockets com distribuições do CloudFront](#)

Criar uma distribuição


O procedimento a seguir explica como usar o console do CloudFront para criar uma distribuição.

Visão geral da criação de uma distribuição

1. Crie um ou mais buckets do Amazon S3 ou configure servidores HTTP como seus servidores de origem. Origem é o local de armazenamento da versão original do conteúdo. Quando o CloudFront recebe uma solicitação dos seus arquivos, ele acessa a origem para obter os

arquivos que distribui nos pontos de presença. É possível usar qualquer combinação de buckets do Amazon S3 e servidores HTTP como seus servidores de origem.

- Se você estiver usando o Amazon S3, o nome do bucket deverá ser composto por letras minúsculas e não poderá conter espaços.
 - Se você usar um servidor do Amazon EC2 ou outra origem personalizada, consulte [Usar o Amazon EC2 \(ou outra origem personalizada\)](#).
 - Para obter o número máximo atual de origens que você pode criar para uma distribuição ou para solicitar uma cota maior, consulte [Cotas gerais para distribuições](#).
2. Faça upload do conteúdo nos seus servidores de origem. É possível tornar os objetos legíveis pelo público em geral ou usar URLs assinados do CloudFront para restringir o acesso ao conteúdo.

 Important

Você é responsável por garantir a segurança do seu servidor de origem. É necessário garantir que o CloudFront tenha permissão para acessar o servidor e que as configurações de segurança protejam o conteúdo.

3. Crie sua distribuição do CloudFront:
 - Para ler um procedimento detalhado que cria uma distribuição no console do CloudFront, consulte [Criar uma distribuição](#).
 - Para ter informações sobre como criar uma distribuição usando a API do CloudFront, consulte [CreateDistribution](#) na Referência da API do Amazon CloudFront.
4. (Opcional) Se você usar o console do CloudFront para criar a distribuição, crie mais comportamentos de cache ou origens para ela. Para obter mais informações sobre comportamentos e origens, consulte [Para atualizar uma distribuição do CloudFront](#).
5. Teste sua distribuição. Para ter mais informações sobre armazenamento, consulte [Testar uma distribuição](#).
6. Desenvolva seu site ou aplicação para acessar seu conteúdo usando o nome de domínio retornado pelo CloudFront depois de criar a distribuição na Etapa 3. Por exemplo, se o CloudFront exibir `d111111abcdef8.cloudfront.net` como o nome de domínio para a distribuição, o URL do arquivo `image.jpg` em um bucket do Amazon S3 ou no diretório raiz em um servidor HTTP será `https://d111111abcdef8.cloudfront.net/image.jpg`.

Se você especificou um ou mais nomes de domínio alternativos (CNAMEs) ao criar a distribuição, poderá usar seu próprio nome de domínio. Nesse caso, o URL de `image.jpg` pode ser `https://www.example.com/image.jpg`.

Observe o seguinte:

- Se você quiser usar signed URLs para restringir o acesso ao seu conteúdo, consulte [Veicular conteúdo privado com URLs e cookies assinados](#).
- Se você quiser fornecer conteúdo compactado, consulte [Fornecer arquivos compactados](#).
- Para obter informações sobre o comportamento de solicitação e resposta do CloudFront para o Amazon S3 e origens personalizadas, consulte [Comportamento de solicitações e respostas](#).

Tópicos

- [Criar uma distribuição do CloudFront no console](#)
- [Valores que o CloudFront exibe no console](#)
- [Links adicionais](#)

Criar uma distribuição do CloudFront no console

Para criar uma distribuição (console)

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Distribuições e Criar distribuição.
3. Especifique as configurações da distribuição. Para ter mais informações, consulte [Referência de configurações da distribuição](#).
4. Salve as alterações.
5. Depois que o CloudFront criar a sua distribuição, o valor da coluna Status da sua distribuição será alterado de Deploying (Implantando) para a hora e a hora em que a distribuição foi implantada. Se você optar por habilitar a distribuição, ela estará pronta para processar solicitações nesse momento.

O nome de domínio que o CloudFront atribuir à sua distribuição será exibido na lista de distribuições. (Ele também é exibido na guia Geral de uma distribuição selecionada.)

i Tip

É possível usar um nome de domínio alternativo, em vez do nome atribuído a você pelo CloudFront, seguindo as etapas em [Usar URLs personalizados adicionando nomes de domínio alternativos \(CNAMEs\)](#).

6. Quando a distribuição for implantada, confirme se é possível acessar seu conteúdo usando o novo URL do CloudFront ou o CNAME. Para ter mais informações, consulte [Testar uma distribuição](#).

Valores que o CloudFront exibe no console

Ao criar uma distribuição ou atualizar uma distribuição existente, o CloudFront exibe as informações abaixo no console do CloudFront.

i Note

Os assinantes confiáveis ativos, as contas da AWS com um par de chaves ativo do CloudFront e que podem ser usadas para criar signed URLs válidos, não estão visíveis no console do CloudFront no momento.

Distribution ID (ID de distribuição)

Ao executar uma ação em uma distribuição usando a API do CloudFront, use o ID de distribuição para especificar qual distribuição será usada, por exemplo, EDFDVBD6EXAMPLE. Você não pode alterar o ID de distribuição de uma distribuição.

Implantação e status

Enquanto a distribuição estiver sendo implantada, o status Deploying (Implantando) na coluna Last modified (Última modificação) é exibido. Aguarde até que a distribuição termine a implantação e certifique-se de que a coluna Status mostre Enabled (Habilitado). Para ter mais informações, consulte [Estado de distribuição](#).

Last modified (Última modificação)

A data e a hora em que a distribuição foi modificada pela última vez, usando o formato ISO 8601, por exemplo, 2012-05-19T19:37:58Z. Para ter mais informações, consulte <https://www.w3.org/TR/NOTE-datetime>.

Nome de domínio

Você usa o nome de domínio da distribuição nos links dos seus objetos. Por exemplo, se o nome do domínio da sua distribuição for `d111111abcdef8.cloudfront.net`, o link de `/images/image.jpg` será `https://d111111abcdef8.cloudfront.net/images/image.jpg`. Não é possível alterar o nome de domínio do CloudFront da sua distribuição. Para mais informações sobre URLs do CloudFront de links dos seus objetos, consulte [Personalizar o formato do URL para arquivos no CloudFront](#).

Se você especificar um ou mais nomes de domínio alternativos (CNAMEs), poderá usar seus próprios nomes de domínio nos links para seus objetos, em vez de usar o nome de domínio do CloudFront. Para obter mais informações sobre os CNAMEs, consulte [Nomes de domínio alternativos \(CNAMEs\)](#).

Note

Os nomes de domínio do CloudFront são exclusivos. O nome de domínio da sua distribuição nunca foi usado por uma distribuição anterior e não será reutilizado por outra distribuição no futuro.

Links adicionais

Para ter mais informações sobre como criar uma distribuição, consulte os links a seguir.

- Para saber como criar uma distribuição que usa uma origem de bucket do Amazon Simple Storage Service (Amazon S3) com controle de acesso à origem (OAC), consulte [Conceitos básicos de uma distribuição básica do CloudFront](#).
- Para ter informações sobre como usar APIs do CloudFront para criar uma distribuição, consulte [CreateDistribution](#) na Referência da API do Amazon CloudFront.
- Consulte informações sobre como atualizar uma distribuição (por exemplo, adicionar ou alterar comportamentos de cache) em [Atualizar uma distribuição](#).

- Para ver o número máximo atual de distribuições que você pode criar para cada conta da AWS ou para solicitar uma cota maior (anteriormente conhecida como limite), consulte [Cotas gerais para distribuições](#).

Referência de configurações da distribuição

Ao usar o [console do CloudFront](#) para criar uma distribuição ou atualizar uma existente, especifique os valores a seguir.

Para mais informações sobre como criar ou atualizar uma distribuição usando o console do CloudFront, consulte [the section called “Criar uma distribuição”](#) ou [the section called “Atualizar uma distribuição”](#).

Tópicos

- [Configurações de origem](#)
- [Configurações de comportamento de cache](#)
- [Configurações de distribuição](#)
- [Páginas de erro personalizadas e erro de armazenamento em cache](#)
- [Restrições geográficas](#)

Configurações de origem

Ao usar o console do CloudFront para criar ou atualizar uma distribuição, você deve fornecer informações sobre um ou mais locais, conhecidos como origens, onde são armazenadas as versões originais de conteúdo da web. O CloudFront obtém seu conteúdo da Web de suas origens e fornece-o aos visualizadores por meio de uma rede mundial de servidores de borda.

Para obter o número máximo atual de origens que você pode criar para uma distribuição ou para solicitar uma cota maior, consulte [the section called “Cotas gerais para distribuições”](#).

Se você quiser excluir uma origem, primeiro, deve editar ou excluir os comportamentos de cache associados a ela.

Important

Se você excluir uma origem, confirme se os arquivos fornecidos anteriormente por ela estão disponíveis em outra origem e se os seus comportamentos de cache estão roteando as solicitações desses arquivos para a nova origem.

Ao criar ou atualizar uma distribuição, você especifica os valores a seguir para cada origem.

Tópicos

- [Domínio de origem](#)
- [Protocolo \(somente origens personalizadas\)](#)
- [Caminho de origem](#)
- [Nome](#)
- [Acesso à origem \(somente origens do Amazon S3\)](#)
- [Adicionar cabeçalho personalizado](#)
- [Habilitar o Origin Shield](#)
- [Tentativas de conexão](#)
- [Tempo limite da conexão](#)
- [Tempo limite de resposta \(somente origens personalizadas\)](#)
- [Tempo limite keep alive \(somente origens personalizadas\)](#)
- [Cotas de tempo limite de resposta e manutenção](#)

Domínio de origem

O domínio de origem é o nome de domínio do DNS do recurso do qual o CloudFront receberá os objetos de sua origem, como um bucket do Amazon S3 ou um servidor HTTP: Por exemplo:

- Bucket do Amazon S3 – *amzn-s3-demo-bucket.s3.us-west-2.amazonaws.com*

Note

Se você criou o bucket do S3 recentemente, a distribuição do CloudFront poderá retornar respostas HTTP 307 Temporary Redirect por até 24 horas. Pode demorar até 24

horas para que o nome do bucket do S3 seja propagado para todas as regiões da AWS. Quando a propagação estiver concluída, a distribuição interromperá automaticamente o envio dessas respostas de redirecionamento sem exigir nenhuma ação. Para obter mais informações, consulte [Por que estou obtendo uma resposta de redirecionamento temporário HTTP 307 do Amazon S3?](#) e [Redirecionamento de solicitação temporário](#).

- Bucket do Amazon S3 configurado como site – *amzn-s3-demo-bucket.s3-website.us-west-2.amazonaws.com*
- Contêiner do MediaStore – *examplemediastore.data.mediastore.us-west-1.amazonaws.com*
- Endpoint do MediaPackage – *examplemediapackage.mediapackage.us-west-1.amazonaws.com*
- Instância do Amazon EC2 – *ec2-203-0-113-25.compute-1.amazonaws.com*
- Balanceador de carga do Elastic Load Balancing – *example-load-balancer-1234567890.us-west-2.elb.amazonaws.com*
- Seu próprio servidor Web: *www.example.com*

Escolha o nome do domínio no campo Origin domain (Domínio de origem) ou digite o nome. Os recursos das regiões que requerem adesão devem ser inseridos manualmente. O nome de domínio não diferencia maiúsculas de minúsculas. Seu domínio de origem deve ter um nome de DNS publicamente resolvível que direcione as solicitações dos clientes aos destinos pela internet.

Se você configurar o CloudFront para se conectar à sua origem por HTTPS, um dos nomes de domínio no certificado deverá corresponder ao nome de domínio especificado em Nome de domínio de origem. Se nenhum nome de domínio apresentar correspondência, o CloudFront exibirá um código de status HTTP 502 (gateway inválido) para o visualizador. Para ter mais informações, consulte [Nomes de domínio na distribuição do CloudFront e no certificado](#) e [Falha de negociação SSL/TLS entre o CloudFront e um servidor de origem personalizado](#).

Note

Se você estiver usando uma política de solicitação de origem que encaminhe o cabeçalho de host do visualizador à origem, esta deverá responder com um certificado correspondente ao cabeçalho de host do visualizador. Para ter mais informações, consulte [Adicionar cabeçalhos de solicitação do CloudFront](#).

Se sua origem for um bucket do Amazon S3, observe o seguinte:

- Se o bucket estiver configurado como um site, insira o endpoint de hospedagem do site estático do Amazon S3 no bucket. Não selecione o nome do bucket na lista do campo Origin domain (Domínio de origem). O endpoint de hospedagem do site estático é exibido no console do Amazon S3, na página Properties (Propriedades) em Static website hosting (Hospedagem de site estático). Para ter mais informações, consulte [the section called “Usar um bucket do Amazon S3 configurado como um endpoint do site”](#).
- Se você configurou o Amazon S3 Transfer Acceleration para seu bucket, não especifique o endpoint `s3-accelerate` para Origin domain (Domínio de origem).
- Se você estiver usando um bucket de uma conta da AWS diferente e ele não estiver configurado como um site, digite o nome no seguinte formato:

bucket-name.s3.*region*.amazonaws.com

Se o bucket estiver em uma região dos EUA e você quiser que o Amazon S3 direcione as solicitações para uma instalação no Norte da Virgínia, use o seguinte formato:

bucket-name.s3.us-east-1.amazonaws.com

- Os arquivos devem ser legíveis ao público, a menos que você proteja o conteúdo no Amazon S3 usando um controle de acesso à origem do CloudFront. Para obter mais informações sobre controle do acesso, consulte [the section called “Restringir o acesso a uma origem do Amazon Simple Storage Service”](#).

Important

Se a origem for um bucket do Amazon S3, o nome do bucket deverá seguir os requisitos de nomenclatura do DNS. Para obter mais informações, acesse [Restrições e limitações de bucket](#) no Guia do usuário do Amazon Simple Storage Service.

Ao alterar o valor de Origin domain (Domínio de origem) de uma origem, o CloudFront começará imediatamente a replicar a alteração para seus locais de borda. Enquanto a configuração de distribuição não for atualizada em determinado local de borda, o CloudFront continuará a encaminhar solicitações à origem anterior. Assim que a configuração da distribuição for atualizada no local de borda, o CloudFront começará a encaminhar solicitações à nova origem.

A alteração da origem não requer que o CloudFront preencha os locais de borda de caches novamente com objetos da nova origem. Contudo que as solicitações do visualizador em sua aplicação não tenham sido alteradas, o CloudFront continuará a fornecer objetos que já estão em um ponto de presença de caches até o TTL de cada objeto expirar ou até objetos solicitados raramente serem removidos.

Protocolo (somente origens personalizadas)

Note

Isto se aplica apenas a origens personalizadas.

A política de protocolo a ser usada pelo CloudFront ao obter objetos de sua origem.

Escolha um dos seguintes valores:

- HTTP only (Somente HTTP): o CloudFront usa somente HTTP para acessar a origem.

Important

HTTP only (Somente HTTP) é a configuração padrão quando a origem é um endpoint de hospedagem de site estático do Amazon S3, porque o Amazon S3 não comporta conexões HTTPS para endpoint de hospedagem de site estático. O console do CloudFront não oferece suporte à alteração dessa configuração para endpoints de hospedagem de sites estáticos do Amazon S3.

- HTTPS only (Somente HTTPS): o CloudFront usa somente HTTPS para acessar a origem.
- Match viewer (Corresponder visualizador): o CloudFront se comunica com sua origem usando HTTP ou HTTPS, dependendo do protocolo da solicitação do visualizador. Observe que o CloudFront armazenará o objeto em cache somente uma vez se os visualizadores fizerem solicitações usando protocolos HTTP e HTTPS.

Important

Para solicitações HTTPS do visualizador que o CloudFront encaminha para essa origem, um dos nomes de domínio no certificado SSL/TLS de seu servidor de origem deverá corresponder ao nome de domínio especificado para Origin domain (Domínio de origem). Do contrário, o CloudFront responderá às solicitações do visualizador com um código de

status HTTP 502 (Gateway inválido), em vez de retornar o objeto solicitado. Para ter mais informações, consulte [the section called “Requisitos para usar certificados SSL/TLS com o CloudFront”](#).

Tópicos

- [Porta HTTP](#)
- [Porta HTTPS](#)
- [Protocolo SSL de origem mínimo](#)

Porta HTTP

Note

Isto se aplica apenas a origens personalizadas.

(Opcional) É possível especificar a porta HTTP na qual a origem personalizada escuta. Os valores válidos incluem as portas 80, 443 e 1024 a 65535. O valor padrão é a porta 80.

Important

A porta 80 é a configuração padrão quando a origem é um endpoint de hospedagem de site estático do Amazon S3, porque ele só oferece suporte à porta 80 para endpoints de hospedagem de site estático. O console do CloudFront não oferece suporte à alteração dessa configuração para endpoints de hospedagem de sites estáticos do Amazon S3.

Porta HTTPS

Note

Isto se aplica apenas a origens personalizadas.

(Opcional) É possível especificar a porta HTTPS na qual a origem personalizada escuta. Os valores válidos incluem as portas 80, 443 e 1024 a 65535. O valor padrão é a porta 443. Quando Protocol

(Protocolo) é definido como HTTP only (Somente HTTP), não é possível especificar um valor para HTTPS port (Porta HTTPS).

Protocolo SSL de origem mínimo

Note

Isto se aplica apenas a origens personalizadas.

Escolha o protocolo TLS/SSL mínimo que o CloudFront poderá usar ao estabelecer uma conexão HTTPS com a origem. Protocolos TLS mais baixos são menos seguros, portanto, recomendamos que você escolha o protocolo TLS mais recente compatível com a origem. Quando Protocol (Protocolo) é definido como HTTP only (Somente HTTP), não é possível especificar um valor para Minimum origin SSL protocol (Protocolo SSL de origem mínimo).

Se você usar a API do CloudFront para definir o protocolo TLS/SSL para uso do CloudFront, não será possível definir um protocolo mínimo. Em vez disso, especifique todos os protocolos TLS/SSL que o CloudFront poderá usar com sua origem. Para mais informações, consulte [OriginSslProtocols](#) na Referência da API do Amazon CloudFront.

Caminho de origem

Se desejar que o CloudFront solicite seu conteúdo de um diretório em sua origem, insira o caminho do diretório, começando com uma barra (/). O CloudFront acrescenta o caminho do diretório no valor de Origin domain (Domínio de origem). Por exemplo, **cf-origin.example.com/production/images**. Não adicione barra (/) no fim do caminho.

Por exemplo, imagine que você especificou os seguintes valores para sua distribuição:

- Origin domain (Domínio de origem): um bucket do Amazon S3 denominado **amzn-s3-demo-bucket**
- Origin path (Caminho da origem): **/production**
- Nomes de domínio alternativos (CNAMEs): **example.com**

Quando um usuário insere `example.com/index.html` no navegador, o CloudFront envia uma solicitação de `amzn-s3-demo-bucket/production/index.html` para o Amazon S3.

Quando um usuário insere `example.com/acme/index.html` no navegador, o CloudFront envia uma solicitação de `amzn-s3-demo-bucket/production/acme/index.html` para o Amazon S3.

Nome

Nome é uma string que identifica exclusivamente essa origem nessa distribuição. Se criar comportamentos de cache além do comportamento de cache padrão, você poderá usar o nome especificado aqui para identificar a origem para a qual o CloudFront encaminhará uma solicitação quando ela tiver o mesmo padrão de caminho desse comportamento de cache.

Acesso à origem (somente origens do Amazon S3)

Note

Isso se aplica somente às origens de bucket do Amazon S3 (aquelas que não estão usando o endpoint de site estático do S3).

Selecione `Origin access control settings (recommended)` [Configurações de controle de acesso à origem (recomendado)] se você quiser tornar possível restringir o acesso a uma origem de bucket do Amazon S3 somente a distribuições específicas do CloudFront.

Selecione `Public (Pública)` se a origem do bucket do Amazon S3 for acessível ao público.

Para ter mais informações, consulte [the section called “Restringir o acesso a uma origem do Amazon Simple Storage Service”](#).

Para obter informações sobre como exigir que os usuários acessem objetos em uma origem personalizada usando somente URLs do CloudFront, consulte [the section called “Restringir o acesso a arquivos em origens personalizadas”](#).

Adicionar cabeçalho personalizado

Se você quiser que o CloudFront inclua cabeçalhos personalizados sempre que encaminhar uma solicitação para sua origem, especifique o nome do cabeçalho e o respectivo valor. Para ter mais informações, consulte [the section called “Adicionar cabeçalhos personalizados às solicitações de origem”](#).

Para saber o número máximo de cabeçalhos personalizados que você pode adicionar, o tamanho máximo do nome e valor de um cabeçalho personalizado e o tamanho total de todos os nomes e valores de cabeçalho, consulte [Cotas](#).

Habilitar o Origin Shield

Escolha Yes (Sim) para habilitar o CloudFront Origin Shield. Para obter mais informações sobre o Origin Shield, consulte [the section called “Usar o escudo de origem”](#).

Tentativas de conexão

É possível definir o número de vezes que o CloudFront tenta se conectar à origem. É possível especificar 1, 2 ou 3 como o número de tentativas. O número padrão (caso nenhum seja especificado) é 3.

Use essa configuração com o Connection timeout (Tempo limite da conexão) para especificar o período que o CloudFront deve esperar antes de tentar se conectar à origem secundária ou retornar uma resposta de erro ao visualizador. Por padrão, o CloudFront aguarda até 30 segundos (3 tentativas de 10 segundos cada) antes de tentar se conectar à origem secundária ou retornar uma resposta de erro. É possível reduzir esse tempo especificando menos tentativas, um tempo limite de conexão mais curto ou ambos.

Se houver falha de conexão no número de tentativas especificado, o CloudFront executará um dos seguintes procedimentos:

- Se a origem fizer parte de um grupo de origens, o CloudFront tentará se conectar à origem secundária. Se houver falha de conexão no número de tentativas especificado para a origem secundária, o CloudFront retornará uma resposta de erro ao visualizador.
- Se a origem não fizer parte de um grupo de origens, o CloudFront retornará uma resposta de erro ao visualizador.

Para uma origem personalizada (incluindo um bucket do Amazon S3 configurado com hospedagem de site estático), essa configuração também especifica o número de vezes que o CloudFront tenta obter uma resposta da origem. Para ter mais informações, consulte [the section called “Tempo limite de resposta \(somente origens personalizadas\)”](#).

Tempo limite da conexão

Tempo limite de conexão é o número de segundos que o CloudFront aguarda ao tentar estabelecer conexão com a origem. É possível especificar um número de segundos entre 1 e 10 (inclusive). O tempo limite padrão (caso nenhum seja especificado) é 10 segundos.

Use essa configuração com Connection attempts (Tentativas de conexão) para especificar o período que o CloudFront deve esperar antes de tentar se conectar à origem secundária ou antes de

retornar uma resposta de erro ao visualizador. Por padrão, o CloudFront aguarda até 30 segundos (3 tentativas de 10 segundos cada) antes de tentar se conectar à origem secundária ou retornar uma resposta de erro. É possível reduzir esse tempo especificando menos tentativas, um tempo limite de conexão mais curto ou ambos.

Se o CloudFront não estabelecer uma conexão com a origem dentro do número especificado de segundos, ele fará o seguinte:

- Se o número especificado de Connection attempts (Tentativas de conexão) for maior que 1, o CloudFront tentará estabelecer uma conexão novamente. O CloudFront tentará até três vezes, conforme determinado pelo valor de Connection attempts (Tentativas de conexão).
- Se houver falha em todas as tentativas de conexão e a origem fizer parte de um grupo de origens, o CloudFront tentará se conectar à origem secundária. Se houver falha de conexão no número de tentativas especificado para a origem secundária, o CloudFront retornará uma resposta de erro ao visualizador.
- Se houver falha em todas as tentativas de conexão e a origem não fizer parte de um grupo de origens, o CloudFront retornará uma resposta de erro ao visualizador.

Tempo limite de resposta (somente origens personalizadas)

O tempo limite da resposta de origem, também conhecido como tempo limite da leitura de origem ou tempo limite da solicitação de origem, aplica-se a estes dois valores:

- O período (em segundos) que o CloudFront aguarda uma resposta após o encaminhamento de uma solicitação à origem.
- O período (em segundos) que o CloudFront aguarda após o recebimento de um pacote de uma resposta da origem e antes do recebimento do próximo pacote.

Tip

Se você quiser aumentar o valor de tempo limite porque os visualizadores estão com erros de código de status HTTP 504, considere explorar outras maneiras de eliminar esses erros antes de alterar o valor de tempo limite. Consulte as sugestões para solução de problemas em [the section called “Código de status HTTP 504 \(tempo limite do gateway\)”](#).

O comportamento do CloudFront depende do método HTTP na solicitação de visualizador:

- Solicitações GET e HEAD: se a origem não responder ou parar de responder dentro da duração do tempo limite da resposta, o CloudFront interromperá a conexão. O CloudFront tentará se conectar novamente de acordo com o valor de [the section called “Tentativas de conexão”](#).
- Solicitações DELETE, OPTIONS, PATCH, PUT e POST: se a origem não responder dentro da duração do tempo limite da leitura, o CloudFront interromperá a conexão e não tentará entrar em contato com a origem novamente. O cliente pode reenviar a solicitação, se necessário.

Tempo limite keep alive (somente origens personalizadas)

O tempo limite de keep-alive é o período (em segundos) em que o CloudFront tentará manter uma conexão com a origem personalizada depois de obter o último pacote de uma resposta. A manutenção de uma conexão persistente economiza o tempo necessário para restabelecer a conexão TCP e executar outro handshake TLS para solicitações subsequentes. O aumento do tempo limite da conexão keep alive ajuda a melhorar a métrica da solicitação por conexão nas distribuições.

Note

Para que o valor Keep-alive timeout (Tempo limite keep alive) entre em vigor, a origem deverá ser configurada para permitir conexões persistentes.

Cotas de tempo limite de resposta e manutenção

Note

Isto se aplica apenas a origens personalizadas.

- O padrão do [tempo limite de resposta](#) é de 30 segundos.
- O padrão do [tempo limite de manutenção](#) é de 5 segundos.
- Para ambas as cotas, é possível especificar um valor de 1 a 60 segundos. Para solicitar um aumento, [crie um caso no AWS Support Center Console](#).

Depois de solicitar um aumento de tempo limite para a Conta da AWS, atualize as origens da distribuição para os valores de tempo limite de resposta e de manutenção que você deseja. Um aumento de cota para sua conta não atualiza automaticamente as origens. Por exemplo, se você

estiver usando uma função do Lambda@Edge para definir um tempo limite de manutenção de 90 segundos, a origem já deverá ter um tempo limite de manutenção de 90 segundos ou mais. Caso contrário, a função do Lambda@Edge poderá falhar na execução.

Consulte mais informações sobre cotas de distribuição em [Cotas gerais para distribuições](#).

Configurações de comportamento de cache

Ao definir o comportamento de cache, é possível configurar uma variedade de funcionalidades do CloudFront para um padrão de caminho de URL específico para arquivos no site. Por exemplo, um comportamento de cache pode se aplicar a todos os arquivos `.jpg` no diretório `images` em um servidor web sendo usado como servidor de origem do CloudFront. A funcionalidade que pode ser configurada para cada comportamento de cache inclui:

- O padrão de caminho.
- Se você tiver configurado várias origens para a distribuição do CloudFront, a origem para a qual você deseja que o CloudFront encaminhe suas solicitações.
- Se deve encaminhar strings de consulta para a origem.
- Se o acesso a arquivos específicos requer URLs assinados.
- Se deve exigir que os usuários usem HTTPS para acessar esses arquivos.
- O tempo mínimo de permanência desses arquivos no cache do CloudFront, independentemente do valor de qualquer cabeçalho `Cache-Control` adicionado por sua origem aos arquivos.

Ao criar uma distribuição, você especifica as configurações do comportamento de cache padrão, que encaminha automaticamente todas as solicitações para a origem especificada ao criar a distribuição. Depois que criar uma distribuição, você poderá criar outros comportamentos de cache que definem como o CloudFront responde ao receber uma solicitação de objetos que correspondem a um padrão de caminho, por exemplo, `*.jpg`. Se você criar mais comportamentos de cache, o comportamento de cache padrão será sempre o último a ser processado. Outros comportamentos de cache são processados na ordem indicada no console do CloudFront ou, se você estiver usando a API do CloudFront, na ordem indicada no elemento `DistributionConfig` da distribuição. Para ter mais informações, consulte [Padrão de caminho](#).

Ao criar um comportamento de cache, especifique a origem da qual você deseja que o CloudFront obtenha objetos. Como resultado, se quiser que o CloudFront distribua objetos de todas as suas origens, você deverá ter, pelo menos, a mesma quantidade de comportamentos de cache (inclusive

o comportamento de cache padrão) que o número de origens. Por exemplo, se você tiver duas origens e somente o comportamento de cache padrão, o comportamento de cache padrão fará com que o CloudFront obtenha objetos de uma das origens, mas a outra origem nunca será usada.

Para obter o número máximo atual de comportamentos de cache que podem ser adicionados a uma distribuição ou solicitar uma cota maior (anteriormente conhecida como limite), consulte [Cotas gerais para distribuições](#).

Tópicos

- [Padrão de caminho](#)
- [Origem ou grupo de origem](#)
- [Política de protocolo do visualizador](#)
- [Métodos HTTP permitidos](#)
- [Configuração da criptografia em nível de campo](#)
- [Métodos HTTP em cache](#)
- [Cache baseado em Cabeçalhos de solicitação selecionados](#)
- [Lista de permissões de cabeçalhos](#)
- [Armazenamento de objetos em cache](#)
- [Minimum TTL](#)
- [Maximum TTL](#)
- [TTL padrão](#)
- [Cookies progressivos](#)
- [Lista de permissões de cookies](#)
- [Encaminhamento e armazenamento em cache de strings de consulta](#)
- [Lista de permissões de strings de consulta](#)
- [Smooth Streaming](#)
- [Restringir acesso do visualizador \(usar URLs ou cookies assinados\)](#)
- [Signatários confiáveis](#)
- [Números de Conta da AWS](#)
- [Compactar objetos automaticamente](#)
- [Evento do CloudFront](#)

- [ARN da função do Lambda.](#)
- [Incluir corpo](#)

Padrão de caminho

Um padrão de caminho (por exemplo, `images/* .jpg`) especifica a quais solicitações aplicar o comportamento de cache. Quando o CloudFront recebe uma solicitação do usuário final, o caminho solicitado é comparado com os padrões de caminho na ordem em que os comportamentos de cache estão listados na distribuição. A primeira correspondência determina qual comportamento de cache é aplicado a essa solicitação. Por exemplo, imagine que você tenha três comportamentos de cache com os seguintes padrões de caminho, na ordem:

- `images/* .jpg`
- `images/*`
- `*.gif`

Note

Você também pode incluir uma barra (`/`) no início do padrão de caminho, por exemplo, `/images/* .jpg`. O comportamento do CloudFront é o mesmo com ou sem a `/` inicial. Se `/` não for especificado no início do caminho, esse caractere estará automaticamente implícito. O CloudFront tratará o caminho da mesma forma com ou sem `/` inicial. Por exemplo, o CloudFront trata `/*product .jpg` da mesma forma que `*product .jpg`

Uma solicitação para o arquivo `images/sample.gif` não satisfaz o primeiro padrão de caminho. Portanto, os comportamentos de cache associados não são aplicados à solicitação. O arquivo satisfaz o segundo padrão de caminho, portanto, os comportamentos de cache associados a ele são aplicados, embora a solicitação também corresponda ao terceiro padrão de caminho.

Note

Ao criar uma distribuição, o valor de Path Pattern para o comportamento de cache padrão é definido como `*` (todos os arquivos) e não pode ser alterado. Esse valor faz com que o CloudFront encaminhe todas as solicitações de objetos para a origem especificada no campo [Domínio de origem](#). Se a solicitação de um objeto não corresponder ao padrão de caminho

dos outros comportamentos de cache, o CloudFront aplicará o comportamento especificado no comportamento de cache padrão.

Important

Defina os padrões de caminho e a sequência deles cuidadosamente, caso contrário, você poderá conceder aos usuários acesso indesejado ao seu conteúdo. Por exemplo, imagine que uma solicitação corresponda ao padrão de caminho de dois comportamentos de cache. O primeiro comportamento de cache não requer signed URLs, mas o segundo comportamento de cache, sim. Os usuários poderão acessar os objetos sem usar um signed URL porque o CloudFront processa o comportamento de cache associado à primeira correspondência.

Se você estiver trabalhando com um canal do MediaPackage, será necessário incluir padrões de caminho específicos para o comportamento de cache que você definiu para o tipo de endpoint de sua origem. Por exemplo, para um endpoint DASH, digite `*.mpd` em Path Pattern (Padrão de caminho). Para mais informações e instruções específicas, consulte [Veicular vídeo ao vivo formatado com o AWS Elemental MediaPackage](#).

O caminho especificado se aplica a solicitações de todos os arquivos no diretório especificado e nos subdiretórios abaixo dele. O CloudFront não considera strings de consulta ou cookies ao avaliar o padrão de caminho. Por exemplo, se um diretório `images` contém subdiretórios `product1` e `product2`, o padrão de caminho `images/*.jpg` se aplica a solicitações de qualquer arquivo `.jpg` nos diretórios `images`, `images/product1` e `images/product2`. Se você quiser aplicar um comportamento de cache nos arquivos do diretório `images/product1` diferente dos arquivos dos diretórios `images` e `images/product2`, crie um comportamento de cache separado para `images/product1` e mova-o para uma posição acima (anterior) do comportamento de cache do diretório `images`.

Você pode usar os seguintes caracteres curinga no padrão de caminho:

- `*` corresponde a 0 ou mais caracteres.
- `?` corresponde a exatamente 1 caractere.

Os seguintes exemplos mostram como funcionam os caracteres curinga:

Padrão de caminho	Os arquivos que correspondem ao padrão de caminho
*.jpg	Todos os arquivos .jpg
images/*.jpg	Todos os arquivos do diretório <code>images</code> e subdiretórios no diretório <code>images</code>
a*.jpg	<ul style="list-style-type: none"> Todos os arquivos .jpg cujo nome começa com <code>a</code>, por exemplo, <code>apple.jpg</code> e <code>appalachian_trail_2012_05_21.jpg</code>. Todos os arquivos .jpg cujo caminho começa com <code>a</code>, por exemplo, <code>abra/cadabra/magic.jpg</code>.
a??.jpg	Todos os arquivos .jpg cujo nome começa com <code>a</code> e é seguido por exatamente dois outros caracteres, por exemplo, <code>ant.jpg</code> e <code>abe.jpg</code> .
.doc	Todos os arquivos cuja extensão do nome de arquivo começa com <code>.doc</code> , por exemplo, os arquivos <code>.doc</code> , <code>.docx</code> e <code>.docm</code> . Você não pode usar o padrão de caminho <code>*.doc?</code> nesse caso, porque ele não se aplica a solicitações de arquivos <code>.doc</code> . O caractere curinga <code>?</code> substitui exatamente um caractere.

O tamanho máximo de um padrão de caminho é 255 caracteres. O valor pode conter espaços ou um destes caracteres:

- A-Z, a-z

Os padrões de caminho diferenciam letras maiúsculas de minúsculas, portanto, o padrão `*.jpg` não se aplica ao arquivo `LOGO.JPG`.

- 0-9
- `_ - . * $ / ~ " ' @ : +`

- `&`, passado e retornado como `&`;

Normalização de caminhos

O CloudFront normaliza os caminhos de URI consistentes [com a RFC 3986](#) e, em seguida, combina o caminho com o comportamento correto do cache. Depois que o comportamento do cache é correspondido, o CloudFront envia o caminho bruto do URI para a origem. Se não forem correspondentes, as solicitações serão correspondentes ao comportamento de cache padrão.

Alguns caracteres são normalizados e removidos do caminho, como várias barras (`//`) ou pontos (`.`). Isso pode alterar o URL que o CloudFront usa para corresponder ao comportamento de cache pretendido.

Example Exemplo

Você especifica os caminhos `/a/b*` e `/a*` para seu comportamento de cache.

- Um visualizador que envia o caminho `/a/b?c=1` corresponderá ao comportamento do cache `/a/b*`.
- Um visualizador que envia o caminho `/a/b/..?c=1` corresponderá ao comportamento do cache `/a*`.

Para contornar os caminhos que estão sendo normalizados, é possível atualizar os caminhos da solicitação ou o padrão do caminho para o comportamento do cache.

Origem ou grupo de origem

Essa configuração aplica-se somente ao criar ou atualizar o comportamento de cache de uma distribuição existente.

Informe o valor de uma origem ou um grupo de origem existente. Isso identifica a origem para a qual você deseja que o CloudFront encaminhe as solicitações quando uma solicitação (como `https://example.com/logo.jpg`) corresponder ao padrão de caminho de um comportamento de cache (como `*.jpg`) ou do comportamento de cache padrão (`*`).

Política de protocolo do visualizador

Escolha a política de protocolo que você deseja que os visualizadores usem para acessar seu conteúdo em pontos de presença do CloudFront.

- HTTP and HTTPS: os visualizadores pode usar os dois protocolos.
- Redirect HTTP to HTTPS: os visualizadores podem usar os dois protocolos, mas solicitações HTTP são automaticamente redirecionadas para HTTPS.
- HTTPS Only: os visualizadores só podem acessar seu conteúdo se estiverem usando HTTPS.

Para ter mais informações, consulte [Exigir HTTPS na comunicação entre visualizadores e CloudFront](#).

Métodos HTTP permitidos

Especifique os métodos HTTP a serem processados e encaminhados pelo CloudFront para sua origem:

- GET, HEAD: é possível usar o CloudFront somente para obter objetos da sua origem ou para obter cabeçalhos do objeto.
- GET, HEAD, OPTIONS:: é possível usar o CloudFront somente para obter objetos da sua origem, obter cabeçalhos do objeto ou recuperar uma lista das opções compatíveis com seu servidor de origem.
- GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE: é possível usar o CloudFront para obter, adicionar, atualizar e excluir objetos, e também para obter os cabeçalhos do objeto. Além disso, você pode executar outras operações de POST, como enviar dados de um formulário da web.

Note

O CloudFront armazena respostas a solicitações GET e HEAD e, opcionalmente, a solicitações OPTIONS. As respostas para as solicitações OPTIONS são armazenadas em cache separadamente das respostas para solicitações GET e HEAD (o método OPTIONS está incluído na [chave de cache](#) para solicitações OPTIONS). O CloudFront não armazena em cache respostas a solicitações que usam outros métodos.

Important

Se você escolher GET, HEAD, OPTIONS ou GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE, talvez seja necessário restringir o acesso ao seu bucket do Amazon S3 ou à sua origem personalizada para evitar que os usuários executem operações indesejadas. Os seguintes exemplos explicam como restringir o acesso:

- Se você estiver usando o Amazon S3 como origem da distribuição: crie uma identidade do acesso de origem do CloudFront para restringir o acesso ao conteúdo do Amazon S3 e conceda permissões ao controle de acesso à origem. Por exemplo, se você configurar o CloudFront para aceitar e encaminhar esses métodos somente porque deseja usar PUT, mesmo assim será necessário configurar políticas de bucket do Amazon S3 para lidar com solicitações DELETE de forma apropriada. Para ter mais informações, consulte [Restringir o acesso a uma origem do Amazon Simple Storage Service](#).
- Se você estiver usando uma origem personalizada: configure seu servidor de origem para lidar com todos os métodos. Por exemplo, se você configurar o CloudFront para aceitar e encaminhar esses métodos somente porque deseja usar POST, mesmo assim, é necessário configurar seu servidor de origem para lidar com solicitações DELETE forma apropriada.

Configuração da criptografia em nível de campo

Para impor a criptografia em nível de campo em campos de dados específicos, na lista suspensa, selecione uma configuração de criptografia em nível de campo.

Para ter mais informações, consulte [Use criptografia de nível de campo para ajudar a proteger dados confidenciais](#).

Métodos HTTP em cache

Especifique se você deseja que o CloudFront armazene em cache a resposta da sua origem quando um visualizador enviar uma solicitação OPTIONS. O CloudFront sempre armazena a resposta às solicitações GET e HEAD em cache.

Cache baseado em Cabeçalhos de solicitação selecionados

Especifique se deseja que o CloudFront armazene os objetos em cache com base nos valores dos cabeçalhos especificados:

- None (Nenhum; melhora cache): o CloudFront não armazena os objetos em cache com base nos valores do cabeçalho.
- Lista de permissões: o CloudFront armazena os objetos em cache com base somente nos valores dos cabeçalhos especificados. Use Lista de permissões de cabeçalhos para escolher os cabeçalhos nos quais deseja que o CloudFront baseie o armazenamento em cache.

- All (Todos): o CloudFront não armazena em cache os objetos associados a esse comportamento de cache. Em vez disso, o CloudFront envia todas as solicitações para a origem. (Não recomendado para origens do Amazon S3.)

Independentemente da opção escolhida, o CloudFront encaminhará determinados cabeçalhos para sua origem e realizará ações específicas com base nos cabeçalhos encaminhados por você. Para mais informações sobre como o CloudFront trata o encaminhamento de cabeçalhos, consulte [Cabeçalhos de solicitação HTTP e comportamento do CloudFront \(origens do Amazon S3 e personalizadas\)](#).

Para mais informações sobre como configurar o armazenamento em cache no CloudFront usando cabeçalhos de solicitação, consulte [Conteúdo em cache com base nos cabeçalhos de solicitação](#).

Lista de permissões de cabeçalhos

Essas configurações aplicam-se somente ao escolher Lista de permissões para Cache baseado nos cabeçalhos da solicitação selecionados.

Especifique os cabeçalhos a serem considerados pelo CloudFront ao armazenar seus objetos em cache. Selecione os cabeçalhos na lista de cabeçalhos disponíveis e escolha Add. Para encaminhar um cabeçalho personalizado, insira o nome dele no campo e escolha Add Custom.

Para saber o número máximo atual de cabeçalhos que podem ser incluídos na lista de permissões para cada comportamento de cache ou para solicitar uma cota maior (anteriormente conhecida como limite), consulte [Cotas para cabeçalhos](#).

Armazenamento de objetos em cache

Se o servidor de origem estiver adicionando um cabeçalho Cache-Control a seus objetos para controlar o tempo de permanência deles no cache do CloudFront e se você não quiser alterar o valor de Cache-Control, escolha Use Origin Cache Headers (Usar cabeçalhos de cache de origem).

Para especificar o tempo mínimo e máximo de permanência dos seus objetos no cache do CloudFront, independentemente dos cabeçalhos Cache-Control, e o tempo padrão de permanência dos seus objetos no cache do CloudFront quando o cabeçalho Cache-Control estiver ausente de um objeto, escolha Customize (Personalizar). Em seguida, especifique valores nos campos Minimum TTL (TTL mínimo), Default TTL (TTL padrão) e Maximum TTL (TTL máximo).

Para ter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Minimum TTL

Especifique o tempo mínimo, em segundos, que você deseja que os objetos permaneçam no cache do CloudFront antes do CloudFront enviar outra solicitação para a origem a fim de determinar se o objeto foi atualizado.

Para ter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Maximum TTL

Especifique o tempo máximo (em segundos) que você deseja que os objetos permaneçam em caches do CloudFront antes de ele consultar sua origem para ver se o objeto foi atualizado. O valor especificado para Maximum TTL é aplicado apenas quando sua origem adiciona cabeçalhos HTTP, como `Cache-Control max-age`, `Cache-Control s-maxage` ou `Expires`, aos objetos. Para ter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Para especificar um valor para Maximum TTL, você deve escolher a opção `Customize` para a configuração `Object Caching`.

O valor padrão de Maximum TTL é 31.536.000 segundos (um ano). Se você alterar o valor de Minimum TTL ou Default TTL para mais de 31.536.000 segundos, o valor padrão de Maximum TTL será alterado para o valor de Default TTL.

TTL padrão

O tempo padrão que você deseja que os objetos permaneçam em caches do CloudFront antes de ele encaminhar outra solicitação para sua origem para determinar se o objeto foi atualizado. O valor especificado para Default TTL (TTL padrão) aplica-se apenas quando sua origem não adiciona cabeçalhos de HTTP, como `Cache-Control max-age`, `Cache-Control s-maxage` ou `Expires`, aos objetos. Para ter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Para especificar um valor para Default TTL, você deve escolher a opção `Customize` para a configuração `Object Caching`.

O valor padrão de Default TTL é 86.400 segundos (um dia). Se você alterar o valor de Minimum TTL para mais de 86.400 segundos, o valor padrão de Default TTL será alterado para o valor de Minimum TTL.

Cookies progressivos

Note

Para as origens do Amazon S3, essa opção se aplica somente a buckets configurados como um endpoint de site.

Especifique se você deseja que o CloudFront encaminhe cookies ao seu servidor de origem e, caso afirmativo, para quais servidores. Se você optar por encaminhar apenas cookies selecionados (uma lista de permissões de cookies), insira os nomes dos cookies no campo Lista de permissões de cookies. Se você escolher All (Todos), o CloudFront encaminhará todos os cookies, independentemente de quantos sua aplicação usar.

O Amazon S3 não processa cookies, e o encaminhamento deles para a origem reduz a capacidade de armazenamento em cache. Para comportamentos de cache que estiverem encaminhando solicitações para uma origem do Amazon S3, escolha None (Nenhum) em Forward Cookies (Encaminhar cookies).

Para obter mais informações sobre como encaminhar cookies para a origem, acesse [Conteúdo em cache com base em cookies](#).

Lista de permissões de cookies

Note

Para as origens do Amazon S3, essa opção se aplica somente a buckets configurados como um endpoint de site.

Se você selecionou Lista de permissões na lista Encaminhar cookies, no campo Lista de permissões de cookies, insira os nomes dos cookies a serem encaminhados pelo CloudFront ao servidor de origem para esse comportamento de cache. Insira cada nome de cookie em uma nova linha.

Você pode especificar os seguintes curingas para especificar nomes de cookies:

- * corresponde a zero ou mais caracteres no nome do cookie
- ? corresponde a exatamente um caractere no nome do cookie

Por exemplo, imagine que o visualizador solicite que um objeto inclua um cookie denominado:

`userid_`*member-number*

Em que cada usuário tem um valor exclusivo para *member-number*. Você deseja que o CloudFront armazene em cache uma versão separada do objeto para cada membro. Isso pode ser feito encaminhando todos os cookies para a sua origem, mas as solicitações do visualizador incluem alguns cookies que você não quer que o CloudFront armazene em cache. Como alternativa, é possível especificar o seguinte valor como um nome de cookie, fazendo com que o CloudFront encaminhe todos os cookies que comecem com `userid_` para a origem:

`userid_*`

Para saber o número máximo atual de nomes de cookies que podem ser incluídos na lista de permissões para cada comportamento de cache ou para solicitar uma cota maior (anteriormente conhecida como limite), consulte [Cotas para cookies \(configurações de cache herdadas\)](#).

Encaminhamento e armazenamento em cache de strings de consulta

O CloudFront pode armazenar em cache diferentes versões do seu conteúdo com base nos valores dos parâmetros da query string. Escolha uma das seguintes opções:

None (Improves Caching)

Escolha essa opção se a sua origem retornar a mesma versão de um objeto, independentemente dos valores dos parâmetros de query string. Isso aumenta a probabilidade de o CloudFront atender a uma solicitação do cache, melhorando a performance e reduzindo a carga na origem.

Encaminhar tudo, cache com base em lista de permissões

Escolha essa opção caso o servidor de origem retorne diferentes versões dos objetos com base em um ou mais parâmetros de query string. Depois, especifique os parâmetros que você deseja que o CloudFront use como base para o armazenamento em cache no campo [Lista de permissões de strings de consulta](#).

Forward all, cache based on all

Escolha essa opção caso o servidor de origem retorne diferentes versões dos objetos para todos os parâmetros de query string.

Para obter mais informações sobre como armazenar em cache com base nos parâmetros de query string, inclusive como melhorar a performance, consulte [Conteúdo em cache com base em parâmetros de string de consulta](#).

Lista de permissões de strings de consulta

Essa configuração aplica-se somente quando a opção Encaminhar tudo, com base em cache na lista de permissões é escolhida para [Encaminhamento e armazenamento em cache de strings de consulta](#). É possível especificar os parâmetros da string de consulta que você deseja que o CloudFront use como base para o armazenamento em cache.

Smooth Streaming

Escolha Yes (Sim) se quiser distribuir arquivos de mídia no formato Microsoft Smooth Streaming e não tiver um servidor IIS.

Escolha No (Não) se você tiver um servidor Microsoft IIS que deseja usar como origem para distribuir arquivos de mídia no formato Microsoft Smooth Streaming, ou se não estiver distribuindo arquivos de mídia do Smooth Streaming.

Note

Se você especificar Yes (Sim), é possível distribuir outro conteúdo usando esse comportamento de cache se o conteúdo corresponder ao valor de Path Pattern (Padrão de caminho).

Para ter mais informações, consulte [Configurar vídeo sob demanda para o Microsoft Smooth Streaming](#).

Restringir acesso do visualizador (usar URLs ou cookies assinados)

Se você quiser solicitações de objetos correspondentes a PathPattern para que esse comportamento de cache use URLs públicos, escolha No.

Se você quiser solicitações de objetos correspondentes a PathPattern para que esse comportamento de cache use signed URLs, escolha Yes. Em seguida, especifique as contas da AWS que você deseja usar para criar signed URLs. Essas contas são conhecidas como assinantes confiáveis.

Para obter mais informações sobre assinantes confiáveis, consulte [Especificar os assinantes que podem criar URLs e cookies assinados](#).

Signatários confiáveis

Essa configuração aplica-se somente ao escolher Sim para Restringir o acesso do visualizador (usar URLs ou cookies assinados).

Selecione quais contas da AWS você quer usar como assinantes confiáveis para esse comportamento de cache:

- Self (Próprio): use a conta na qual você está conectado no AWS Management Console como assinante confiável. Se você estiver conectado como um usuário do IAM, a conta da AWS associada será adicionada como assinante confiável.
- Specify Accounts (Especificar contas): insira números de contas para signatários confiáveis no campo AWS Account Numbers (Números de contas da AWS).

Para criar signed URLs, uma conta da AWS deve ter pelo menos um par de chaves do CloudFront ativo.

Important

Se você está atualizando uma distribuição que já está usando para distribuir conteúdo, adicione assinantes confiáveis somente quando estiver pronto para começar a gerar signed URLs para seus objetos. Depois de adicionar assinantes confiáveis a uma distribuição, os usuários devem usar signed URLs para acessar os objetos correspondentes a PathPattern para esse comportamento de cache.

Números de Conta da AWS

Essa configuração aplica-se somente ao selecionar Especificar contas para Signatários confiáveis.

Se quiser criar signed URLs usando Contas da AWS além ou em vez da conta atual, insira um número de Conta da AWS por linha neste campo. Observe o seguinte:

- As contas especificadas devem ter pelo menos um par de chaves do CloudFront ativo. Para ter mais informações, consulte [Criar pares de chaves para os assinantes](#).

- Não é possível criar pares de chaves do CloudFront para usuários do IAM, portanto, você não pode usar usuários do IAM como assinantes confiáveis.
- Consulte informações sobre como obter o número da Conta da AWS de uma conta em [View Conta da AWS identifiers](#) no Guia de referência de gerenciamento de Conta da AWS.
- Se você inserir o número da conta atual, o CloudFront marcará automaticamente a caixa de seleção Self (Próprio) e removerá o número da conta da lista AWS Account Numbers (Números de contas da AWS).

Compactar objetos automaticamente

Se você desejar que o CloudFront compacte automaticamente arquivos de determinados tipos quando os visualizadores forem compatíveis com o conteúdo compactado, escolha Yes (Sim). Quando o CloudFront compacta o conteúdo, os downloads são mais rápidos porque os arquivos são menores, e os sites são renderizados mais rapidamente para os usuários. Para ter mais informações, consulte [Fornecer arquivos compactados](#).

Evento do CloudFront

Essa configuração aplica-se às Associações de funções do Lambda.

É possível optar por executar uma função do Lambda quando um ou mais dos seguintes eventos do CloudFront ocorrerem:

- Quando o CloudFront receber uma solicitação de um visualizador (solicitação do visualizador)
- Antes do CloudFront encaminhar uma solicitação para a origem (solicitação da origem)
- Quando o CloudFront receber uma resposta da origem (resposta da origem)
- Antes do CloudFront retornar a resposta para o visualizador (resposta do visualizador)

Para ter mais informações, consulte [Escolher o evento para acionar a função](#).

ARN da função do Lambda.

Essa configuração aplica-se às Associações de funções do Lambda.

Especifique o Nome de recurso da Amazon (ARN) da função do Lambda para a qual deseja adicionar um trigger. Para saber mais sobre como obter o ARN de uma função, consulte a etapa 1 do procedimento [Adicionar triggers usando o console do CloudFront](#).

Incluir corpo

Essa configuração aplica-se às Associações de funções do Lambda.

Consulte mais informações em [Acesso ao corpo da solicitação com escolha da opção de inclusão de corpo](#).

Configurações de distribuição

Os valores a seguir se aplicam a toda a distribuição.

Tópicos

- [Classe de preço](#)
- [ACL da WEB do AWS WAF](#)
- [Nomes de domínio alternativos \(CNAMEs\)](#)
- [Certificado SSL](#)
- [Suporte ao cliente SSL personalizado](#)
- [Política de segurança \(versão mínima de SSL/TLS\)](#)
- [Versões de HTTP compatíveis](#)
- [Objeto raiz padrão](#)
- [Registro em log](#)
- [Bucket para logs](#)
- [Prefixo de log](#)
- [Registro em log de cookies](#)
- [Enable IPv6](#)
- [Comentário](#)
- [Estado de distribuição](#)

Classe de preço

Escolha o preço que corresponde ao preço máximo que você está disposto a pagar pelo serviço do CloudFront. Por padrão, o CloudFront fornece seus objetos de pontos de presença em todas as regiões do CloudFront.

Para ter mais informações sobre as classes de preço e como sua escolha de classe de preço afeta a performance do CloudFront para sua distribuição, consulte [Definição de preços do CloudFront](#).

ACL da WEB do AWS WAF

Você pode proteger sua distribuição do CloudFront com o [AWS WAF](#), um firewall de aplicações da Web que permite proteger suas aplicações e APIs da web para bloquear solicitações antes que elas cheguem aos servidores. Você pode usar o [Habilitar o AWS WAF para distribuições](#) ao criar ou editar uma distribuição do CloudFront.

Você também pode configurar posteriormente proteções de segurança adicionais para outras ameaças específicas de sua aplicação no console do AWS WAF em <https://console.aws.amazon.com/wafv2/>.

Para obter mais informações sobre o AWS WAF, consulte o [Guia do desenvolvedor do AWS WAF](#).

Nomes de domínio alternativos (CNAMEs)

Opcional. Especifique um ou mais nomes de domínio que você deseja usar nas URLs de seus objetos em vez do nome de domínio atribuído pelo CloudFront ao criar sua distribuição. Você deve possuir o nome de domínio, ou ter autorização para usá-lo, o que você verifica adicionando um certificado SSL/TLS.

Por exemplo, se você quiser que o URL do objeto:

`/images/image.jpg`

Seja parecido com:

`https://www.example.com/images/image.jpg`

Em vez de:

`https://d111111abcdef8.cloudfront.net/images/image.jpg`

Adicione um CNAME para `www.example.com`.

Important

Se você adicionar um CNAME para `www.example.com` à distribuição, também deverá fazer o seguinte:

- Crie (ou atualize) um registro CNAME no serviço de DNS para rotear consultas de `www.example.com` para `d111111abcdef8.cloudfront.net`.

- Adicione um certificado ao CloudFront de uma autoridade de certificação (CA) confiável que abranja o nome de domínio (CNAME) que você adicionar à distribuição, para validar a autorização para usar o nome de domínio.

Você deve ter permissão para criar um registro CNAME com o provedor de serviço de DNS para o domínio. Normalmente, isso significa que você possui o domínio ou que está desenvolvendo um aplicativo para o proprietário do domínio.

Para saber o número máximo atual de nomes de domínio alternativos que podem ser adicionados a uma distribuição ou para solicitar uma cota maior (anteriormente conhecida como limite), consulte [Cotas gerais para distribuições](#).

Para obter mais informações sobre nomes de domínio alternativo, consulte [Usar URLs personalizados adicionando nomes de domínio alternativos \(CNAMEs\)](#). Para mais informações sobre URLs do CloudFront, consulte [Personalizar o formato do URL para arquivos no CloudFront](#).

Certificado SSL

Se você especificou um nome de domínio alternativo para usar com a distribuição, selecione Custom SSL Certificate (Certificado SSL personalizado) e, para validar a autorização para usar o nome de domínio alternativo, escolha um certificado que o abranja. Se você quiser que os visualizadores usem HTTPS para acessar seus objetos, escolha as configurações que oferecem suporte para isso.

Note

Antes de poder especificar um certificado SSL personalizado, você precisa especificar um nome de domínio alternativo válido. Para ter mais informações, consulte [Requisitos para o uso de nomes de domínio alternativos](#) e [Usar nomes de domínio alternativos e HTTPS](#).

- Default CloudFront Certificate (Certificado padrão do CloudFront) (*.cloudfront.net): escolha essa opção se quiser usar o nome de domínio do CloudFront nos URLs para seus objetos, como `https://d1111111abcdef8.cloudfront.net/image1.jpg`.
- Custom SSL Certificate (Certificado SSL personalizado): escolha essa opção se quiser usar seu próprio nome de domínio nos URLs dos seus objetos como um nome de domínio alternativo, por exemplo `https://example.com/image1.jpg`. Em seguida, escolha um certificado para usar

que abranja o nome de domínio alternativo. A lista de certificados pode incluir qualquer um dos seguintes:

- Certificados fornecidos pelo AWS Certificate Manager
- Os certificados adquiridos de uma autoridade de certificação de terceiros e carregados no ACM
- Os certificados adquiridos de autoridades de certificação de terceiros e carregados no armazenamento de certificados do IAM

Se você escolher essa configuração, recomendamos que use apenas um nome de domínio alternativo nos URLs dos seus objetos (<https://example.com/logo.jpg>). Se você usar o nome de domínio da distribuição do CloudFront (<https://d1111111abcdef8.cloudfront.net/logo.jpg>) e um cliente usar um visualizador mais antigo que não seja compatível com SNI, como o visualizador responderá vai depender do valor escolhido para Clients Supported (Clientes compatíveis):

- All Clients (Todos os clientes): o visualizador exibe um aviso porque o nome de domínio do CloudFront não corresponde ao nome de domínio do certificado SSL/TLS.
- Only Clients that Support Server Name Indication (SNI) (Somente os clientes que oferecem suporte à indicação de nome de servidor (SNI)): o CloudFront interrompe a conexão com o visualizador sem retornar o objeto.

Suporte ao cliente SSL personalizado

Aplica-se somente quando você escolhe Certificado SSL personalizado (exemplo.com) para Certificado SSL. Se você especificou um ou mais nomes de domínio alternativos e um certificado SSL personalizado para a distribuição, defina como você deseja que o CloudFront atenda às solicitações HTTPS:

- Clientes que oferecem suporte a SNI (Indicação de nome de servidor) – (Recomendado): com essa configuração, praticamente todos os navegadores e clientes da Web modernos podem se conectar à distribuição, porque eles oferecem suporte a SNI. No entanto, alguns visualizadores podem usar navegadores da Web mais antigos ou clientes incompatíveis com SNI, o que significa que não conseguem se conectar à distribuição.

Para aplicar essa configuração usando a API do CloudFront, especifique `sni-only` no campo `SSLSupportMethod`. No AWS CloudFormation, o campo recebe o nome `SslSupportMethod` (observe o tamanho diferente das letras).

- Suporte de clientes herdados: com essa configuração, navegadores da Web e clientes mais antigos incompatíveis com SNI podem se conectar à distribuição. No entanto, essa configuração

gera cobranças mensais adicionais. Para saber o preço exato, acesse a página [Definição de preços do Amazon CloudFront](#) e pesquise SSL personalizado de IP dedicado na página.

Para aplicar essa configuração usando a API do CloudFront, especifique `vip` no campo `SSLSupportMethod`. No AWS CloudFormation, o campo recebe o nome `SslSupportMethod` (observe o tamanho diferente das letras).

Para ter mais informações, consulte [Escolher como o CloudFront atende a solicitações HTTPS](#).

Política de segurança (versão mínima de SSL/TLS)

Especifique a política de segurança que você deseja que o CloudFront use para conexões HTTPS com os visualizadores (clientes). Uma política de segurança determina duas configurações:

- O protocolo SSL/TLS mínimo que o CloudFront usa para se comunicar com os visualizadores
- A criptografia que o CloudFront pode usar para criptografar o conteúdo que retorna aos visualizadores.

Para mais informações sobre as políticas de segurança, incluindo os protocolos e as cifras que cada uma inclui, consulte [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

As políticas de segurança que estão disponíveis dependem dos valores que você especifica para o SSL Certificate (Certificado SSL) e o Custom SSL Client Support (Suporte ao cliente SSL personalizado) (conhecidos como `CloudFrontDefaultCertificate` e `SSLSupportMethod` na API do CloudFront):

- Quando o SSL Certificate (Certificado SSL) for Default CloudFront Certificate (`*.cloudfront.net`) (Certificado padrão do CloudFront (`*.cloudfront.net`)) (quando `CloudFrontDefaultCertificate` for `true` na API), o CloudFront definirá automaticamente a política de segurança como TLSv1.
- Quando SSL Certificate (Certificado SSL) for Custom SSL Certificate (`example.com`) (Certificado SSL personalizado, `example.com`) e Custom SSL Client Support (Suporte ao cliente SSL personalizado) for Clients that Support Server Name Indication (SNI) – (Recommended) (Clientes que comportam a indicação de nome do servidor, SNI – recomendado) (quando `CloudFrontDefaultCertificate` for `false` e `SSLSupportMethod` for `sni-only` na API), será possível escolher entre as seguintes políticas de segurança:
 - TLSv1.2_2021

- TLSv1.2_2019
 - TLSv1.2_2018
 - TLSv1.1_2016
 - TLSv1_2016
 - TLSv1
- Quando SSL Certificate (Certificado SSL) for Custom SSL Certificate (example.com) (Certificado SSL personalizado, example.com) e Custom SSL Client Support (Suporte ao cliente SSL personalizado) for Legacy Clients Support (Suporte a clientes herdados) (quando CloudFrontDefaultCertificate for false e SSLSupportMethod for vip na API), será possível escolher entre as seguintes políticas de segurança:
 - TLSv1
 - SSLv3

Nesta configuração, as políticas de segurança TLSv1.2_2021, TLSv1.2_2019, TLSv1.2_2018, TLSv1.1_2016, e TLSv1_2016 não estão disponíveis na API nem no console do CloudFront. Se quiser utilizar uma dessas políticas de segurança, você terá as seguintes opções:

- Avalie se a distribuição precisa de suporte a clientes legados com endereços IP dedicados. Se seus visualizadores comportarem a [indicação de nome de servidor \(SNI\)](#), recomendamos atualizar a definição Custom SSL Client Support (Suporte ao cliente SSL personalizado) de sua distribuição para Clients that Support Server Name Indication (SNI) (Clientes que oferecem suporte a indicação de nome de servidor, SNI) (defina SSLSupportMethod como sni-only na API). Isso permite usar qualquer uma das políticas de segurança TLS disponíveis, e também pode reduzir as cobranças do CloudFront.
- Se for necessário manter o suporte a clientes herdados com endereços IP dedicados, será possível solicitar uma das outras políticas de segurança para o TLS (TLSv1.2_2021, TLSv1.2_2019, TLSv1.2_2018, TLSv1.1_2016 ou TLSv1_2016) criando um caso na [Central de Suporte da AWS](#).

Note

Antes de entrar em contato com o AWS Support para solicitar essa alteração, considere o seguinte:

- Quando você adiciona uma dessas políticas de segurança (TLSv1.2_2021, TLSv1.2_2019, TLSv1.2_2018, TLSv1.1_2016 ou TLSv1_2016) a uma distribuição de suporte a clientes herdados, a política de segurança é aplicada a todas as solicitações

de visualizador que não são de SNI para todas as distribuições de suporte a clientes herdadas em sua conta da AWS. No entanto, quando os visualizadores enviam solicitações SNI para uma distribuição com o Suporte a clientes legados, a política de segurança dessa distribuição se aplica. Para garantir que a política de segurança desejada seja aplicada a todas as solicitações do visualizador enviadas a todas as distribuições de suporte de clientes herdadas em sua conta da AWS, adicione a política de segurança desejada a cada distribuição individualmente.

- Por definição, a nova política de segurança não é compatível com as mesmas criptografias e protocolos que a antiga. Por exemplo, se optar por atualizar a política de segurança de uma distribuição de TLSv1 para TLSv1.1_2016, essa distribuição não será mais compatível com a criptografia DES-CBC3-SHA. Para mais informações sobre as cifras e os protocolos com os quais cada política de segurança é compatível, consulte [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

Versões de HTTP compatíveis

Escolha as versões HTTP às quais deseja que a distribuição ofereça suporte quando os visualizadores se comunicarem com o CloudFront.

Para os visualizadores e o CloudFront usarem HTTP/3, os visualizadores devem ser compatíveis com TLSv1.2 ou posterior e com Server Name Indication (SNI). O CloudFront não oferece suporte nativo para gRPC por HTTP/2.

Para os visualizadores e o CloudFront usarem HTTP/3, os visualizadores devem ser compatíveis com TLSv1.3 e com Server Name Indication (SNI). O CloudFront é compatível com a migração de conexão HTTP/3 para permitir que o visualizador alterne de rede sem perder a conexão. Para obter mais informações sobre migração de conexões, consulte [Connection Migration](#) (Migração de conexões) no RFC 9000.

Note

Para obter mais informações sobre as criptografias TLSv1.3 compatíveis, consulte [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

Objeto raiz padrão

Opcional. O objeto que você deseja que o CloudFront solicite da sua origem (por exemplo, `index.html`) quando o visualizador solicitar o URL raiz da sua distribuição (`https://www.example.com/`), em vez de um objeto nela (`https://www.example.com/product-description.html`). A especificação de um objeto raiz padrão evita a exposição do conteúdo da distribuição.

O tamanho máximo do nome é 255 caracteres. O nome pode conter espaços ou um destes caracteres:

- A-Z, a-z
- 0-9
- `_ - . * $ / ~ " ' "`
- `&`, passado e retornado como `&`;

Ao especificar o objeto raiz padrão, insira apenas o nome do objeto, por exemplo, `index.html`. Não adicione `/` antes do nome do objeto.

Para ter mais informações, consulte [Especificar um objeto raiz padrão](#).

Registro em log

Se você deseja que o CloudFront registre informações sobre cada solicitação de um objeto e armazene os arquivos de log em um bucket do Amazon S3. É possível habilitar ou desabilitar o registro a qualquer momento. Não há custo adicional ao habilitar o registro em log, mas você acumula as cobranças normais do Amazon S3 pelo armazenamento e acesso a arquivos em um bucket do Amazon S3. Você pode excluir os logs a qualquer momento. Para mais informações sobre os logs de acesso do CloudFront, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Bucket para logs

Se você escolher On (Ativado) em Logging (Registro em log), o bucket do Amazon S3 no qual você deseja que o CloudFront armazene logs de acesso, por exemplo, `myLogs-amzn-s3-demo-bucket.s3.amazonaws.com`.

⚠ Important

Não escolha um bucket do Amazon S3 nas Regiões da AWS a seguir. O CloudFront não entrega logs padrão para buckets nestas regiões:

- Africa (Cape Town)
- Ásia-Pacífico (Hong Kong)
- Ásia-Pacífico (Hyderabad)
- Ásia-Pacífico (Jacarta)
- Ásia-Pacífico (Melbourne)
- Oeste do Canadá (Calgary)
- Europa (Milão)
- Europa (Espanha)
- Europa (Zurique)
- Israel (Tel Aviv)
- Oriente Médio (Barém)
- Oriente Médio (Emirados Árabes Unidos)

Se você permitir o registro em log, o CloudFront registrará informações sobre cada solicitação do usuário final de um objeto e armazenará os arquivos no bucket do Amazon S3 especificado. É possível habilitar ou desabilitar o registro a qualquer momento. Para mais informações sobre os logs de acesso do CloudFront, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

ℹ Note

Você deve ter as permissões necessárias para obter e atualizar as ACLs do bucket do Amazon S3, e a ACL do S3 do bucket deve conceder a você FULL_CONTROL. Assim, o CloudFront pode conceder à conta `awslogsdelivery` permissão para salvar os arquivos de log no bucket. Para ter mais informações, consulte [Permissões necessárias para configurar o registro em log padrão e acessar os arquivos de log](#).

Prefixo de log

Opcional. Se você escolher On (Ativado) em Logging (Registro em log), especifique a string, se houver, a ser prefixada pelo CloudFront nos nomes de arquivo do log de acesso dessa distribuição, por exemplo, `exampleprefix/`. A barra no final (/) é opcional, mas recomendada para simplificar a navegação em seus arquivos de log. Para mais informações sobre os logs de acesso do CloudFront, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Registro em log de cookies

Se você quiser que o CloudFront inclua cookies nos logs de acesso, escolha On (Ativado). Se você optar por incluir cookies nos logs, o CloudFront registrará em log todos os cookies, independentemente da configuração dos comportamentos de cache dessa distribuição: encaminhar todos os cookies, nenhum cookie ou uma lista específica de cookies para a origem.

O Amazon S3 não processa cookies, portanto, a menos que sua distribuição também inclua um Amazon EC2 ou outra origem personalizada, recomendamos que você escolha Off (Desativado) para o valor de Cookie Logging (Registro em log de cookies).

Para obter mais informações sobre cookies, acesse [Conteúdo em cache com base em cookies](#).

Enable IPv6

IPv6 é uma nova versão do protocolo IP. É a substituição eventual do IPv4 e usa um espaço de endereço maior. O CloudFront sempre responde a solicitações do IPv4. Se quiser que o CloudFront responda a solicitações de endereços IP IPv4 (como 192.0.2.44) e IPv6 (como 2001:0db8:85a3::8a2e:0370:7334), selecione Enable IPv6 (Habilitar IPv6).

Em geral, você deverá habilitar o IPv6 se tiver usuários em redes IPv6 que desejam acessar seu conteúdo. No entanto, se você estiver usando signed URLs ou cookies para restringir o acesso ao seu conteúdo e uma política personalizada que inclui o parâmetro `IpAddress` para restringir os endereços IP que podem acessar seu conteúdo, não habilite o IPv6. Se você quiser restringir o acesso a algum conteúdo, mas não a outros, por endereço IP (ou restringir o acesso, mas não por endereço IP), é possível criar duas distribuições. Para obter informações sobre como criar signed URLs usando uma política personalizada, consulte [Criar um URL assinado usando uma política personalizada](#). Para obter informações sobre como criar signed cookies usando uma política personalizada, consulte [Definir cookies assinados usando uma política personalizada](#).

Se estiver usando um conjunto de registros de recursos de alias do Route 53 para rotear tráfego para sua distribuição do CloudFront, você deverá criar um segundo conjunto de registros de recursos de alias quando:

- Você habilitar o IPv6 para a distribuição
- Você estiver usando nomes de domínio alternativos nos URLs dos seus objetos

Para mais informações, consulte [Como rotear o tráfego para uma distribuição do Amazon CloudFront usando seu nome de domínio](#) no Guia do desenvolvedor do Amazon Route 53.

Se você criar um conjunto de registros de recursos de CNAME com o Route 53 ou outro serviço de DNS, não será necessário fazer nenhuma alteração. Um registro CNAME roteará o tráfego para a distribuição, independentemente do formato de endereço IP da solicitação do visualizador.

Se você habilitar o IPv6 e os logs de acesso do CloudFront, a coluna `c-ip` incluirá os valores nos formatos IPv4 e IPv6. Para ter mais informações, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Note

Para manter a alta disponibilidade do cliente, o CloudFront responderá a solicitações do visualizador usando IPv4 se nossos dados sugerirem que o IPv4 fornecerá uma experiência de usuário melhor. Para descobrir qual porcentagem de solicitações do CloudFront é por IPv6, habilite o registro em log do CloudFront em sua distribuição e analise a coluna `c-ip`, que contém o endereço IP do visualizador que fez a solicitação. Essa porcentagem deve aumentar com o tempo, mas continuará sendo parte mínima do tráfego, pois o IPv6 ainda não é compatível pelo todas as redes dos visualizadores em todo o mundo. Algumas redes de visualizador têm excelente suporte a IPv6, mas outras não são compatíveis com ele. (Uma rede de visualizador é semelhante à sua operadora de Internet ou sem fio doméstica.) Para mais informações sobre nosso suporte a IPv6, consulte [Perguntas frequentes sobre o CloudFront](#). Para obter informações sobre como permitir logs de acesso, consulte os campos [Registro em log](#), [Bucket para logs](#) e [Prefixo de log](#).

Comentário

Opcional. Ao criar uma distribuição, você pode incluir um comentário com até 128 caracteres. Você pode atualizá-lo qualquer momento.

Estado de distribuição

Indica se você deseja que a distribuição seja ativada ou desativada após a implantação:

- **Enabled:** assim que a distribuição for totalmente implantada, você poderá implantar links que usam o nome de domínio da distribuição, e os usuários poderão recuperar o conteúdo. Sempre que uma distribuição estiver ativada, o CloudFront aceitará e lidará com qualquer solicitação do usuário final de conteúdo que use o nome de domínio associado a essa distribuição.

Ao criar, modificar ou excluir uma distribuição do CloudFront, leva um tempo para que as alterações sejam propagadas para o banco de dados do CloudFront. Uma solicitação imediata de informações sobre uma distribuição pode não refletir a alteração. A propagação é normalmente concluída em minutos, mas uma alta carga do sistema ou partição de rede pode aumentar esse tempo.

- **Disabled:** mesmo que a distribuição possa ser implantada e esteja pronta para ser usada, os usuários não poderão usá-la. Sempre que uma distribuição estiver desativada, o CloudFront não aceitará nenhuma solicitação do usuário final que use o nome de domínio associado a essa distribuição. Enquanto você não alternar a distribuição de desativada para habilitada (atualizando a configuração dela), ninguém poderá usá-la.

Você pode alternar uma distribuição entre desativada e ativada sempre que desejar. Siga o processo de atualização da configuração de uma distribuição. Para ter mais informações, consulte [Atualizar uma distribuição](#).

Páginas de erro personalizadas e erro de armazenamento em cache

É possível solicitar que o CloudFront retorne um objeto para o visualizador (por exemplo, um arquivo HTML) quando o Amazon S3 ou a origem personalizada retornar um código de status HTTP 4xx ou 5xx para o CloudFront. Também é possível especificar o tempo de armazenamento no cache do CloudFront de uma resposta de erro da sua origem ou de uma página de erro personalizada. Para ter mais informações, consulte [Criar uma página de erro personalizada para códigos de status HTTP específicos](#).

Note

Os valores a seguir não são incluídos no assistente "Criar distribuição", portanto, você poderá configurar páginas de erro personalizadas somente quando atualizar uma distribuição.

Tópicos

- [Código de erro HTTP](#)
- [Caminho da página de resposta](#)
- [Código de resposta HTTP](#)
- [Erro ao armazenar TTL mínimo em cache \(segundos\)](#)

Código de erro HTTP

O código de status HTTP para o qual você deseja que o CloudFront retorne uma página de erro personalizada. É possível configurar o CloudFront para retornar páginas de erro personalizadas para nenhum, alguns ou todos os códigos de status HTTP armazenados em cache pelo CloudFront.

Caminho da página de resposta

O caminho para a página de erro personalizada (por exemplo, `/4xx-errors/403-forbidden.html`) que você deseja que o CloudFront retorne para um visualizador quando sua origem retorna o código de status HTTP especificado para Error Code (Código de erro) (por exemplo, 403). Se você quiser armazenar seus objetos e páginas de erro personalizadas em locais diferentes, sua distribuição deverá incluir um comportamento de cache para o qual o seguinte é verdadeiro:

- O valor de Path Pattern é correspondente às suas mensagens de erro personalizadas. Por exemplo, imagine que você salvou páginas de erro personalizadas para erros 4xx em um bucket do Amazon S3 em um diretório denominado `/4xx-errors`. Sua distribuição deverá incluir um comportamento de cache para o qual o padrão de caminho roteia solicitações de suas páginas de erro personalizadas para esse local, por exemplo `/4xx-errors/*`.
- O valor de Origin especifica o valor de Origin ID da origem que contém suas páginas de erro personalizadas.

Código de resposta HTTP

O código de status HTTP que você deseja que o CloudFront retorne para o visualizador com a página de erro personalizada.

Erro ao armazenar TTL mínimo em cache (segundos)

O tempo mínimo que você deseja que o CloudFront armazene em cache as respostas de erro do seu servidor de origem.

Restrições geográficas

Se precisar impedir que usuários de alguns países acessem o conteúdo, configure a distribuição do CloudFront com uma Lista de permissões ou uma Lista de bloqueio. Não há custo adicional para configurar restrições geográficas. Para ter mais informações, consulte [Restringir a distribuição geográfica do conteúdo](#).

Testar uma distribuição

Depois de criar a distribuição, o CloudFront saberá onde seu servidor de origem está, e você saberá o nome de domínio associado à distribuição. Para testar a distribuição, faça o seguinte:

1. Espere até que sua distribuição seja implantada.
 - Veja os detalhes da distribuição no console. Quando sua distribuição terminar a implantação, o campo Last modified (Última modificação) muda de Deploying (Implantando) para uma data e hora.
2. Crie links para seus objetos com o nome de domínio do CloudFront usando o procedimento a seguir.
3. Teste os links. O CloudFront oferece os objetos para sua página da web ou aplicação.

Criar links para seus objetos

Use o procedimento a seguir para criar links de teste para os objetos na distribuição web do CloudFront.

Para criar links para objetos em uma distribuição na web

1. Copie o código HTML a seguir em um novo arquivo, substitua *domain-name* pelo nome do domínio da sua distribuição e substitua *object-name* pelo nome do seu objeto.

```
<html>
<head>My CloudFront Test</head>
<body>
<p>My text content goes here.</p>
<p>
</html>
```

Por exemplo, caso seu nome de domínio seja `d111111abcdef8.cloudfront.net` e seu objeto seja `image.jpg`, o URL do link será:

```
https://d111111abcdef8.cloudfront.net/image.jpg.
```

Se o objeto estiver em uma pasta no seu servidor de origem, a pasta também deverá ser incluída no URL. Por exemplo, se `image.jpg` estiver localizado na pasta de imagens em seu servidor de origem, o URL será:

```
https://d111111abcdef8.cloudfront.net/images/image.jpg
```

2. Salve o código HTML em um arquivo com extensão `.html`.
3. Abra a página da web em um navegador para garantir que o objeto possa ser visualizado.

O navegador retornará a página com o arquivo de imagem incorporado, fornecido pelo ponto de presença que o CloudFront determinou como apropriado para fornecer o objeto.

Atualizar uma distribuição

No console do CloudFront, você vê as distribuições do CloudFront associadas à sua conta da AWS, visualiza as configurações de uma distribuição e atualiza a maioria das configurações. Lembre-se de que as alterações de configurações que você fizer não entrarão em vigor até que a distribuição seja propagada para os locais da borda da AWS.

Para atualizar uma distribuição do CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Selecione o ID de uma distribuição. A lista inclui todas as distribuições associadas à conta da AWS usada ao fazer login no console do CloudFront.
3. Para editar as configurações de uma distribuição, escolha a aba Distribution Settings (Configurações de distribuição).
4. Para atualizar configurações gerais, escolha Edit (Editar). Caso contrário, escolha a guia para as configurações que você deseja atualizar: Origens ou Comportamentos.
5. Faça as atualizações e, em seguida, para salvar suas alterações, escolha Yes, Edit (Sim, editar). Para obter informações sobre os campos, consulte os seguintes tópicos:
 - General settings (Configurações gerais: [Configurações de distribuição](#))
 - Origin settings (Configurações de origem: [Configurações de origem](#))
 - Cache behavior settings (Configurações de comportamento de cache: [Configurações de comportamento de cache](#))
6. Se você deseja excluir uma origem em sua distribuição, faça o seguinte:
 - a. Escolha Comportamentos e certifique-se de ter movido para outra origem todos os comportamentos de cache padrão associados à origem.
 - b. Selecione Origens, depois selecione uma origem.
 - c. Escolha Delete.

Também é possível atualizar uma distribuição usando a API do CloudFront:

- Para atualizar uma distribuição, consulte [UpdateDistribution](#) na Referência da API do Amazon CloudFront.

Important

Ao atualizar sua distribuição, esteja ciente de que são necessários vários campos adicionais que não são necessários para criar uma distribuição. Para ajudar a garantir que todos os campos obrigatórios sejam incluídos quando você usar a API do CloudFront para atualizar

uma distribuição, siga as etapas descritas em [UpdateDistribution](#) na Referência de API do Amazon CloudFront.

Ao salvar as alterações na configuração da sua distribuição, o CloudFront começará a propagá-las para todos os pontos de presença. Alterações de configuração sucessivas se propagam na respectiva ordem. Enquanto sua configuração não for atualizada em um ponto de presença, o CloudFront continuará fornecendo seu conteúdo desse local com base na configuração anterior. Após a atualização da sua configuração em um ponto de presença, o CloudFront imediatamente começará a fornecer seu conteúdo desse local com base na nova configuração.

Suas alterações não são propagadas para todos os locais de borda ao mesmo tempo. Enquanto o CloudFront propaga as alterações, não é possível determinar se um local da borda está fornecendo seu conteúdo com base na configuração anterior ou na nova configuração.

Para ver quando suas alterações são propagadas, veja os detalhes da distribuição no console. O campo Última modificação muda de Implantando para uma data e hora quando a implantação é concluída.

Marcar uma distribuição

Tags são palavras ou frases que podem ser usadas para identificar e organizar seus recursos da AWS. É possível adicionar várias tags a cada recurso, e cada tag inclui uma chave e um valor definidos por você. Por exemplo, a chave pode ser "domínio" e o valor pode ser "exemplo.com". Você pode pesquisar e filtrar seus recursos de acordo com as tags que adicionar.

É possível usar tags com o CloudFront, como estes exemplos:

- Aplique permissões baseadas em tags a distribuições do CloudFront. Para ter mais informações, consulte [ABAC com o CloudFront](#).
- Monitore as informações de faturamento em categorias diferentes. Quando você aplica tags a distribuições do CloudFront ou a outros recursos da AWS (como instâncias do Amazon EC2 ou buckets do Amazon S3) e as ativa, a AWS gera um relatório de alocação de custos como um arquivo CSV (valores separado por vírgula) com seu uso e custos agregados por tags ativas.

É possível aplicar tags que representem categorias de negócios (como centros de custos, nomes de aplicativos ou proprietários) para organizar seus custos de vários serviços. Para obter mais informações sobre como usar tags para alocação de custos, consulte [Como usar tags de alocação de custo](#) no Manual do usuário do AWS Billing.

Observações

- Você pode marcar distribuições com tags, mas não pode marcar origin access identities ou invalidações com tags.
- O [Editor de tags](#) e os [Grupos de recursos](#) não são compatíveis com o CloudFront no momento.
- Para saber o número máximo atual de tags que podem ser adicionadas a uma distribuição, consulte [Cotas gerais](#).

Sumário

- [Restrições de tags](#)
- [Adicionar, editar e excluir tags para distribuições](#)
- [Marcação programática](#)

Restrições de tags

As restrições básicas a seguir se aplicam a tags:

- Consulte o número máximo de tags por distribuição em [Cotas gerais](#).
- Comprimento máximo da chave: 128 caracteres Unicode
- Comprimento máximo do valor: 256 caracteres Unicode
- Valores válidos de chave e valor: a-z, A-Z, 0-9, espaço e os seguintes caracteres: `_ . : / = + - e @`
- As chaves e os valores de tags diferenciam maiúsculas de minúsculas
- Não use `aws :` como prefixo para chaves do. O uso deste prefixo é reservado para a AWS.

Adicionar, editar e excluir tags para distribuições

É possível usar o console do CloudFront para gerenciar tags para suas distribuições.

Para adicionar, editar ou excluir tags de uma distribuição

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.

2. Escolha o ID da distribuição que você deseja atualizar.
3. Escolha a guia Tags.
4. Selecione Gerenciar tags.
5. Na página Gerenciar tags é possível fazer o seguinte:
 - Para adicionar uma tag, insira uma chave e, se desejar, um valor para ela. Selecione Adicionar nova tag para adicionar mais tags.
 - Para editar uma etiqueta, modifique sua chave e/ou seu valor. Você pode excluir o valor de uma etiqueta, mas a chave é obrigatória.
 - Para excluir uma tag, escolha Remove (Remove).
6. Escolha Salvar alterações.

Marcação programática

Também é possível usar a API do CloudFront, a AWS Command Line Interface (AWS CLI), SDKs da AWS e o AWS Tools for Windows PowerShell para aplicar tags. Para obter mais informações, consulte os tópicos a seguir.

- Operações da API do CloudFront:
 - [ListTagsForResource](#)
 - [TagResource](#)
 - [UntagResource](#)
- AWS CLI: consulte [cloudfront](#) na Referência de comandos da AWS CLI.
- SDKs da AWS: consulte a documentação do SDK aplicável na página [Documentação da AWS](#).
- Tools for Windows PowerShell: consulte [Amazon CloudFront](#) na [Referência de Cmdlet do AWS Tools for PowerShell](#)

Excluir uma distribuição do

O procedimento a seguir exclui uma distribuição usando o console do CloudFront. Consulte informações sobre como excluir com a API do CloudFront em [DeleteDistribution](#) na Referência da API do Amazon CloudFront.

Se você precisar excluir uma distribuição com um OAC anexado a um bucket do S3, consulte [Excluir uma distribuição com um OAC anexado a um bucket do S3](#) para obter detalhes importantes.

Note

Lembre-se de que antes de excluir uma distribuição, você deverá desabilitá-la, o que requer permissão para atualizar a distribuição.

Se você desabilitar uma distribuição que tenha um nome de domínio alternativo associado a ela, o CloudFront deixará de aceitar o tráfego para esse nome de domínio (como `www.exemplo.com`), mesmo que outra distribuição tenha um nome de domínio alternativo com um caractere curinga (*) correspondente ao mesmo domínio (como `*.exemplo.com`).

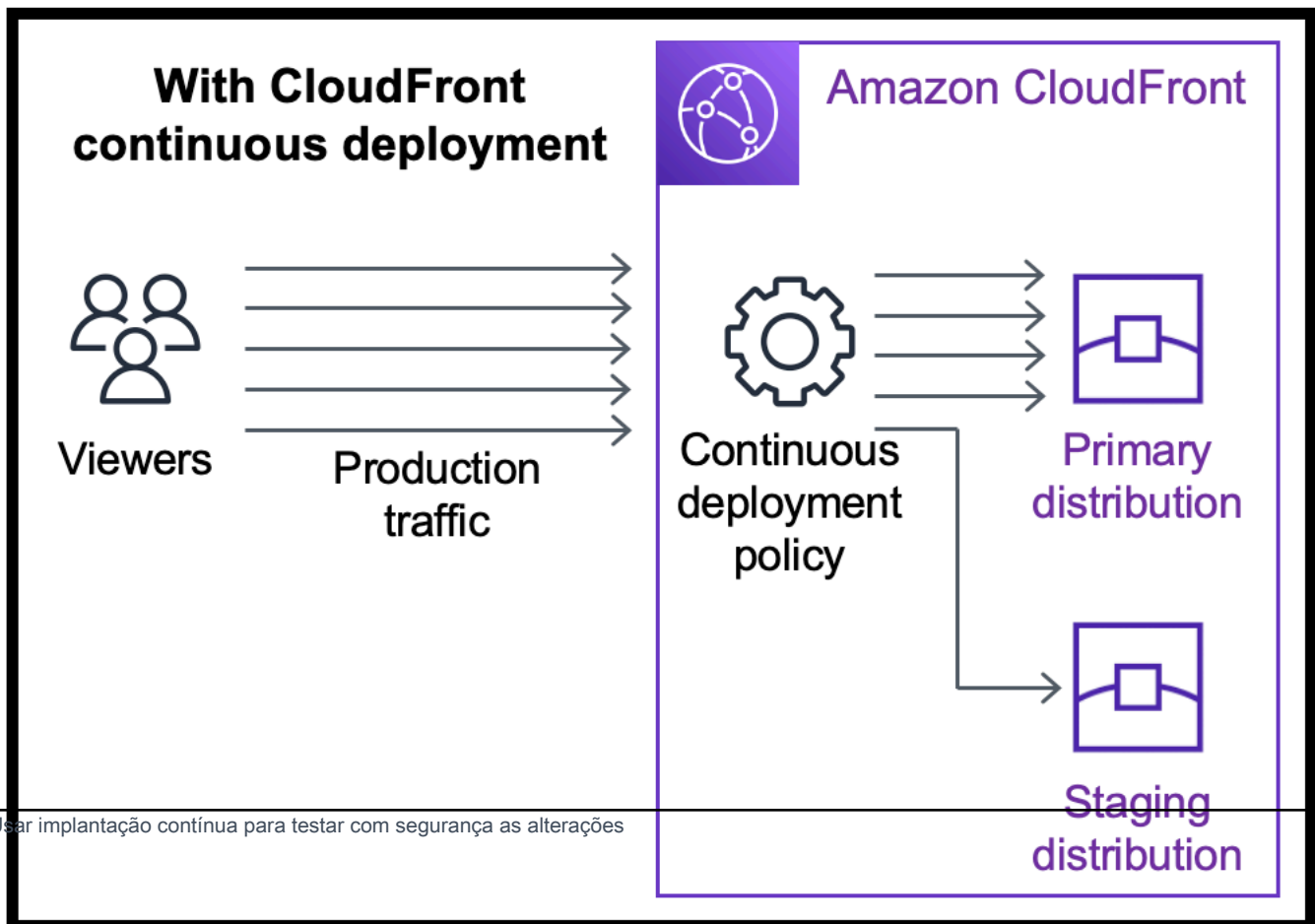
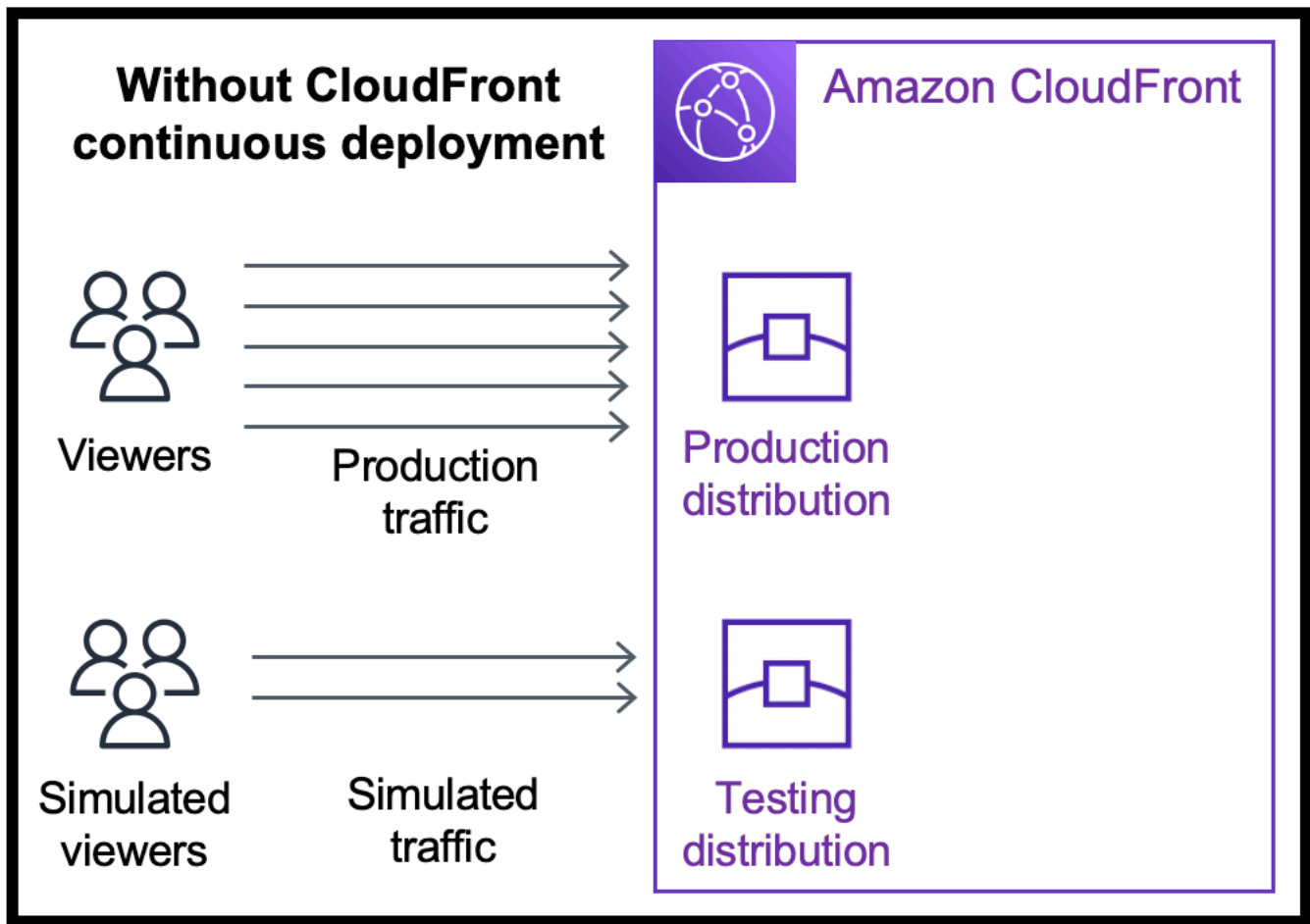
Para excluir uma distribuição do CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel à direita do console do CloudFront, encontre a distribuição que você deseja excluir.
 - Se a coluna Status exibir Disabled (Desabilitado), vá para a Etapa 6.
 - Se Status exibir Enabled (Habilitado), mas a distribuição ainda mostrar Deploying (Implantando) na coluna Last modified (Última modificação), aguarde a conclusão da implantação antes de continuar com a etapa 3.
3. No painel direito no console do CloudFront, marque a caixa de seleção da distribuição que você deseja excluir.
4. Selecione Disable (Desabilitar) para desabilitar a distribuição e selecione Yes, Disable (Sim, desabilitar) para confirmar. Em seguida, selecione Close (Fechar).
 - O valor da coluna Status é imediatamente alterado para Disabled (Desabilitado).
5. Espere até que o novo carimbo de data/hora apareça na coluna Last modified (Última modificação).
 - Pode levar alguns minutos para que o CloudFront propague a alteração para todos os locais da borda.
6. Marque a caixa de seleção da distribuição que você deseja excluir.
7. Escolha Delete (Excluir), Yes, Delete (Sim, excluir).
 - Se a opção Excluir não estiver disponível, isso significa que o CloudFront ainda está propagando a alteração para os locais da borda. Espere até que o novo carimbo de data e hora apareça na coluna Última modificação e, depois, repita as etapas 6 e 7.

Usar a implantação contínua do CloudFront para testar com segurança as alterações na configuração da CDN

Com a implantação contínua do Amazon CloudFront, você pode implantar com segurança as alterações em sua configuração de CDN realizando testes iniciais com um subconjunto do tráfego de produção. Você pode usar uma distribuição de preparação e uma política de implantação contínua para enviar tráfego de visualizadores reais (produção) para a nova configuração de CDN e validar se ela funciona conforme o esperado. Você pode monitorar o desempenho da nova configuração em tempo real e promover a nova configuração para encaminhar todo o tráfego pela distribuição primária quando estiver pronto.

O diagrama a seguir mostra a vantagem de usar a implantação contínua do CloudFront. Sem ela, você teria que testar as alterações na configuração de CDN com tráfego simulado. Com a implantação contínua, você pode testar as alterações com um subconjunto do tráfego de produção, depois promover as alterações para a distribuição primária quando estiver pronto.



Saiba mais sobre como trabalhar com implantação contínua nos tópicos a seguir.

Tópicos

- [Fluxo de trabalho de implantação contínua do CloudFront](#)
- [Trabalhar com uma distribuição de preparação e uma política de implantação contínua](#)
- [Monitorar uma distribuição de preparação](#)
- [Saiba como funciona a implantação contínua](#)
- [Cotas e outras considerações para implantação contínua](#)

Fluxo de trabalho de implantação contínua do CloudFront

O fluxo de trabalho de alto nível a seguir explica como testar e implantar com segurança as alterações de configuração com a implantação contínua do CloudFront.

1. Escolha a distribuição que deseja usar como distribuição primária. A distribuição primária é aquela que está lidando com o tráfego de produção no momento.
2. Da distribuição primária, crie uma distribuição de preparação. Uma distribuição de preparação começa como uma cópia da distribuição primária.
3. Crie uma configuração de tráfego dentro de uma política de implantação contínua e anexe-a à distribuição primária. Isso determina como o CloudFront direciona o tráfego para a distribuição de preparação. Para obter mais informações sobre o roteamento de solicitações para uma distribuição de preparação, consulte [the section called “Encaminhar solicitações para a distribuição de preparação”](#).
4. Atualize a configuração da distribuição de preparação. Para obter mais informações sobre as configurações que você pode atualizar, consulte [the section called “Atualizar distribuições principais e de preparação”](#).
5. Monitore a distribuição de preparação para determinar se as alterações de configuração funcionam conforme o esperado. Para obter mais informações sobre o monitoramento de uma distribuição de preparação, consulte [the section called “Monitorar uma distribuição de preparação”](#).

Ao monitorar a distribuição de preparação, você pode:

- Atualizar a configuração da distribuição de preparação novamente para continuar testando alterações de configuração.
- Atualizar a política de implantação contínua (configuração de tráfego) para enviar mais ou menos tráfego para a distribuição de preparação.

6. Quando estiver satisfeito com o desempenho da distribuição de preparação, promova a configuração da distribuição de preparação para a distribuição primária. Essa ação copiará a configuração da distribuição de preparação para a distribuição primária. Isso também desativará a política de implantação contínua, o que significa que o CloudFront encaminhará todo o tráfego para a distribuição primária.

Você pode criar uma automação que monitore o desempenho da distribuição de preparação (etapa 5) e promova a configuração automaticamente (etapa 6) quando determinados critérios forem atendidos.

Depois de promover uma configuração, você pode reutilizar a mesma distribuição de preparação na próxima vez que quiser testar uma alteração na configuração.

Para obter mais informações sobre como trabalhar com distribuições de preparação e políticas de implantação contínua no console do CloudFront, na AWS CLI ou na API do CloudFront, consulte a seção a seguir.

Trabalhar com uma distribuição de preparação e uma política de implantação contínua

É possível criar, atualizar e modificar distribuições de preparação e políticas de implantação contínua usando o console do CloudFront, a AWS Command Line Interface (AWS CLI) ou a API do CloudFront.

Criar com uma distribuição de preparação e uma política de implantação contínua

Os procedimentos a seguir mostram como criar uma distribuição de preparação com uma política de implantação contínua.

Console

É possível criar uma distribuição de preparação com uma política de implantação contínua usando o AWS Management Console.

Como criar uma distribuição de preparação e uma política de implantação contínua (console)

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Distribuições.

3. Escolha a distribuição que deseja usar como distribuição primária. A distribuição primária é aquela que lida atualmente com o tráfego de produção, aquela a partir da qual você criará a distribuição de preparação.
4. Na seção Continuous deployment (Implantação contínua), escolha Create staging distribution (Criar distribuição de preparação). Isso abre o assistente Create staging distribution (Criar distribuição de preparação).
5. No assistente Create staging distribution (Criar distribuição de preparação), faça o seguinte:
 - a. (Opcional) Digite uma descrição para a distribuição de preparação.
 - b. Escolha Próximo.
 - c. Modifique a configuração da distribuição de preparação. Para obter mais informações sobre as configurações que você pode atualizar, consulte [the section called “Atualizar distribuições principais e de preparação”](#).

Quando terminar de modificar a configuração da distribuição de preparação, escolha Next (Avançar).

- d. Use o console para especificar a Traffic configuration (Configuração de tráfego). Isso determina como o CloudFront direciona o tráfego para a distribuição de preparação. (O CloudFront armazena a configuração de tráfego em uma política de implantação contínua.)

Para obter mais informações sobre as opções para uma Traffic configuration (Configuração de tráfego), consulte [the section called “Encaminhar solicitações para a distribuição de preparação”](#).

Quando você concluir a Traffic configuration (Configuração de tráfego), escolha Next (Avançar).

- e. Revise a configuração da distribuição de preparação, incluindo a configuração de tráfego, depois escolha Create staging distribution (Criar distribuição de preparação).

Quando você concluir o assistente Create staging distribution (Criar distribuição de preparação) no console do CloudFront, o CloudFront fará o seguinte:

- Criará uma distribuição de preparação com as configurações que você especificou (na etapa 5c)

- Criará uma política de implantação contínua com a configuração de tráfego que você especificou (na etapa 5d)
- Anexará a política de implantação contínua à distribuição primária que você usou para criar a distribuição de preparação

Quando a configuração da distribuição primária, com a política de implantação contínua anexada, é implantada em locais da borda, o CloudFront começa a enviar a parte especificada do tráfego para a distribuição de preparação com base na configuração de tráfego.

CLI

Para criar uma distribuição de preparação e uma política de implantação contínua com a AWS CLI, use os procedimentos a seguir.

Como criar uma distribuição de preparação (CLI)

1. Use os comandos `aws cloudfront get-distribution` e `grep` juntos para obter o valor de ETag da distribuição que você deseja usar como distribuição primária. A distribuição primária é aquela que lida atualmente com o tráfego de produção, da qual você criará a distribuição de preparação.

O seguinte comando mostra um exemplo. No exemplo a seguir, substitua *primary_distribution_ID* pelo ID da distribuição primária.

```
aws cloudfront get-distribution --id primary_distribution_ID | grep 'ETag'
```

Copie o valor de ETag porque ele será necessário na próxima etapa.

2. Use o comando `aws cloudfront copy-distribution` para criar uma distribuição de preparação. O exemplo de comando a seguir usa caracteres de escape (`\`) e quebras de linha para facilitar a leitura, mas você deve omiti-los do comando. No exemplo de comando a seguir:
 - Substitua *primary_distribution_ID* pelo ID da distribuição primária.
 - Substitua *primary_distribution_ETag* pelo valor de ETag da distribuição primária (que você obteve na etapa anterior).
 - (Opcional) Substitua *CLI_example* pelo ID de referência do chamador desejado.


```
aws cloudfront copy-distribution --primary-distribution-id primary_distribution_ID \  
                                --if-match primary_distribution_ETag \  
                                --staging \  
                                --caller-reference 'CLI_example'
```

A saída do comando mostra informações sobre a distribuição de preparação e sua configuração. Copie o nome de domínio do CloudFront da distribuição de preparação porque ele será necessário na próxima etapa.

Como criar uma política de implantação contínua (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `continuous-deployment-policy.yaml` que contém todos os parâmetros de entrada para o comando `create-continuous-deployment-policy`. O comando a seguir usa caracteres de escape (`\`) e quebras de linha para facilitar a leitura, mas você deve omiti-los do comando.

```
aws cloudfront create-continuous-deployment-policy --generate-cli-skeleton yml-  
input \  
                                                    > continuous-deployment-  
policy.yaml
```

2. Abra o arquivo chamado `continuous-deployment-policy.yaml` que você acabou de criar. Edite o arquivo para especificar as configurações de política de implantação contínua desejadas e salve o arquivo. Ao editar o arquivo:
 - Na seção `StagingDistributionDnsNames`, selecione:
 - Altere o valor de `Quantity` para 1.
 - Em `Items`, cole o nome de domínio do CloudFront da distribuição de preparação (que você salvou em uma etapa anterior).
 - Na seção `TrafficConfig`, selecione:
 - Escolha um `Type`: `SingleWeight` ou `SingleHeader`.

- Remova as configurações do outro tipo. Por exemplo, se você quiser uma configuração de tráfego baseada em peso, defina Type como SingleWeight, depois remova as configurações de SingleHeaderConfig.
- Para usar uma configuração de tráfego baseada em peso, defina o valor de Weight como um número decimal entre .01 (1%) e .15 (15%).

Para obter mais informações sobre essas opções em TrafficConfig, consulte [the section called “Encaminhar solicitações para a distribuição de preparação”](#) e [the section called “Persistência da sessão para configurações baseadas em peso”](#).

3. Use o seguinte comando para criar a política de implantação contínua usando parâmetros de entrada do arquivo continuous-deployment-policy.yaml.

```
aws cloudfront create-continuous-deployment-policy --cli-input-yaml file://
continuous-deployment-policy.yaml
```

Copie o valor de Id na saída do comando. Esse é o ID da política de implantação contínua, que será necessário na próxima etapa.

Como anexar uma política de implantação contínua a uma distribuição primária (CLI com arquivo de entrada)

1. Use o comando a seguir para salvar a configuração da distribuição primária em um arquivo chamado primary-distribution.yaml. Substitua *primary_distribution_ID* pelo ID da distribuição primária.

```
aws cloudfront get-distribution-config --id primary_distribution_ID --output
yaml > primary-distribution.yaml
```

2. Abra o arquivo chamado primary-distribution.yaml que você acabou de criar. Edite o arquivo fazendo as seguintes alterações:
 - Cole o ID da política de implantação contínua (que você copiou em uma etapa anterior) no campo ContinuousDeploymentPolicyId.
 - Renomeie o campo ETag para IfMatch, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

3. Use o comando a seguir para atualizar a distribuição primária para utilizar a política de implantação contínua. Substitua *primary_distribution_ID* pelo ID da distribuição primária.

```
aws cloudfront update-distribution --id primary_distribution_ID --cli-input-yaml  
file://primary-distribution.yaml
```

Quando a configuração da distribuição primária, com a política de implantação contínua anexada, é implantada em locais da borda, o CloudFront começa a enviar a parte especificada do tráfego para a distribuição de preparação com base na configuração de tráfego.

API

Para criar uma distribuição de preparação e uma política de implantação contínua com a API do CloudFront, use as seguintes operações de API:

- [CopyDistribution](#)
- [CreateContinuousDeploymentPolicy](#)

Para obter mais informações sobre os campos especificados nessas chamadas de API, consulte o seguinte:

- [the section called “Encaminhar solicitações para a distribuição de preparação”](#)
- [the section called “Persistência da sessão para configurações baseadas em peso”](#)
- A documentação de referência da API do AWS SDK ou de outro cliente de API

Depois de criar uma distribuição de preparação e uma política de implantação contínua, use [UpdateDistribution](#) (na distribuição primária) para anexar a política de implantação contínua à distribuição primária.

Atualizar uma distribuição de preparação

Os procedimentos a seguir mostram como atualizar uma distribuição de preparação com uma política de implantação contínua.

Console

É possível atualizar determinadas configurações para as distribuições principal e de preparação. Para ter mais informações, consulte [Atualizar distribuições principais e de preparação](#).

Como atualizar uma distribuição de preparação (console)

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Distribuições.
3. Escolha a distribuição primária. É a distribuição que lida atualmente com o tráfego de produção, aquela da qual você criou a distribuição de preparação.
4. Escolha View staging distribution (Exibir distribuição de preparação).
5. Use o console para modificar a configuração da distribuição de preparação. Para obter mais informações sobre as configurações que você pode atualizar, consulte [the section called “Atualizar distribuições principais e de preparação”](#).

Assim que a configuração da distribuição de preparação é implantada em locais da borda, ela entra em vigor para o tráfego de entrada que é roteado para a distribuição de preparação.

CLI

Como atualizar uma distribuição de preparação (CLI com arquivo de entrada)

1. Use o comando a seguir para salvar a configuração da distribuição de preparação em um arquivo chamado `staging-distribution.yaml`. Substitua *staging_distribution_ID* pelo ID da distribuição de preparação.

```
aws cloudfront get-distribution-config --id staging_distribution_ID --output  
yaml > staging-distribution.yaml
```

2. Abra o arquivo chamado `staging-distribution.yaml` que você acabou de criar. Edite o arquivo fazendo as seguintes alterações:

- Modifique a configuração da distribuição de preparação. Para obter mais informações sobre as configurações que você pode atualizar, consulte [the section called “Atualizar distribuições principais e de preparação”](#).
- Renomeie o campo ETag para IfMatch, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

3. Use o comando a seguir para atualizar a configuração da distribuição de preparação. Substitua *staging_distribution_ID* pelo ID da distribuição de preparação.

```
aws cloudfront update-distribution --id staging_distribution_ID --cli-input-yaml  
file://staging-distribution.yaml
```

Assim que a configuração da distribuição de preparação é implantada em locais da borda, ela entra em vigor para o tráfego de entrada que é roteado para a distribuição de preparação.

API

Para atualizar a configuração de uma distribuição de preparação, use [UpdateDistribution](#) (na distribuição de preparação) para modificar a configuração da distribuição de preparação. Para obter mais informações sobre as configurações que você pode atualizar, consulte [the section called “Atualizar distribuições principais e de preparação”](#).

Atualizar uma política de implantação contínua

Os procedimentos a seguir mostram como atualizar uma política de implantação contínua.

Console

É possível atualizar a configuração de tráfego de sua distribuição atualizando a política de implantação contínua.

Como atualizar uma política de implantação contínua (console)

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Distribuições.

3. Escolha a distribuição primária. É a distribuição que lida atualmente com o tráfego de produção, aquela a partir da qual você criou a distribuição de preparação.
4. Na seção Continuous deployment (Implantação contínua), escolha Edit policy (Editar política).
5. Modifique a configuração de tráfego em uma política de implantação contínua. Ao concluir, escolha Save changes.

Quando a configuração da distribuição primária, com a política de implantação contínua atualizada, é implantada em locais da borda, o CloudFront começa a enviar tráfego para a distribuição de preparação com base na configuração de tráfego atualizada.

CLI

Como atualizar uma política de implantação contínua (CLI com arquivo de entrada)

1. Use o comando a seguir para salvar a configuração da política de implantação contínua em um arquivo chamado `continuous-deployment-policy.yaml`. Substitua `continuous_deployment_policy_ID` pelo ID da política de implantação contínua. O comando a seguir usa caracteres de escape (`\`) e quebras de linha para facilitar a leitura, mas você deve omiti-los do comando.

```
aws cloudfront get-continuous-deployment-policy-config --  
id continuous_deployment_policy_ID \  
continuous-deployment-policy.yaml --output yaml >
```

2. Abra o arquivo chamado `continuous-deployment-policy.yaml` que você acabou de criar. Edite o arquivo fazendo as seguintes alterações:
 - Modifique a configuração da política de implantação contínua como quiser. Por exemplo, você pode deixar de usar uma configuração de tráfego baseada em cabeçalho para passar a usar uma configuração de tráfego baseada em peso, ou você pode alterar a porcentagem de tráfego (peso) para uma configuração com base em peso. Para ter mais informações, consulte [the section called “Encaminhar solicitações para a distribuição de preparação”](#) e [the section called “Persistência da sessão para configurações baseadas em peso”](#).
 - Renomeie o campo ETag para IfMatch, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

3. Use o comando a seguir para atualizar a política de implantação contínua. Substitua *continuous_deployment_policy_ID* pelo ID da política de implantação contínua. O comando a seguir usa caracteres de escape (\) e quebras de linha para facilitar a leitura, mas você deve omiti-los do comando.

```
aws cloudfront update-continuous-deployment-policy --  
id continuous_deployment_policy_ID \  
continuous-deployment-policy.yaml --cli-input-yaml file://
```

Quando a configuração da distribuição primária, com a política de implantação contínua atualizada, é implantada em locais da borda, o CloudFront começa a enviar tráfego para a distribuição de preparação com base na configuração de tráfego atualizada.

API

Para atualizar uma política de implantação contínua, use [UpdateContinuousDeploymentPolicy](#).

Promover a configuração de uma distribuição de preparação

Os procedimentos a seguir mostram como promover uma configuração de distribuição de preparação.

Console

Quando você promove uma distribuição de preparação, o CloudFront copia a configuração da distribuição de preparação para a distribuição primária. O CloudFront também desativará a política de implantação contínua e encaminhará todo o tráfego para a distribuição primária.

Depois de promover uma configuração, você pode reutilizar a mesma distribuição de preparação na próxima vez que quiser testar uma alteração na configuração.

Como promover a configuração de uma distribuição de preparação (console)

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Distribuições.

3. Escolha a distribuição primária. É a distribuição que lida atualmente com o tráfego de produção, aquela a partir da qual você criou a distribuição de preparação.
4. Na seção Continuous deployment (Implantação contínua), escolha Promote (Promover).
5. Digite **confirm**, depois escolha Promote (Promover).

CLI

Quando você promove uma distribuição de preparação, o CloudFront copia a configuração da distribuição de preparação para a distribuição primária. O CloudFront também desativará a política de implantação contínua e encaminhará todo o tráfego para a distribuição primária.

Depois de promover uma configuração, você pode reutilizar a mesma distribuição de preparação na próxima vez que quiser testar uma alteração na configuração.

Como promover a configuração de uma distribuição de preparação (CLI)

- Use o comando `aws cloudfront update-distribution-with-staging-config` para promover a configuração da distribuição de preparação para a distribuição primária. O exemplo de comando a seguir usa caracteres de escape (\) e quebras de linha para facilitar a leitura, mas você deve omiti-los do comando. No exemplo de comando a seguir:
 - Substitua *primary_distribution_ID* pelo ID da distribuição primária.
 - Substitua *staging_distribution_ID* pelo ID da distribuição de preparação.
 - Substitua *primary_distribution_ETag* e *staging_distribution_ETag* pelos valores de ETag das distribuições primária e de preparação. Certifique-se de que o valor da distribuição primária seja o primeiro, conforme mostrado no exemplo.

```
aws cloudfront update-distribution-with-staging-config --
id primary_distribution_ID \
                                                    --staging-distribution-
id staging_distribution_ID \
                                                    --if-match
'primary_distribution_ETag, staging_distribution_ETag'
```


API

Para promover a configuração de uma distribuição de preparação para a distribuição primária, use [UpdateDistributionWithStagingConfig](#).

Monitorar uma distribuição de preparação

Para monitorar o desempenho de uma distribuição de preparação, você pode usar as mesmas [métricas, logs e relatórios](#) que o CloudFront fornece para todas as distribuições. Por exemplo:

- Você pode visualizar as [métricas de distribuição padrão do CloudFront](#) (como total de solicitações e taxa de erros) no console do CloudFront e [ativar métricas adicionais](#) (como taxa de acertos de cache e taxa de erros por código de status) por um custo adicional. Você também pode criar alarmes com base nessas métricas.
- Você pode visualizar [logs padrão](#) e [logs em tempo real](#) para obter informações detalhadas sobre as solicitações recebidas pela distribuição de preparação. Os logs padrão contêm os dois campos a seguir que ajudam a identificar a distribuição primária para a qual a solicitação foi originalmente enviada antes do CloudFront encaminhá-la para a distribuição de preparação: `primary-distribution-id` e `primary-distribution-dns-name`.
- Você pode visualizar e baixar [relatórios](#) no console do CloudFront, por exemplo, o relatório de estatísticas de cache.

Saiba como funciona a implantação contínua

Os tópicos a seguir explicam como funciona a implantação contínua do CloudFront.

Tópicos

- [Encaminhar solicitações para a distribuição de preparação](#)
- [Persistência da sessão para configurações baseadas em peso](#)
- [Atualizar distribuições principais e de preparação](#)
- [Distribuições primária e de preparação não compartilham cache](#)

Encaminhar solicitações para a distribuição de preparação

Se você usar a implantação contínua do CloudFront, não precisará alterar nada nas solicitações do visualizador. Os visualizadores não podem enviar solicitações diretamente para uma distribuição de

preparação usando um nome DNS, endereço IP ou CNAME. Em vez disso, os visualizadores enviam solicitações para a distribuição primária (produção), e o CloudFront encaminha algumas dessas solicitações para a distribuição de preparação com base nas configurações de tráfego na política de implantação contínua. Há dois tipos de configurações de tráfego:

Baseada em peso

Uma configuração baseada em peso direciona a porcentagem especificada de solicitações do visualizador para a distribuição de preparação. Ao usar uma configuração baseada em peso, você também pode ativar a permanência da sessão, o que ajuda a garantir que o CloudFront trate as solicitações do mesmo visualizador como parte de uma única sessão. Para ter mais informações, consulte [the section called “Persistência da sessão para configurações baseadas em peso”](#).

Baseado em cabeçalho

Uma configuração baseada em cabeçalho direciona as solicitações para a distribuição de preparação quando a solicitação do visualizador contém um cabeçalho HTTP específico (você especifica o cabeçalho e o valor). As solicitações que não contêm o cabeçalho e o valor especificados são encaminhadas para a distribuição primária. Essa configuração é útil para testes locais ou quando você tem controle sobre as solicitações do visualizador.

Note

Os cabeçalhos roteados para sua distribuição de preparação devem conter o prefixo `aws-cf-cd-`.

Persistência da sessão para configurações baseadas em peso

Ao usar uma configuração baseada em peso para direcionar tráfego a uma distribuição de preparação, você também pode ativar a permanência da sessão, o que ajuda a garantir que o CloudFront trate as solicitações do mesmo visualizador como parte de uma única sessão. Quando você ativa a permanência da sessão, o CloudFront define um cookie para que todas as solicitações do mesmo visualizador em uma única sessão sejam atendidas por uma distribuição, primária ou de preparação.

Ao ativar a permanência da sessão, você também pode especificar o tempo ocioso. Se o visualizador ficar inativo (não enviar solicitações) durante esse período, a sessão expirará e o CloudFront tratará

as solicitações futuras desse visualizador como uma nova sessão. Especifique o tempo ocioso como um número em segundos, de 300 (cinco minutos) a 3.600 (uma hora).

Nos casos a seguir, o CloudFront redefine todas as sessões (até mesmo as ativas) e considera todas as solicitações como uma nova sessão:

- Você desativa ou ativa a política de implantação contínua
- Você desativa ou ativa a configuração de permanência da sessão

Atualizar distribuições principais e de preparação

Quando uma distribuição primária tem uma política de implantação contínua anexada, as seguintes alterações de configuração estão disponíveis para distribuições primárias e de preparação:

- Todas as configurações de comportamento do cache, incluindo o comportamento padrão do cache
- Todas as configurações de origem (origens e grupos de origens)
- Respostas de erro personalizadas (páginas de erro)
- Restrições geográficas
- Objeto raiz padrão
- Configurações de registro em log
- Descrição (comentário)

Você também pode atualizar recursos externos referenciados na configuração de uma distribuição, como uma política de cache, uma política de cabeçalhos de resposta, uma função do CloudFront ou uma função do Lambda@Edge.

Distribuições primária e de preparação não compartilham cache

As distribuições primária e de preparação não compartilham um cache. Quando o CloudFront envia a primeira solicitação para uma distribuição de preparação, seu cache fica vazio. À medida que as solicitações chegam à distribuição de preparação, as respostas começam a ser armazenadas em cache (se configurada para isso).

Cotas e outras considerações para implantação contínua

A implantação contínua do CloudFront está sujeita às seguintes cotas e outras considerações.

Cotas

- Número máximo de distribuições de preparação por Conta da AWS: 20
- Número máximo de políticas de implantação contínua por Conta da AWS: 20
- Porcentagem máxima de tráfego que você pode enviar para uma distribuição de preparação em uma configuração baseada em peso: 15%
- Valores mínimo e máximo para o tempo ocioso da permanência da sessão: 300–3.600 segundos

Para ter mais informações, consulte [Cotas](#).

Note

Ao usar a implantação contínua e sua distribuição primária estiver configurada com OAC para acesso ao bucket do S3, atualize sua política do bucket do S3 para permitir o acesso à distribuição de preparação. Para obter exemplos de políticas de bucket do S3, consulte [the section called “Conceder permissão ao controle de acesso à origem para acessar o bucket do S3”](#).

ACLs da Web do AWS WAF

Se você habilitar a distribuição contínua para a distribuição, as seguintes considerações se aplicarão ao AWS WAF:

- Não é possível associar uma lista de controle de acesso (ACL) da web do AWS WAF à distribuição pela primeira vez.
- Também não é possível desassociar uma ACL da web do AWS WAF da distribuição.

Antes de realizar as tarefas anteriores, é necessário excluir a política de implantação contínua para a distribuição da produção. Isso também exclui a distribuição de preparação. Para ter mais informações, consulte [Usar proteções do AWS WAF](#).

Casos em que o CloudFront envia todas as solicitações para a distribuição primária

Em certos casos, como períodos de alta utilização de recursos, o CloudFront pode enviar todas as solicitações para a distribuição primária, independentemente do que esteja especificado na política de implantação contínua.

O CloudFront envia todas as solicitações para a distribuição primária durante os horários de pico de tráfego, independentemente do que esteja especificado na política de implantação contínua. O pico de tráfego se refere ao tráfego no serviço CloudFront, e não ao tráfego na sua distribuição.

HTTP/3

Você não pode usar a implantação contínua com uma distribuição que seja compatível com HTTP/3.

Usar várias origens com distribuições do CloudFront

Ao criar uma distribuição, especifique a origem onde o CloudFront enviará solicitações para os arquivos. É possível usar vários tipos diferentes de origens com o CloudFront. Por exemplo, você pode usar um bucket do Amazon S3, um contêiner do MediaStore, um canal do MediaPackage, um Application Load Balancer ou um URL da função do AWS Lambda.

Tópicos

- [Usar um bucket do Amazon S3](#)
- [Usar um contêiner do MediaStore ou um canal do MediaPackage](#)
- [Usar um Application Load Balancer](#)
- [Usar um URL da função do Lambda](#)
- [Usar o Amazon EC2 \(ou outra origem personalizada\)](#)
- [Usar grupos de origem do CloudFront](#)

Usar um bucket do Amazon S3

Os tópicos a seguir descrevem as diferentes maneiras de usar um bucket do Amazon S3 como a origem de uma distribuição do CloudFront.

Tópicos

- [Usar um bucket padrão do Amazon S3](#)
- [Usar o Amazon S3 Object Lambda](#)
- [Usar Pontos de Acesso Amazon S3](#)
- [Usar um bucket do Amazon S3 configurado como um endpoint do site](#)
- [Adicionar o CloudFront a um bucket existente do Amazon S3](#)

- [Mover um bucket do Amazon S3 para uma Região da AWS diferente](#)

Usar um bucket padrão do Amazon S3

Ao usar o Amazon S3 como origem para a distribuição, coloque os objetos que deseja que o CloudFront entregue em um bucket do Amazon S3. É possível usar qualquer método compatível com o Amazon S3 para colocar os objetos no Amazon S3. Por exemplo, você pode usar o console ou a API do Amazon S3 ou uma ferramenta de terceiros. É possível criar uma hierarquia no bucket para armazenar os objetos, da mesma forma que você faria com qualquer outro bucket padrão do Amazon S3.

Usar um bucket do Amazon S3 existente como seu servidor de origem do CloudFront não altera o bucket. Ainda é possível usá-lo normalmente como você faria para armazenar e acessar objetos do Amazon S3 no preço padrão do Amazon S3. Você será cobrado o valor normal do Amazon S3 para armazenar os objetos no bucket. Para obter mais informações sobre as cobranças para usar o CloudFront, consulte [Preço do Amazon CloudFront](#). Para obter mais informações sobre como usar o CloudFront com um bucket do S3 existente, consulte [the section called “Adicionar o CloudFront a um bucket existente do Amazon S3”](#).

Important

Para que seu bucket funcione com o CloudFront, o nome deve estar de acordo com os requisitos de nomenclatura do DNS. Para obter mais informações, acesse [Regras de atribuição de nomes de buckets](#) no Manual do usuário do Amazon Simple Storage Service.

Ao especificar um bucket do Amazon S3 como uma origem do CloudFront, recomendamos que você use o seguinte formato:

bucket-name.s3.*region*.amazonaws.com

Ao especificar o nome do bucket nesse formato, é possível usar os seguintes recursos do CloudFront:

- Configure o CloudFront para se comunicar com o bucket do Amazon S3 usando SSL/TLS. Para ter mais informações, consulte [the section called “Usar HTTPS com o CloudFront”](#).
- Use um controle de acesso à origem para exigir que os visualizadores acessem seu conteúdo usando URLs do CloudFront, não do Amazon S3. Para ter mais informações, consulte [the section called “Restringir o acesso a uma origem do Amazon Simple Storage Service”](#).

- Atualize o conteúdo do seu bucket enviando solicitações POST e PUT para o CloudFront. Para obter mais informações, consulte [the section called “Métodos HTTP”](#) no tópico [the section called “Como o CloudFront processa e encaminha solicitações à sua origem do Amazon S3”](#).

Não especifique o bucket usando os seguintes formatos:

- O estilo de caminho do Amazon S3: `s3.amazonaws.com/bucket-name`
- O CNAME do Amazon S3

Usar o Amazon S3 Object Lambda

Quando você [cria um ponto de acesso do Object Lambda](#), o Amazon S3 gera automaticamente um alias exclusivo para seu ponto de acesso do Object Lambda. É possível [usar esse alias](#) em vez de um nome de bucket do Amazon S3 como uma origem de sua distribuição do CloudFront.

Ao usar um alias de ponto de acesso do Object Lambda como uma origem do CloudFront, recomendamos que você use o seguinte formato:

`alias.s3.region.amazonaws.com`

Para obter informações sobre como encontrar o *alias*, consulte [Como usar um alias em estilo de bucket para seu ponto de acesso do S3 Object Lambda](#) no Guia do usuário do Amazon S3.

Important

Ao usar um ponto de acesso do Object Lambda como origem para o CloudFront, você deverá usar o [controle de acesso de origem](#).

Para ver um exemplo de caso de uso, consulte [Usar o Amazon S3 Object Lambda com o Amazon CloudFront para personalizar o conteúdo para usuários finais](#).

O CloudFront trata uma origem de ponto de acesso do Object Lambda da mesma forma que [uma origem de bucket padrão do Amazon S3](#).

As quatro permissões a seguir devem ser configuradas ao usar o Amazon S3 Object Lambda como origem para sua distribuição.

Object Lambda Access Point

Para adicionar permissões para o ponto de acesso Object Lambda

1. Faça login no AWS Management Console e abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. No painel de navegação à esquerda, escolha Pontos de Acesso Object Lambda.
3. Escolha o Ponto de Acesso Object Lambda que deseja usar.
4. Escolha a aba Permissões.
5. Escolha Editar na seção Política de ponto de acesso do Object Lambda.
6. Cole a política a seguir no campo Política.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": "s3-object-lambda:Get*",
      "Resource": "arn:aws:s3-object-lambda:region:AWS-account-ID:accesspoint/Object-Lambda-Access-Point-name",
      "Condition": {
        "StringEquals": {
          "aws:SourceArn": "arn:aws:cloudfront::AWS-account-ID:distribution/CloudFront-distribution-ID"
        }
      }
    }
  ]
}
```

7. Escolha Salvar alterações.

Amazon S3 Access Point

Para adicionar permissões para o Ponto de Acesso Amazon S3

1. Faça login no AWS Management Console e abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. No painel de navegação, escolha Pontos de acesso.
3. Escolha o Ponto de Acesso Amazon S3 que deseja usar.
4. Escolha a aba Permissões.
5. Escolha Editar na seção Política de ponto de acesso.
6. Cole a política a seguir no campo Política.

```
{
  "Version": "2012-10-17",
  "Id": "default",
  "Statement": [
    {
      "Sid": "s3objlambda",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:region:AWS-account-ID:accesspoint/Access-Point-  
name",
        "arn:aws:s3:region:AWS-account-ID:accesspoint/Access-Point-name/  
object/*"
      ],
      "Condition": {
        "ForAnyValue:StringEquals": {
          "aws:CalledVia": "s3-object-lambda.amazonaws.com"
        }
      }
    }
  ]
}
```

7. Escolha Salvar.

Amazon S3 bucket

Como adicionar permissões para o bucket do Amazon S3

1. Faça login no AWS Management Console e abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. No painel de navegação, escolha Buckets.
3. Escolha o bucket do Amazon S3 que deseja usar.
4. Escolha a aba Permissões.
5. Na seção Política do bucket, escolha Editar.
6. Cole a política a seguir no campo Política.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "*",
      "Resource": [
        "arn:aws:s3:::bucket-name",
        "arn:aws:s3:::bucket-name/*"
      ],
      "Condition": {
        "StringEquals": {
          "s3:DataAccessPointAccount": "AWS-account-ID"
        }
      }
    }
  ]
}
```

7. Escolha Salvar alterações.

AWS Lambda function

Como adicionar permissões para a função do Lambda

1. Faça login no AWS Management Console e abra o console do AWS Lambda em <https://console.aws.amazon.com/lambda/>.
2. Selecione Funções no painel de navegação.
3. Escolha a função do AWS Lambda que deseja publicar.
4. Escolha a guia Configuração e, depois, Permissões.
5. Escolha Adicionar permissões na seção Declarações de políticas baseadas em recursos.
6. Selecione Conta da AWS.
7. Insira um nome para o ID da declaração.
8. Insira `ccloudfront.amazonaws.com` em Entidade principal.
9. Escolha `lambda:InvokeFunction` no menu suspenso Ação.
10. Escolha Salvar.

Usar Pontos de Acesso Amazon S3

Quando você [usa um ponto de acesso do S3](#), o Amazon S3 gera um alias exclusivo para você. É possível usar esse alias em vez de um nome de bucket do Amazon S3 como uma origem de sua distribuição do CloudFront.

Ao usar um alias de Ponto de Acesso Amazon S3 como uma origem do CloudFront, recomendamos que você use o seguinte formato:

alias.s3.*region*.amazonaws.com

Para ter mais informações sobre como encontrar o *alias*, consulte [Usar um alias em estilo de bucket para seu ponto de acesso do S3](#) no Guia do usuário do Amazon S3.

Important

Ao usar um Ponto de Acesso Amazon S3 como origem para o CloudFront, você deverá usar o [controle de acesso à origem](#).

O CloudFront trata uma origem de Ponto de Acesso Amazon S3 da mesma forma que [uma origem de bucket padrão do Amazon S3](#).

As duas permissões a seguir devem ser configuradas ao usar o Amazon S3 Object Lambda como origem para sua distribuição.

Amazon S3 Access Point

Para adicionar permissões para o Ponto de Acesso Amazon S3

1. Faça login no AWS Management Console e abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. No painel de navegação, escolha Pontos de acesso.
3. Escolha o Ponto de Acesso Amazon S3 que deseja usar.
4. Escolha a aba Permissões.
5. Escolha Editar na seção Política de ponto de acesso.
6. Cole a política a seguir no campo Política.

```
{
  "Version": "2012-10-17",
  "Id": "default",
  "Statement": [
    {
      "Sid": "s3objlambda",
      "Effect": "Allow",
      "Principal": {"Service": "cloudfront.amazonaws.com"},
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:region:AWS-account-ID:accesspoint/Access-Point-name",
        "arn:aws:s3:region:AWS-account-ID:accesspoint/Access-Point-name/object/*"
      ],
      "Condition": {
        "StringEquals": {"aws:SourceArn": "arn:aws:cloudfront::AWS-account-ID:distribution/CloudFront-distribution-ID"}
      }
    }
  ]
}
```

7. Escolha Salvar.

Amazon S3 bucket

Como adicionar permissões para o bucket do Amazon S3

1. Faça login no AWS Management Console e abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. No painel de navegação, escolha Buckets.
3. Escolha o bucket do Amazon S3 que deseja usar.
4. Escolha a aba Permissões.
5. Na seção Política do bucket, escolha Editar.
6. Cole a política a seguir no campo Política.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "*",
      "Resource": [
        "arn:aws:s3:::bucket-name",
        "arn:aws:s3:::bucket-name/*"
      ],
      "Condition": {
        "StringEquals": {
          "s3:DataAccessPointAccount": "AWS-account-ID"
        }
      }
    }
  ]
}
```

7. Escolha Salvar alterações.

Usar um bucket do Amazon S3 configurado como um endpoint do site

É possível usar um bucket do Amazon S3 configurado como um endpoint de site como uma origem personalizada com o CloudFront. Ao configurar a distribuição do CloudFront, para a origem, insira o endpoint de hospedagem de site estático do Amazon S3 para seu bucket. Esse valor é exibido no [console do Amazon S3](#), na guia Properties (Propriedades), no painel Static website hosting (Hospedagem de site estático). Por exemplo:

```
http://bucket-name.s3-website-region.amazonaws.com
```

Para obter mais informações sobre como especificar endpoints de site estáticos do Amazon S3, consulte [Endpoints de site](#) no Manual do usuário do Amazon Simple Storage Service.

Ao especificar o nome do bucket nesse formato como sua origem, é possível usar redirecionamentos e documentos de erro personalizados do Amazon S3. Para obter mais informações, consulte [Configurar um documento de erro personalizado](#) e [Configurar um redirecionamento](#) no Guia do usuário do Amazon Simple Storage Service. (O CloudFront também fornece páginas de erro personalizadas. Para obter mais informações, consulte [the section called “Criar uma página de erro personalizada para códigos de status HTTP específicos”](#).)

Usar um bucket do Amazon S3 como servidor de origem do CloudFront não o altera o bucket de forma alguma. Ainda é possível usá-lo como você faria normalmente e o Amazon S3 será cobrado pelos valores normais. Para obter mais informações sobre as cobranças para usar o CloudFront, consulte [Preço do Amazon CloudFront](#).

Note

Se usar a API do CloudFront para criar sua distribuição com um bucket do Amazon S3 que está configurado como o endpoint de um site, você deverá configurá-lo usando `CustomOriginConfig`, mesmo que o site esteja hospedado em um bucket do Amazon S3. Para mais informações sobre a criação de distribuições usando a API do CloudFront, consulte [CreateDistribution](#) na Referência da API do Amazon CloudFront.

Adicionar o CloudFront a um bucket existente do Amazon S3

Se armazenar os objetos em um bucket do Amazon S3, você poderá fazer com que os usuários os obtenham diretamente no S3 ou poderá configurar o CloudFront para obtê-los no S3 e distribuí-los para seus usuários. Usar o CloudFront pode ser mais econômico se os seus usuários acessarem

seus objetos com frequência. Com um uso maior, o preço da transferência de dados do CloudFront é menor que o do Amazon S3. Além disso, os downloads são mais rápidos com o CloudFront do que só com o Amazon S3 porque seus objetos são armazenados mais próximos dos usuários.

Note

Se quiser que o CloudFront respeite as configurações de compartilhamento de recursos entre origens do Amazon S3, configure o CloudFront para encaminhar o cabeçalho `Origin` para o Amazon S3. Para ter mais informações, consulte [the section called “Conteúdo em cache com base nos cabeçalhos de solicitação”](#).

Se, no momento, você distribui conteúdo diretamente do bucket do Amazon S3 usando seu próprio nome de domínio (como `example.com`) em vez do nome de domínio do bucket do Amazon S3 (como `amzn-s3-demo-bucket.s3.us-west-2.amazonaws.com`), é possível adicionar o CloudFront sem interrupção usando o procedimento a seguir.

Como adicionar o CloudFront quando você já estiver distribuindo seu conteúdo pelo Amazon S3

1. Crie uma distribuição na do CloudFront. Para ter mais informações, consulte [the section called “Criar uma distribuição”](#).

Ao criar a distribuição, especifique o nome do seu bucket do Amazon S3 como o servidor de origem.

Important

Para que seu bucket funcione com o CloudFront, o nome deve estar de acordo com os requisitos de nomenclatura do DNS. Para obter mais informações, acesse [Regras de atribuição de nomes de buckets](#) no Manual do usuário do Amazon Simple Storage Service.

Se estiver usando um CNAME com o Amazon S3, especifique-o para a distribuição.

2. Crie uma página da web de teste que contenha links para objetos publicamente legíveis no bucket do Amazon S3 e teste os links. Para esse teste inicial, use o nome de domínio do CloudFront da distribuição nos URLs dos objetos, por exemplo, `https://d111111abcdef8.cloudfront.net/images/image.jpg`.

Para mais informações sobre o formato dos URLs do CloudFront, consulte [the section called “Personalizar URLs de arquivos”](#).

3. Se estiver usando CNAMEs do Amazon S3, a aplicação usará seu nome de domínio (por exemplo, exemplo.com) para fazer referência aos objetos do bucket do Amazon S3, em vez de usar o nome do bucket (por exemplo, amzn-s3-demo-bucket.s3.amazonaws.com). Para continuar usando seu nome de domínio para fazer referência aos objetos, em vez do nome de domínio do CloudFront para sua distribuição (por exemplo, d111111abcdef8.cloudfront.net), é necessário atualizar as configurações com seu provedor de serviço de DNS.

Para que os CNAMEs do Amazon S3 funcionem, o provedor de serviço de DNS deve ter um conjunto de registros de recurso CNAME para seu domínio que encaminhe consultas para o domínio do seu bucket do Amazon S3. Por exemplo, se um usuário solicitar este objeto:

```
https://example.com/images/image.jpg
```

A solicitação será roteada novamente de forma automática, e o usuário verá este objeto:

```
https://amzn-s3-demo-bucket.s3.amazonaws.com/images/image.jpg
```

Para rotear consultas para a distribuição do CloudFront, em vez do bucket do Amazon S3, use o método fornecido por seu provedor de serviço de DNS para atualizar o conjunto de registros de recurso CNAME do seu domínio. Esse registro CNAME atualizado redirecionará as consultas de DNS do domínio para o nome de domínio do CloudFront da distribuição. Para obter mais informações, consulte a documentação fornecida por seu provedor de serviço de DNS.

Note

Se estiver usando o Route 53 como o serviço de DNS, você poderá usar um conjunto de registros de recurso CNAME ou um conjunto de registros de recurso de alias. Para obter informações sobre como editar conjuntos de registros do recurso, consulte [Editar registros](#). Para obter informações sobre conjuntos de registros do recurso de alias, consulte [Escolher entre registros de alias e não alias](#). Ambos os tópicos estão no Guia do desenvolvedor do Amazon Route 53.

Para mais informações sobre como usar CNAMEs com o CloudFront, consulte [the section called “Usar URLs personalizados”](#).

Depois de atualizar o conjunto de registros de recurso CNAME, pode levar até 72 horas para que as alterações sejam propagadas em todo o sistema DNS, embora normalmente ocorra mais rápido. Durante esse período, algumas solicitações do seu conteúdo continuarão sendo roteadas para o bucket do Amazon S3 e outras serão direcionadas para o CloudFront.

Mover um bucket do Amazon S3 para uma Região da AWS diferente

Se estiver usando o Amazon S3 como a origem de uma distribuição do CloudFront e mover o bucket para uma Região da AWS diferente, o CloudFront poderá levar até uma hora para atualizar os registros para usar uma nova região quando ocorrer ambas as situações:

- Você estiver usando uma identidade de acesso de origem (OAI) do CloudFront para restringir o acesso ao bucket
- Você mover o bucket para uma região do Amazon S3 que requer o Signature versão 4 para autenticação.

Quando estiver usando OAIs, o CloudFront você usará a região (entre outros valores) para calcular a assinatura usada para solicitar objetos do seu bucket. Para obter mais informações sobre OAIs, consulte [the section called “Usar uma identidade de acesso à origem \(herdada, não recomendado\)”](#). Para obter uma lista de Regiões da AWS compatíveis com o Signature versão 2, consulte [Processo de assinatura do Signature versão 2](#) na Referência geral da Amazon Web Services.

Para forçar uma atualização mais rápida dos registros do CloudFront, é possível atualizar a distribuição do CloudFront, por exemplo, atualizando o campo Description (Descrição) na guia General (Geral) do console do CloudFront. Quando você atualiza uma distribuição, o CloudFront verifica imediatamente a região em que o bucket está. A propagação da alteração para todos os locais de borda deve levar apenas alguns minutos.

Usar um contêiner do MediaStore ou um canal do MediaPackage

Para fazer uma transmissão de vídeo usando o CloudFront, é possível configurar um bucket do Amazon S3 definido como um contêiner do MediaStore, ou criar um canal e endpoints com o MediaPackage. Depois, você poderá criar e configurar uma distribuição no CloudFront para transmitir o vídeo.

Para obter mais informações e instruções de todas as etapas, consulte os tópicos a seguir:

- [the section called “Distribuir vídeo usando o AWS Elemental MediaStore como origem”](#)
- [the section called “Veicular vídeo ao vivo formatado com o AWS Elemental MediaPackage”](#)

Usar um Application Load Balancer

Se a origem for um ou mais servidores HTTP(S) (servidores web) hospedados em uma ou mais instâncias do Amazon EC2, você poderá usar um Application Load Balancer voltado para a internet e distribuir o tráfego para as instâncias. Um balanceador de carga voltado para a internet tem um nome de DNS que pode ser resolvido publicamente e roteia solicitações de clientes até destinos na internet.

Para obter mais informações sobre como usar um Application Load Balancer como a origem para o CloudFront, incluindo como garantir que os visualizadores só possam acessar os servidores Web por meio do CloudFront e não acessando o balanceador de carga diretamente, consulte [the section called “Restringir o acesso aos Application Load Balancers”](#).

Usar um URL da função do Lambda

Um [URL da função do Lambda](#) é um endpoint HTTPS dedicado para uma função do Lambda. É possível usar um URL da função do Lambda para criar uma aplicação web sem servidor inteiramente dentro do Lambda. Você pode invocar a aplicação Web do Lambda diretamente por meio do URL da função, sem a necessidade de se integrar ao API Gateway ou a um Application Load Balancer.

Se criar uma aplicação web sem servidor usando funções do Lambda com URLs da função, você poderá adicionar o CloudFront para obter os seguintes benefícios:

- Acelerar a aplicação armazenando conteúdo em cache mais próximo dos visualizadores
- Usar um nome de domínio personalizado para a aplicação Web
- Encaminhar diferentes caminhos de URL para diferentes funções do Lambda usando comportamentos de cache do CloudFront
- Bloquear solicitações específicas usando restrições geográficas do CloudFront ou o AWS WAF (ou ambos)
- Usar o AWS WAF com o CloudFront para ajudar a proteger a aplicação contra bots mal-intencionados, ajudar a evitar violações comuns de aplicações e aprimorar a proteção contra ataques DDoS

Para usar um URL da função do Lambda como a origem de uma distribuição do CloudFront, especifique o nome de domínio completo do URL da função do Lambda como o domínio de origem. Um nome de domínio do URL da função do Lambda usa o seguinte formato:

function-URL-ID.lambda-url.AWS-Region.on.aws

Ao usar um URL da função do Lambda como origem para uma distribuição do CloudFront, o URL da função deve estar acessível publicamente. Para fazer isso, use uma destas opções:

- Se você usa o controle de acesso à origem (OAC), o parâmetro `AuthType` do URL da função do Lambda deverá usar o valor `AWS_IAM` e autorizar a permissão `lambda:InvokeFunctionUrl` em uma política baseada em recursos. Consulte mais informações sobre como usar URLs de função do Lambda para OAC em [Restringir o acesso a uma origem do URL de função do AWS Lambda](#).
- Se não usar o OAC, você poderá definir o parâmetro `AuthType` do URL da função como `NONE` e autorizar a permissão `lambda:InvokeFunctionUrl` em uma política baseada em recurso.

Também é possível [adicionar um cabeçalho de origem personalizado](#) às solicitações que o CloudFront envia à origem e escrever o código da função para retornar uma resposta de erro caso o cabeçalho não esteja presente na solicitação. Isso ajuda a garantir que os usuários só possam acessar a aplicação Web por meio do CloudFront, e não usar diretamente o URL da função do Lambda.

Para obter mais informações sobre os URLs de função do Lambda, consulte os seguintes tópicos no Guia do desenvolvedor do AWS Lambda:

- [URLs da função do Lambda](#): uma visão geral do recurso de URLs da função do Lambda
- [Chamar URLs da função do Lambda](#): inclui detalhes sobre as cargas úteis de solicitação e resposta a serem usadas para codificar a aplicação Web sem servidor
- [Modelo de segurança e autenticação para URLs de função do Lambda](#): inclui detalhes sobre os tipos de autenticação do Lambda

Usar o Amazon EC2 (ou outra origem personalizada)

Uma origem personalizada é um servidor web HTTP(S) com um nome DNS que pode ser resolvido publicamente e encaminha solicitações de clientes a destinos pela internet. O servidor HTTP(S) pode ser hospedado na AWS (por exemplo, uma instância do Amazon EC2) ou em outro local. Uma

origem do Amazon S3 configurada como um endpoint do site também é considerada como uma origem personalizada. Para ter mais informações, consulte [the section called “Usar um bucket do Amazon S3 configurado como um endpoint do site”](#).

Ao usar seu próprio servidor HTTP como uma origem personalizada, especifique o nome DNS do servidor, além das portas HTTP e HTTPS e o protocolo que o CloudFront deve usar para obter objetos da origem.

A maioria dos recursos do CloudFront são compatíveis ao usar uma origem personalizada com a exceção de conteúdo privado. Embora seja possível usar um URL assinado para distribuir conteúdo de uma origem personalizada, para que o CloudFront acesse a origem personalizada, ela deve permanecer publicamente acessível. Para ter mais informações, consulte [the section called “Restringir conteúdo com URLs e cookies assinados”](#).

Siga essas diretrizes de como usar instâncias do Amazon EC2 e outras origens personalizadas com o CloudFront.

- Hospede e forneça o mesmo conteúdo em todos os servidores que estão fornecendo conteúdo da mesma origem do CloudFront. Para obter mais informações, consulte [the section called “Configurações de origem”](#) [the section called “Configurações de distribuição”](#) no tópico.
- Registre as entradas do cabeçalho X-Amz-Cf-Id em todos os servidores, caso você precise que o AWS Support ou o CloudFront use esse valor para depuração.
- Restrinja as solicitações às portas HTTP e HTTPS em que a origem personalizada escuta.
- Sincronizar os relógios de todos os servidores na sua implementação. Observe que o CloudFront usa o Tempo Universal Coordenado (UTC) para signed URLs assinados e cookies assinados, para logs e relatórios. Além disso, se você monitorar a atividade do CloudFront usando métricas do CloudWatch. Observe que o CloudWatch também usa UTC.
- Use servidores redundantes para lidar com falhas.
- Para obter informações sobre como usar uma origem personalizada para fornecer conteúdo privado, consulte [the section called “Restringir o acesso a arquivos em origens personalizadas”](#).
- Para obter informações sobre o comportamento da solicitação e da resposta e os códigos de status HTTP compatíveis, consulte [Comportamento de solicitações e respostas](#).

Se você usa o Amazon EC2 para uma origem personalizada, recomendamos que faça o seguinte:

- Use uma imagem de máquina da Amazon que instala automaticamente o software para um servidor da web. Para mais informações, consulte a [documentação do Amazon EC2](#).

- Use um balanceador de carga do Elastic Load Balancing para lidar com o tráfego em várias instâncias do Amazon EC2 e isolar sua aplicação das alterações nas instâncias do Amazon EC2. Por exemplo, se usar um balanceador de carga, você poderá adicionar e excluir instâncias do Amazon EC2 sem alterar a aplicação. Para mais informações, consulte a [documentação do Elastic Load Balancing](#).
- Ao criar sua distribuição do CloudFront, especifique o URL do balanceador de carga para o nome de domínio do seu servidor de origem. Para ter mais informações, consulte [the section called “Criar uma distribuição”](#).

Usar grupos de origem do CloudFront

É possível especificar um grupo de origens para sua origem do CloudFront se, por exemplo, você quiser configurar o failover de origem para cenários quando precisar de alta disponibilidade. Use o failover de origem para designar uma origem principal para o CloudFront mais uma segunda origem para a qual o CloudFront muda automaticamente quando a origem primária retorna respostas específicas de falha com código de status HTTP.

Para obter mais informações, incluindo as etapas para configurar um grupo de origens, consulte [the section called “Aumentar disponibilidade com o failover de origem”](#).

Usar URLs personalizados adicionando nomes de domínio alternativos (CNAMEs)

Quando você cria uma distribuição, o CloudFront fornece um nome de domínio a ela, por exemplo, d111111abcdef8.cloudfront.net. Em vez de usar esse nome de domínio fornecido, é possível usar um nome de domínio alternativo (também conhecido como CNAME).

Saiba como usar seu próprio nome de domínio, como www.exemplo.com. Consulte os seguintes tópicos:

Tópicos

- [Requisitos para o uso de nomes de domínio alternativos](#)
- [Restrições de uso de nomes de domínio alternativos](#)
- [Adicionar um nome de domínio alternativo](#)
- [Mover um nome de domínio alternativo para uma distribuição diferente](#)

- [Remover um nome de domínio alternativo](#)
- [Usar curingas em nomes de domínio alternativos](#)

Requisitos para o uso de nomes de domínio alternativos

Ao adicionar um nome de domínio alternativo, como `www.example.com`, a uma distribuição do CloudFront, os requisitos são os seguintes:

Os nomes de domínio alternativos devem estar em letras minúsculas

Todos os nomes de domínio alternativos (CNAMEs) devem estar em letras minúsculas.

Os nomes de domínio alternativos devem ser cobertos por um certificado SSL/TLS válido

Para adicionar um nome de domínio alternativo (CNAME) a uma distribuição do CloudFront, é necessário anexar à distribuição um certificado SSL/TLS válido e confiável, que abranja o nome de domínio alternativo. Isso garante que somente as pessoas com acesso ao certificado do domínio possam associar ao CloudFront um CNAME relacionado ao seu domínio.

Um certificado confiável é aquele que é emitido pelo AWS Certificate Manager (ACM) ou por outra autoridade de certificação (CA) válida. Você pode usar um certificado autoassinado para validar um CNAME existente, mas não para um CNAME novo. O CloudFront oferece suporte às mesmas autoridades de certificação que o Mozilla. Para ver a lista atual, consulte [Mozilla Included CA Certificate List](#).

Para verificar um nome de domínio alternativo usando o certificado que você anexar, incluindo nomes de domínio alternativos que incluem curingas, o CloudFront verifica o nome de assunto alternativo (SAN, subject alternative name) no certificado. O nome de domínio alternativo que você estiver adicionando deverá ser coberto pelo SAN.

Note

É possível anexar somente um certificado por vez a uma distribuição do CloudFront.

Você prova que está autorizado a adicionar um determinado nome de domínio alternativo à distribuição de uma das seguintes maneiras:

- Anexar um certificado que inclui o nome de domínio alternativo, como `nome_do_produto.exemplo.com`.

- Anexando um certificado que inclui um caractere curinga * no início de um nome de domínio, para cobrir vários subdomínios com um certificado. Ao especificar um caractere curinga, é possível adicionar vários subdomínios como nomes de domínio alternativos no CloudFront.

Os exemplos a seguir ilustram como usar caracteres curinga em nomes de domínio em um trabalho certificado para autorizar a adição de nomes de domínio alternativos específicos no CloudFront.

- Você deseja adicionar marketing.exemplo.com como um nome de domínio alternativo. Você lista no seu certificado o seguinte nome de domínio: *.exemplo.com. Ao anexar esse certificado ao CloudFront, você poderá adicionar qualquer nome de domínio alternativo à distribuição que substitua o caractere curinga nesse nível, incluindo marketing.exemplo.com. Você também pode, por exemplo, adicionar os seguintes nomes de domínio alternativos:
 - produto.exemplo.com
 - api.example.com

No entanto, você não pode adicionar nomes de domínio alternativos que estão em níveis maiores ou menores que o caractere curinga. Por exemplo, você não pode adicionar os nomes de domínio alternativos exemplo.com nem marketing.produto.exemplo.com.

- Você deseja adicionar exemplo.com como um nome de domínio alternativo. Para fazer isso, você deve listar o próprio nome de domínio exemplo.com no certificado que anexa à distribuição.
- Você deseja adicionar marketing.produto.exemplo.com como um nome de domínio alternativo. Para fazer isso, você pode listar *.produto.exemplo.com no certificado ou pode listar o próprio marketing.produto.exemplo.com no certificado.

Permissão para alterar a configuração de DNS

Ao adicionar nomes de domínio alternativos, é necessário criar registros CNAME para rotear consultas de DNS para os nomes de domínio alternativos à distribuição do CloudFront. Para fazer isso, você deve ter permissão para criar registros CNAME com o provedor de serviço DNS para os nomes de domínio alternativos que está usando. Normalmente, isso significa que você tem os domínios, mas pode estar desenvolvendo um aplicativo para o proprietário dele.

Nomes de domínio alternativos e HTTP

Se quiser que os visualizadores usem HTTPS com um nome de domínio alternativo, será necessário realizar algumas configurações adicionais. Para ter mais informações, consulte [Usar nomes de domínio alternativos e HTTPS](#).

Restrições de uso de nomes de domínio alternativos

Observe as seguintes restrições de uso de nomes de domínio alternativos:

Número máximo de nomes de domínio alternativos

Para saber o número máximo atual de nomes de domínio alternativos que podem ser adicionados a uma distribuição ou para solicitar uma cota maior (anteriormente conhecida como limite), consulte [Cotas gerais para distribuições](#).

Duplicar e substituir nomes de domínio alternativos

Não será possível adicionar um nome de domínio alternativo a uma distribuição do CloudFront se o mesmo nome de domínio alternativo já existir em outra distribuição do CloudFront, mesmo se a sua conta da AWS for a proprietária da outra distribuição.

No entanto, você pode adicionar um nome de domínio alternativo curinga, como *.exemplo.com, que inclui (sobrepõe) um nome de domínio alternativo não curinga, como www.exemplo.com. Se você tiver nomes de domínio alternativos sobrepostos em duas distribuições, o CloudFront enviará a solicitação para a distribuição com a correspondência de nome mais específica, independentemente da distribuição para a qual o registro de DNS apontar. Por exemplo, marketing.domínio.com é mais específico que *.domínio.com.

Se você tiver uma entrada de DNS curinga existente que aponta para uma distribuição do CloudFront e receber um erro de DNS configurado incorretamente ao tentar adicionar um novo CNAME com um nome mais específico, consulte [O CloudFront retorna um erro de registro DNS configurado incorretamente quando tento adicionar um novo CNAME](#).

Domain fronting

O CloudFront inclui proteção contra a ocorrência de domain fronting entre contas diferentes da AWS. O domain fronting é um cenário em que um cliente não padrão cria uma conexão TLS/SSL com um nome de domínio em uma conta da AWS, mas faz uma solicitação HTTPS para um nome não relacionado em outra conta da AWS. Por exemplo, a conexão TLS pode se conectar a www.exemplo.com e enviar uma solicitação HTTP para www.exemplo.org.

Para evitar casos em que o domain fronting passe por diferentes contas da AWS, o CloudFront garante que a conta da AWS que possui o certificado que serve para determinada conexão sempre corresponda à conta da AWS que possui a solicitação que ele processa na mesma conexão.

Se os dois números de conta da AWS não corresponderem, o CloudFront responderá com uma resposta HTTP 421 Misdirected Request (solicitação indevida) para oferecer ao cliente uma oportunidade de se conectar usando o domínio correto.

Adicionar um nome de domínio alternativo no nó superior (apex de zona) de um domínio

Ao adicionar um nome de domínio alternativo a uma distribuição, geralmente, você cria um registro CNAME na sua configuração de DNS para rotear consultas de DNS do nome de domínio para sua distribuição do CloudFront. No entanto, não é possível criar um registro CNAME no nó superior de um namespace DNS, também conhecido como o apex de zona. O protocolo DNS não permite isso. Por exemplo, se você registrar o nome do DNS exemplo.com, o apex de zona será exemplo.com. Você não pode criar um registro CNAME para exemplo.com, mas pode criar registros CNAME para www.exemplo.com, produtonovo.exemplo.com e assim por diante.

Se estiver usando o Route 53 como o serviço de DNS, você poderá criar um conjunto de registros de recurso de alias, que tem duas vantagens sobre os registros CNAME. Você pode criar um conjunto de registros de recurso de alias para um nome de domínio no nó superior (example.com). Além disso, ao usar um conjunto de registros de recurso de alias, você não paga pelas consultas do Route 53.

Note

Se ativar o IPv6, você deve criar dois conjuntos de registros de recurso de alias: um para rotear o tráfego IPv4 (um registro A) e outro para rotear o tráfego IPv6 (um registro AAAA). Para obter mais informações, consulte [Enable IPv6](#) no tópico [Referência de configurações da distribuição](#).

Para mais informações, consulte [Rotear o tráfego para uma distribuição na Web do Amazon CloudFront usando seu nome de domínio](#) no Guia do desenvolvedor do Amazon Route 53.

Adicionar um nome de domínio alternativo

A lista de tarefas a seguir descreve como usar o console do CloudFront para adicionar um nome de domínio alternativo à distribuição, para que você possa usar seu próprio nome de domínio nos links, em vez do nome de domínio do CloudFront. Para obter informações sobre como atualizar sua distribuição usando a API do CloudFront, consulte [Configurar distribuições](#).

Note

Se você quiser que os visualizadores usem HTTPS com seu nome de domínio alternativo, consulte [Usar nomes de domínio alternativos e HTTPS](#).

Antes de começar: faça o seguinte antes de atualizar a distribuição para adicionar um nome de domínio alternativo:

- Registre o nome de domínio com o Route 53 ou outro provedor de domínio.
- Obtenha um certificado SSL/TLS de uma autoridade de certificação (CA) autorizada que cubra o nome de domínio. Adicione o certificado à sua distribuição para validar que você está autorizado a usar o domínio. Para ter mais informações, consulte [Requisitos para o uso de nomes de domínio alternativos](#).

Adicionar um nome de domínio alternativo

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha o ID da distribuição que você deseja atualizar.
3. Na guia General, escolha Edit.
4. Atualize os seguintes valores:

Alternate Domain Names (CNAMEs)

Como adicionar nomes de domínio alternativos. Separe os nomes de domínio com vírgulas ou digite cada nome de domínio em uma nova linha.

Certificado SSL

Escolha a seguinte configuração:

- Usar HTTPS: escolha Custom SSL Certificate (Certificado SSL personalizado) e escolha um certificado na lista. A lista inclui certificados provisionados pelo AWS Certificate Manager (ACM), certificados adquiridos de outra CA e carregados no ACM e certificados adquiridos de outra CA e carregados no armazenamento de certificados do IAM.

Se você fez upload de um certificado no armazenamento de certificados do IAM, mas ele não for exibido na lista, reveja o procedimento [Importar um certificado SSL/TLS](#) para confirmar se fez upload do certificado corretamente.

Se você escolher essa configuração, recomendamos que use apenas um nome de domínio alternativo nos URLs dos seus objetos (<https://www.example.com/logo.jpg>). Se você usar o nome de domínio da distribuição do CloudFront (<https://d1111111abcdef8.cloudfront.net.cloudfront.net/logo.jpg>), um visualizador poderá se comportar da seguinte forma, dependendo do valor escolhido para Clients Supported (Clientes compatíveis):

- All Clients (Todos os clientes): se o visualizador não for compatível com SNI, ele receberá um aviso, porque o nome de domínio do CloudFront não corresponde ao nome de domínio do certificado TLS/SSL.
- Only Clients that Support Server Name Indication (SNI) (Somente os clientes que oferecem suporte à indicação de nome de servidor (SNI)): o CloudFront interrompe a conexão com o visualizador sem retornar o objeto.

Clients Supported

Escolha uma opção:

- All Clients (Todos os clientes): o CloudFront fornece seu conteúdo HTTP usando endereços IP dedicados. Se você selecionar essa opção, será cobrado encargos adicionais ao associar seu certificado SSL/TLS a uma distribuição ativada. Para mais informações, consulte [Definição de preços do Amazon CloudFront](#).
- Only Clients that Support Server Name Indication (SNI) (Recommended) *(Somente clientes compatíveis com o Server Name Indication [SNI] [Recomendado]): navegadores antigos ou outros clientes não compatíveis com SNI devem usar outro método para acessar o conteúdo.

Para ter mais informações, consulte [Escolher como o CloudFront atende a solicitações HTTPS](#).

5. Escolha Yes, Edit.
6. Na guia General na distribuição, confirme se Distribution Status foi alterado para Deployed. Se você tentar usar um nome de domínio alternativo antes de as atualizações da sua distribuição serem implementados, os links criados nas etapas a seguir poderão não funcionar.

- Configure o serviço de DNS para o nome de domínio alternativo (como `www.exemplo.com`) para rotear tráfego para o nome de domínio do CloudFront para sua distribuição (por exemplo, `d111111abcdef8.cloudfront.net`). O método usado depende se você está usando o Route 53 como o provedor de serviço de DNS para o domínio ou outro provedor.

 Note


Se o registro DNS já aponta para uma distribuição diferente daquela que você está atualizando, adicione o nome de domínio alternativo à distribuição somente após atualizar o DNS. Para ter mais informações, consulte [Restrições de uso de nomes de domínio alternativos](#).

Route 53

Crie um conjunto de registros de recursos de alias. Com um conjunto de registros de recurso de alias, você não paga pelas consultas do Route 53. Além disso, é possível criar um conjunto de registros de recurso de alias para o nome de domínio raiz (`example.com`), que não é permitido pelo DNS para CNAMEs. Para mais informações, consulte [Rotear o tráfego para uma distribuição na web do Amazon CloudFront usando seu nome de domínio](#) no Guia do desenvolvedor do Amazon Route 53.

Outro provedor de serviço de DNS

Use o método fornecido pelo provedor de serviço de DNS para adicionar um registro CNAME ao domínio. Esse novo registro CNAME redirecionará consultas de DNS do seu nome de domínio alternativo (como `www.exemplo.com`) para o nome de domínio do CloudFront da sua distribuição (por exemplo, `d111111abcdef8.cloudfront.net`). Para obter mais informações, consulte a documentação fornecida por seu provedor de serviço de DNS.

 Important

Se você já tiver um registro CNAME para seu nome de domínio alternativo, atualize-o ou substitua-o por um novo que aponte para o nome de domínio do CloudFront da sua distribuição.

- Usando `dig` ou uma ferramenta de DNS semelhante, confirme se a configuração de DNS criada na etapa anterior aponta para o nome de domínio da sua distribuição.

O exemplo a seguir mostra uma solicitação dig no domínio `www.example.com`, bem como a parte relevante da resposta.

```
PROMPT> dig www.example.com

; <<> DiG 9.3.3rc2 <<> www.example.com
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15917
;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 2, ADDITIONAL: 0

;; QUESTION SECTION:
;www.example.com.      IN      A

;; ANSWER SECTION:
www.example.com. 10800 IN CNAME d111111abcdef8.cloudfront.net.
...
```

A seção de resposta mostra um registro CNAME que roteia consultas de `www.exemplo.com` para o nome de domínio de distribuição do CloudFront `d111111abcdef8.cloudfront.net`. Se o nome à direita de CNAME for o nome de domínio de sua distribuição do CloudFront, o registro CNAME estará configurado corretamente. Se o valor for qualquer outro, por exemplo, o nome de domínio do bucket do Amazon S3, o registro CNAME está configurado incorretamente. Nesse caso, volte para a etapa 7 e corrija o registro CNAME de modo que ele aponte para o nome de domínio da sua distribuição.

9. Teste o nome de domínio alternativo acessando URLs com seu nome de domínio em vez do nome de domínio do CloudFront da distribuição.
10. Na aplicação, altere os URLs para que os objetos usem o nome de domínio alternativo, em vez do nome de domínio da distribuição do CloudFront.

Mover um nome de domínio alternativo para uma distribuição diferente

Quando você tentar adicionar um nome de domínio alternativo a uma distribuição, mas o nome de domínio alternativo já estiver em uso em uma distribuição diferente, você receberá um erro `CNAMEAlreadyExists` (One or more of the CNAMEs you provided are already associated with a different resource (Um ou mais CNAMEs fornecidos já estão associados a um recurso diferente)).

Por exemplo, você recebe esse erro ao tentar adicionar `www.exemplo.com` a uma distribuição, mas `www.exemplo.com` já está associado a uma distribuição diferente.

Nesse caso, talvez você queira mover o nome de domínio alternativo existente de uma distribuição (a distribuição de origem) para outra (a distribuição de destino). As etapas a seguir são uma visão geral do processo. Para obter mais informações, siga o link em cada etapa da visão geral.

Para mover um nome de domínio alternativo

1. Configure a distribuição de destino. Essa distribuição deve ter um certificado SSL/TLS que cubra o nome de domínio alternativo que você está movendo. Para obter mais informações, consulte [Configurar a distribuição de destino](#).
2. Localize a distribuição de origem. Você pode usar o AWS Command Line Interface (AWS CLI) para localizar a distribuição à qual o nome de domínio alternativo está associado. Para obter mais informações, consulte [Localizar a distribuição de origem](#).
3. Mova o nome de domínio alternativo. O modo de fazer isso depende se as distribuições de origem e de destino estão na mesma conta da AWS. Para obter mais informações, consulte [the section called “Mover o nome de domínio alternativo”](#).

Configurar a distribuição de destino

Antes de mover um nome de domínio alternativo, você deve configurar a distribuição de destino (a distribuição para a qual você está movendo o nome de domínio alternativo).

Para configurar a distribuição de destino

1. Obtenha um certificado SSL/TLS que inclua o nome de domínio alternativo que você está movendo. Caso não possua um, poderá solicitá-lo no [AWS Certificate Manager \(ACM\)](#) ou obtê-lo junto a outra autoridade de certificação (CA) e importá-lo para o ACM. Certifique-se de solicitar ou importar o certificado na região Leste dos EUA (Norte da Virgínia) (`us-east-1`).
2. Se você ainda não criou a distribuição de destino, crie-a agora. Como parte da criação da distribuição de destino, associe seu certificado (da etapa anterior) à distribuição. Para obter mais informações, consulte [Criar uma distribuição](#).

Se você já tiver uma distribuição de destino, associe seu certificado (da etapa anterior) à distribuição de destino. Para obter mais informações, consulte [Atualizar uma distribuição](#).

3. Crie um registro TXT de DNS que associe o nome de domínio alternativo ao nome de domínio de distribuição da distribuição de destino. Crie seu registro TXT com um sublinhado (_) na frente do nome de domínio alternativo. O exemplo a seguir mostra um registro TXT de DNS:

```
_www.example.com TXT d111111abcdef8.cloudfront.net
```

O CloudFront usa esse registro TXT para validar que você é o proprietário do nome de domínio alternativo.

Localizar a distribuição de origem

Antes de mover um nome de domínio alternativo de uma distribuição para outra, será necessário localizar a distribuição de origem (a distribuição em que o nome de domínio alternativo está atualmente em uso). Quando você conhece o ID da conta da AWS das distribuições de origem e de destino, pode determinar como o nome de domínio alternativo será movido.

Para localizar a distribuição de origem para o nome de domínio alternativo

1. Use o [comando list-conflicting-aliases do CloudFront AWS Command Line Interface \(AWS CLI\)](#) conforme mostrado no exemplo a seguir. Substitua *www.exemplo.com* pelo nome de domínio alternativo e *EDFDVBD6EXAMPLE* pelo ID da distribuição de destino [configurada anteriormente](#). Execute este comando usando credenciais que estão na mesma conta da AWS que a distribuição de destino. Para usar esse comando, você deve ter permissões `cloudfront:GetDistribution` e `cloudfront:ListConflictingAlias` na distribuição de destino.

```
aws cloudfront list-conflicting-aliases --alias www.example.com --distribution-id EDFDVBD6EXAMPLE
```

A saída do comando mostra uma lista de todos os nomes de domínio alternativos que entram em conflito ou se sobrepõem ao fornecido. Por exemplo:

- Se você fornecer *www.exemplo.com* ao comando, a saída do comando incluirá *www.exemplo.com* e o nome de domínio alternativo curinga sobreposto (**.exemplo.com*), se ele existir.

- Se você fornecer *.exemplo.com ao comando, a saída do comando incluirá *.exemplo.com e quaisquer nomes de domínio alternativos cobertos por esse curinga (por exemplo, www.exemplo.com, teste.exemplo.com, dev.exemplo.com e assim por diante).

Para cada nome de domínio alternativo na saída do comando, você poderá ver o ID da distribuição à qual ele está associado e o ID da conta da AWS proprietária da distribuição. Os IDs de distribuição e de conta são parcialmente ocultados, o que permite identificar as distribuições e contas que você possui, mas ajuda a proteger as informações daquelas que você não possui.

2. Na saída do comando, localize a distribuição para o nome de domínio alternativo que você está movendo e observe o ID da conta da AWS da distribuição de origem. Compare o ID da conta da distribuição de origem com o ID da conta em que você criou a distribuição de destino e determine se essas duas distribuições estão na mesma conta da AWS. Isso ajuda você a determinar como mover o nome de domínio alternativo.

Para mover o nome de domínio alternativo, consulte o tópico a seguir.

Mover o nome de domínio alternativo

Dependendo da sua situação, escolha uma das seguintes formas de mover o nome de domínio alternativo:

Se as distribuições de origem e de destino estiverem na mesma conta da AWS

Use o comando `associate-alias` na AWS CLI para mover o nome de domínio alternativo. Esse método funciona para todas as movimentações da mesma conta, inclusive quando o nome de domínio alternativo é um domínio apex (também chamado de domínio raiz, como exemplo.com). Para obter mais informações, consulte [the section called “Usar associate-alias para mover um nome de domínio alternativo”](#).

Se as distribuições de origem e de destino estiverem em contas da AWS diferentes

Se você tiver acesso à distribuição de origem, o nome de domínio alternativo não será um domínio apex (também chamado de domínio raiz, como exemplo.com) e, se ainda não estiver usando um curinga que se sobreponha a esse nome de domínio alternativo, use um curinga para mover o nome de domínio alternativo. Para ter mais informações, consulte [the section called “Usar um curinga para mover um nome de domínio alternativo”](#).

Se não tiver acesso à conta da AWS da distribuição de origem, você poderá tentar usar o comando `associate-alias` na AWS CLI para mover o nome de domínio alternativo. Se a distribuição de origem estiver desabilitada, você poderá mover o nome de domínio alternativo. Para ter mais informações, consulte [the section called “Usar `associate-alias` para mover um nome de domínio alternativo”](#). Se o comando `associate-alias` não funciona, entre em contato: AWS Support. Para ter mais informações, consulte [the section called “Entre em contato com o AWS Support para mover um nome de domínio alternativo”](#).

Usar `associate-alias` para mover um nome de domínio alternativo

Se a distribuição de origem estiver na mesma conta da AWS que a distribuição de destino, ou se ela estiver em uma conta diferente, mas desabilitada, você pode usar o [comando `associate-alias` do CloudFront na AWS CLI](#) para mover o nome de domínio alternativo.

Para usar `associate-alias` para mover um nome de domínio alternativo

1. Use a AWS CLI para executar o comando `associate-alias` do CloudFront, conforme mostrado no exemplo a seguir. Substitua `www.exemplo.com` pelo nome de domínio alternativo e `EDFDVBD6EXAMPLE` pelo ID da distribuição de destino. Execute este comando usando credenciais que estão na mesma conta da AWS que a distribuição de destino. Observe as seguintes restrições ao uso desse comando:
 - Você deve ter permissões `cloudfront:AssociateAlias` e `cloudfront:UpdateDistribution` na distribuição de destino.
 - Se as distribuições de origem e de destino estiverem na mesma conta da AWS, você deverá ter permissão `cloudfront:UpdateDistribution` na distribuição de origem.
 - Se as distribuições de origem e de destino estiverem em contas da AWS diferentes, a distribuição de origem deverá ser desabilitada.
 - A distribuição de destino deve ser configurada conforme descrito em [the section called “Configurar a distribuição de destino”](#).

```
aws cloudfront associate-alias --alias www.example.com --target-distribution-id EDFDVBD6EXAMPLE
```

Este comando atualiza ambas as distribuições removendo o nome de domínio alternativo da distribuição de origem e adicionando-o à distribuição de destino.

2. Depois que a distribuição de destino estiver totalmente implantada, atualize sua configuração de DNS para apontar o registro de DNS do nome de domínio alternativo para o nome de domínio da distribuição de destino.

Usar um curinga para mover um nome de domínio alternativo

Se a distribuição de origem estiver em uma conta da AWS diferente da distribuição de destino e a distribuição de origem estiver habilitada, você poderá usar um curinga para mover o nome de domínio alternativo.

Note

Você não pode usar um caractere curinga para mover um domínio apex (como exemplo.com). Para mover um domínio apex quando as distribuições de origem e de destino estiverem em contas da AWS diferentes, entre em contato com o AWS Support. Para obter mais informações, consulte [the section called “Entre em contato com o AWS Support para mover um nome de domínio alternativo”](#).

Para usar um curinga para mover um nome de domínio alternativo

Note

Esse processo envolve várias atualizações para suas distribuições. Aguarde até que cada distribuição implante totalmente a última alteração antes de prosseguir para a próxima etapa.

1. Atualize a distribuição de destino para adicionar um nome de domínio alternativo curinga que cubra o nome de domínio alternativo que você está movendo. Por exemplo, se o nome de domínio alternativo que você está movendo for `www.exemplo.com`, adicione o nome de domínio alternativo `*.exemplo.com` à distribuição de destino. Para fazer isso, o certificado SSL/TLS na distribuição de destino deve incluir o nome de domínio curinga. Para obter mais informações, consulte [the section called “Atualizar uma distribuição”](#).
2. Atualize as configurações de DNS para o nome de domínio alternativo para apontar para o nome de domínio da distribuição de destino. Por exemplo, se o nome de domínio alternativo que você está movendo for `www.exemplo.com`, atualize o registro de DNS para `www.exemplo.com` para rotear o tráfego para o nome de domínio da distribuição de destino (por exemplo, `d111111abcdef8.cloudfront.net`).

Note

Mesmo depois que as configurações de DNS forem atualizadas, o nome de domínio alternativo ainda será atendido pela distribuição de origem, pois é nela que o nome de domínio alternativo está configurado no momento.

3. Atualize a distribuição de origem para remover o nome de domínio alternativo. Para obter mais informações, consulte [Atualizar uma distribuição](#).
4. Atualize a distribuição de destino para adicionar o nome de domínio alternativo. Para obter mais informações, consulte [Atualizar uma distribuição](#).
5. Use o dig (ou uma ferramenta de consulta de DNS semelhante) para validar que o registro de DNS para o nome de domínio alternativo está sendo resolvido para o nome de domínio da distribuição de destino.
6. (Opcional) Atualize a distribuição de destino para remover o nome de domínio alternativo curinga.

Entre em contato com o AWS Support para mover um nome de domínio alternativo

Se as distribuições de origem e de destino estiverem em contas da AWS diferentes e você não tiver acesso à conta da AWS da distribuição de origem ou não puder desabilitar a distribuição de origem, você poderá entrar em contato com o AWS Support para mover o nome de domínio alternativo.

Para entrar em contato com o AWS Support para mover um nome de domínio alternativo

1. Configure uma distribuição de destino, incluindo o registro TXT de DNS que aponta para a distribuição de destino. Para obter mais informações, consulte [Configurar a distribuição de destino](#).
2. [Entre em contato com o AWS Support](#) para solicitar uma verificação de que você é o proprietário do domínio e mover o domínio para a nova distribuição do CloudFront para você.
3. Depois que a distribuição de destino estiver totalmente implantada, atualize sua configuração de DNS para apontar o registro de DNS do nome de domínio alternativo para o nome de domínio da distribuição de destino.

Remover um nome de domínio alternativo

Se você quiser interromper o roteamento do tráfego de um domínio ou subdomínio para uma distribuição do CloudFront, siga as etapas nesta seção para atualizar a configuração de DNS e a configuração de distribuição do CloudFront.

É importante que você remova os nomes de domínio alternativos da distribuição e atualize sua configuração DNS. Isso ajudará a evitar problemas posteriormente se você quiser associar o nome de domínio a outra distribuição do CloudFront. Se um nome de domínio alternativo já estiver associado a uma distribuição, ele não poderá ser configurado com outra.

Note

Se você quiser remover o nome de domínio alternativo dessa distribuição para que possa adicioná-lo a outra, siga as etapas em [Mover um nome de domínio alternativo para uma distribuição diferente](#). Se você seguir as etapas descritas aqui (para remover um domínio) e depois adicionar o domínio a outra distribuição, haverá um período durante o qual o domínio não será vinculado à nova distribuição porque o CloudFront estará propagando as atualizações para os pontos de presença.

Para remover o nome de domínio alternativo de uma distribuição

1. Para começar, roteie o tráfego da Internet do seu domínio para outro recurso que não seja sua distribuição do CloudFront, como um balanceador de carga do Elastic Load Balancing. Se preferir, exclua o registro de DNS que direciona o tráfego para o CloudFront.

Siga um destes procedimentos, dependendo do serviço DNS do seu domínio:

- Se estiver usando o Route 53, atualize ou exclua registros de alias ou registros CNAME. Para mais informações, consulte [Editar registros](#) ou [Excluir registros](#).
 - Se estiver usando outro provedor de serviços de DNS, use o método fornecido pelo provedor de serviços de DNS para atualizar ou excluir o registro CNAME que direciona o tráfego para o CloudFront. Para obter mais informações, consulte a documentação fornecida por seu provedor de serviço de DNS.
2. Depois de atualizar os registros DNS do seu domínio, aguarde até que as alterações sejam propagadas e os resolvedores de DNS estejam roteando o tráfego para o novo recurso. Você

pode verificar quando o processo foi concluído criando alguns links de teste que usam seu domínio no URL.

3. Faça login no AWS Management Console, abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home> e atualize sua distribuição do CloudFront para remover o nome de domínio fazendo o seguinte:
 - a. Escolha o ID da distribuição que você deseja atualizar.
 - b. Na guia General, escolha Edit.
 - c. Em Alternate Domain Names (CNAMEs), remova o nome de domínio alternativo (ou nomes de domínio) que você não quer mais usar na sua distribuição.
 - d. Escolha Yes, Edit.

Usar curingas em nomes de domínio alternativos

Ao adicionar nomes de domínio alternativos, você pode usar o curinga "*" no início de um nome de domínio, em vez de adicionar cada subdomínio. Por exemplo, com um nome de domínio alternativo de *.exemplo.com, você pode usar qualquer nome de domínio que termine com exemplo.com em seus URLs, como www.exemplo.com, nome_do_produto.exemplo.com, marketing.nome_do_produto.exemplo.com e assim por diante. Caminho para o objeto permanece o mesmo, independentemente do nome de domínio, por exemplo:

- www.exemplo.com/images/image.jpg
- nome_do_produto.exemplo.com/images/image.jpg
- marketing.nome_do_produto.exemplo.com/images/image.jpg

Siga estes requisitos para nomes de domínio alternativos que incluem curingas:

- O nome de domínio alternativo deve começar com um asterisco e um ponto (*.).
- Não é possível usar um curinga para substituir parte de um nome de subdomínio, como em *domínio.exemplo.com.
- Não é possível substituir um subdomínio no meio de um nome de domínio, como em subdomínio.*.exemplo.com.
- Todos os nomes de domínio alternativos, incluindo nomes de domínio alternativo que usam caracteres curingas, devem estar cobertos pelo nome alternativo da entidade (SAN) no certificado.

Um nome de domínio alternativo curinga, como *.exemplo.com, pode incluir outro nome de domínio alternativo em uso, como exemplo.com.

Usar WebSockets com distribuições do CloudFront

O Amazon CloudFront oferece suporte ao uso de WebSocket, um protocolo baseado em TCP que é útil quando forem necessárias conexões bidirecionais de longa duração entre clientes e servidores. Uma conexão persistente normalmente é um requisito com aplicativos em tempo real. Os cenários nos quais convém usar Websockets incluem plataformas de bate-papo sociais, espaços de trabalho de colaboração online, jogos multijogador e serviços que fornecem dados em tempo real, como feeds de plataformas de transações financeiras. Os dados sobre uma conexão WebSocket pode fluir em ambas as direções para comunicação full-duplex.

A funcionalidade WebSocket é habilitada automaticamente para que funcione com qualquer distribuição. Para usar o WebSockets, configure uma das seguintes opções no comportamento do cache anexado à sua distribuição:

- Encaminhe todos os cabeçalhos para sua origem. (É possível usar a [política de solicitação de origem gerenciada do AllViewer](#).)
- Encaminhe especificamente os cabeçalhos de solicitação Sec-WebSocket-Key e Sec-WebSocket-Version em sua política de solicitação de origem.

Como o protocolo WebSocket funciona

O protocolo WebSocket é um protocolo independente baseado em TCP que permite que você evite parte da sobrecarga, e possivelmente maior latência, de HTTP.

Para estabelecer uma conexão WebSocket, o cliente envia uma solicitação HTTP normal que usa semântica de atualização do HTTP para alterar o protocolo. O servidor pode concluir o handshake. A conexão WebSocket permanece aberta, e o cliente ou servidor pode enviar dados quadros entre si sem a necessidade de estabelecer novas conexões a cada vez.

Por padrão, o protocolo WebSocket usa a porta 80 para conexões WebSocket regulares e a porta 443 para conexões WebSocket sobre TLS/SSL. As opções [Política de protocolo do visualizador](#) e [Protocolo \(somente origens personalizadas\)](#) que você escolher para o CloudFront se aplicam a conexões WebSocket, bem como ao tráfego HTTP.

Requisitos de WebSocket

Solicitações WebSocket devem estar em conformidade com a [RFC 6455](#) nos formatos padrão a seguir.

Exemplo de solicitação do cliente:

```
GET /chat HTTP/1.1
Host: server.example.com
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: dGhlIHNhbXBsZSBub25jZQ==
Origin: https://example.com
Sec-WebSocket-Protocol: chat, superchat
Sec-WebSocket-Version: 13
```

Exemplo de resposta de servidor:

```
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: s3pPLMBiTxaQ9kYGzzhZRbK+x0o=
Sec-WebSocket-Protocol: chat
```

Se a conexão WebSocket for desconectada pelo cliente, servidor ou interrupção da rede, os aplicativos cliente deverão reiniciar a conexão com o servidor.

Cabeçalhos WebSocket recomendados

Para evitar problemas inesperados relacionados a compressão ao usar o WebSockets, recomendamos que você inclua os seguintes cabeçalhos em uma [política de solicitação de origem](#):

- Sec-WebSocket-Key
- Sec-WebSocket-Version
- Sec-WebSocket-Protocol
- Sec-WebSocket-Accept
- Sec-WebSocket-Extensions

Armazenamento em cache e disponibilidade

É possível usar o CloudFront para reduzir o número de solicitações às quais o servidor de origem deve responder diretamente. Com o armazenamento em cache do CloudFront, mais objetos são fornecidos de pontos de presença do CloudFront que estão mais próximos de seus usuários. Isso reduz a carga no servidor de origem e reduz a latência.

Quanto mais solicitações de cache de borda o CloudFront puder fornecer, menos solicitações de visualizador o CloudFront deve encaminhar à origem para obter a versão mais recente ou uma versão exclusiva de um objeto. Para otimizar o CloudFront para fazer o menor número possível de solicitações para a origem, considere usar um CloudFront Origin Shield. Para obter mais informações, consulte [Usar o Amazon CloudFront Origin Shield](#).

A proporção de solicitações fornecidas diretamente do cache do CloudFront em comparação com todas as solicitações é chamada de taxa de acertos do cache. É possível visualizar a porcentagem de solicitações do visualizador atendidas, não atendidas e de erros no console do CloudFront. Para obter mais informações, consulte [Visualizar relatórios de estatísticas de cache do CloudFront](#).

Vários fatores afetam a taxa de acertos do cache. É possível ajustar a configuração da distribuição do CloudFront para melhorar a taxa de acertos do cache seguindo as orientações em [Aumentar a proporção de solicitações fornecidas diretamente de caches do CloudFront \(taxa de acertos do cache\)](#).

Para saber como adicionar e remover o conteúdo que você quer que seja fornecido pelo CloudFront, consulte [Adicionar, remover ou substituir conteúdo distribuído pelo CloudFront](#).

Tópicos

- [Aumentar a proporção de solicitações fornecidas diretamente de caches do CloudFront \(taxa de acertos do cache\)](#)
- [Usar o Amazon CloudFront Origin Shield](#)
- [Otimizar a alta disponibilidade com o failover de origem do CloudFront](#)
- [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#)
- [Conteúdo em cache com base em parâmetros de string de consulta](#)
- [Conteúdo em cache com base em cookies](#)
- [Conteúdo em cache com base nos cabeçalhos de solicitação](#)

Aumentar a proporção de solicitações fornecidas diretamente de caches do CloudFront (taxa de acertos do cache)

É possível melhorar a performance aumentando a taxa de solicitações do visualizador fornecidas diretamente do cache do CloudFront, em vez de acessar os servidores de origem em busca de conteúdo. Isso é conhecido como melhorar a taxa de acertos de cache.

As seções a seguir explicam como melhorar sua taxa de acertos do cache.

Tópicos

- [Especificar o tempo durante o qual o CloudFront armazena os objetos em cache](#)
- [Usar o Origin Shield](#)
- [Armazenar em cache com base em parâmetros de string de consulta](#)
- [Armazenar em cache com base nos valores dos cookies](#)
- [Armazenar em cache com base nos cabeçalhos de solicitação](#)
- [Remova o cabeçalho Accept-Encoding quando a compactação não for necessária](#)
- [Disponibilize conteúdo de mídia via HTTP](#)

Especificar o tempo durante o qual o CloudFront armazena os objetos em cache

Para aumentar sua taxa de acertos do cache, é possível configurar sua origem para adicionar uma diretiva [Cache-Control max-age](#) aos seus objetos e especificar o maior valor prático para max-age. Quanto menor for a duração do cache, maior será a frequência com que o CloudFront enviará solicitações para a origem a fim de determinar se um objeto foi alterado e para obter a versão mais recente. Você pode complementar max-age com as diretivas `stale-while-revalidate` e `stale-if-error` para melhorar ainda mais a taxa de acerto do cache sob certas condições. Para ter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Usar o Origin Shield

O CloudFront Origin Shield pode ajudar a melhorar a taxa de acertos do cache da distribuição do CloudFront, pois ele fornece uma camada adicional de cache à frente da origem. Ao usar o Origin Shield, todas as solicitações de todas as camadas de cache do CloudFront para a origem são

recebidas de um único local. O CloudFront pode recuperar cada objeto usando uma única solicitação do Origin Shield, e todas as outras camadas do cache do CloudFront (pontos de presença e [caches de borda regionais](#)) podem recuperar o objeto do Origin Shield.

Para obter mais informações, consulte [Usar o Amazon CloudFront Origin Shield](#).

Armazenar em cache com base em parâmetros de string de consulta

Configurar o CloudFront para armazenamento em cache com base nos parâmetros de string de consulta poderá melhorar o armazenamento se você fizer o seguinte:

- Configurar o CloudFront para encaminhar somente os parâmetros de string de consulta para os quais a origem retorna objetos exclusivos.
- Usar as mesmas letras (maiúsculas e minúsculas) para todas as instâncias do mesmo parâmetro. Por exemplo, se uma solicitação contiver `parameter1=A` e outra, `parameter1=a`, o CloudFront encaminhará solicitações separadas para sua a quando uma solicitação contiver `parameter1=A` e `parameter1=a`. Depois, o CloudFront armazena separadamente em cache os objetos correspondentes retornados pela origem, mesmo que eles sejam idênticos. Se você usar apenas A ou a, o CloudFront encaminhará menos solicitações para a origem.
- Indique os parâmetros na mesma ordem. Assim como ocorre com diferenças nas letras, se uma solicitação de um objeto contiver a string de consulta `parameter1=a¶meter2=b` e outra solicitação do mesmo objeto contiver `parameter2=b¶meter1=a`, o CloudFront encaminhará as duas para a origem e armazenará os objetos correspondentes separadamente, mesmo que sejam idênticos. Se você sempre usar a mesma ordem para parâmetros, o CloudFront encaminhará menos solicitações para a origem.

Para obter mais informações, consulte [Conteúdo em cache com base em parâmetros de string de consulta](#). Para revisar as strings de consulta que o CloudFront encaminha para a origem, consulte os valores na coluna `cs-uri-query` dos arquivos de log do CloudFront. Para obter mais informações, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Armazenar em cache com base nos valores dos cookies

Se você configurar o CloudFront para armazenamento em cache baseado nos valores dos cookies, poderá melhorar o armazenamento em cache se:

- Configurar o CloudFront para encaminhar apenas os cookies especificados, em vez de todos os cookies. Para os cookies configurados pelo CloudFront para serem encaminhados à origem, o

CloudFront encaminha todas as combinações de nome e valor de cookie. Depois, armazenará separadamente em cache os objetos retornados pela origem, mesmo se todos forem idênticos.

Por exemplo, imagine que os visualizadores incluam dois cookies em cada solicitação, cada cookie tenha três valores possíveis e todas as combinações de valor de cookie sejam possíveis. O CloudFront encaminha até seis solicitações diferentes para a origem de cada objeto. Se a origem retornar diferentes versões de um objeto com base apenas em um dos cookies, o CloudFront encaminhará mais solicitações para a origem do que o necessário e armazenará em cache várias versões do objeto desnecessariamente.

- Crie comportamentos de cache separados para conteúdo estático e dinâmico, e configure o CloudFront para encaminhar cookies para a origem apenas para conteúdo dinâmico.

Por exemplo, suponha que você tenha apenas um comportamento de cache para a distribuição e que esteja usando a distribuição para conteúdo dinâmico, como arquivos `.js`, e para arquivos `.css`, que raramente são alterados. O CloudFront armazena versões separadas dos seus arquivos `.css` em cache com base nos valores de cookie, de modo que cada ponto de presença do CloudFront encaminhe uma solicitação para a origem de cada novo valor ou combinação de valores de cookie.

Se você criar um comportamento de cache para o qual o padrão de caminho é `*.css` e que o CloudFront não armazena em cache com base nos valores de cookie, o CloudFront encaminhará solicitações de arquivos `.css` para sua origem apenas para a primeira solicitação recebida por um ponto de presença de um arquivo `.css` e para a primeira solicitação após a expiração de um arquivo `.css`.


- Se possível, crie comportamentos de cache separados para conteúdo dinâmico quando os valores de cookie forem exclusivos para cada usuário (como um ID de usuário) e que varie com base em um número menor de valores exclusivos.²

Para obter mais informações, consulte [Conteúdo em cache com base em cookies](#). Para revisar os cookies que o CloudFront encaminha para a origem, consulte os valores na coluna `cs(Cookie)` dos arquivos de log do CloudFront. Para obter mais informações, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Armazenar em cache com base nos cabeçalhos de solicitação

Se você configurar o CloudFront para armazenamento em cache com base nos cabeçalhos de solicitação, poderá melhorar o armazenamento se:

- Configure o CloudFront para encaminhar e armazenar em cache com base somente em cabeçalhos específicos, não em todos. Para os cabeçalhos especificados, o CloudFront encaminhará todas as combinações de nome e valor de cabeçalho. Depois, armazenará separadamente em cache os objetos retornados pela origem, mesmo se todos forem idênticos.

 Note

O CloudFront sempre encaminha para sua origem os cabeçalhos especificados nos seguintes tópicos:

- Como o CloudFront processa e encaminha solicitações para o servidor de origem do Amazon S3 > [Cabeçalhos de solicitação HTTP removidos ou atualizados pelo CloudFront](#)
- Como o CloudFront processa e encaminha solicitações para seu servidor de origem personalizado > [Cabeçalhos de solicitação HTTP e comportamento do CloudFront \(origens do Amazon S3 e personalizadas\)](#)

Ao configurar o CloudFront para armazenamento em cache com base nos cabeçalhos da solicitação, você não altera os cabeçalhos encaminhados por ele, apenas se ele armazenar os objetos com base nos valores de cabeçalho.

- Tente evitar o armazenamento em cache com base nos cabeçalhos de solicitação com um grande número de valores exclusivos.

Por exemplo, para fornecer diferentes tamanhos de uma imagem com base no dispositivo do usuário, não configure o CloudFront para armazenamento em cache com base no cabeçalho `User-Agent`, que tem um grande número de valores possíveis. Em vez disso, configure o CloudFront para cache com base nos cabeçalhos do tipo de dispositivo do CloudFront `CloudFront-Is-Desktop-Viewer`, `CloudFront-Is-Mobile-Viewer`, `CloudFront-Is-SmartTV-Viewer` e `CloudFront-Is-Tablet-Viewer`. Além disso, se você estiver retornando a mesma versão da imagem para tablets e desktops, encaminhe somente o cabeçalho `CloudFront-Is-Tablet-Viewer`, não o `CloudFront-Is-Desktop-Viewer`.

Para ter mais informações, consulte [Conteúdo em cache com base nos cabeçalhos de solicitação](#).

Remova o cabeçalho **Accept-Encoding** quando a compactação não for necessária

Se a compactação não estiver habilitada, porque a origem não é compatível, o CloudFront não é compatível ou o conteúdo não é compactável, você poderá aumentar a taxa de acertos do cache associando um comportamento de cache na distribuição a uma origem que defina o Custom Origin Header da seguinte forma:

- Header name (Nome do cabeçalho): Accept-Encoding
- Header value (Valor do cabeçalho): (mantenha em branco)

Ao usar essa configuração, o CloudFront remove o cabeçalho Accept-Encoding da chave de cache e não o inclui em solicitações de origem. Essa configuração se aplica a todo o conteúdo fornecido pelo CloudFront com a distribuição dessa origem.

Disponibilize conteúdo de mídia via HTTP

Para obter informações sobre como otimizar o conteúdo de vídeo sob demanda (VOD) e de vídeo por streaming, consulte [Vídeo sob demanda e vídeo de transmissão ao vivo com o CloudFront](#).

Usar o Amazon CloudFront Origin Shield

O Amazon CloudFront Origin Shield é uma camada adicional na infraestrutura de armazenamento em cache do CloudFront que ajuda a minimizar a carga da origem, melhorar a disponibilidade e reduzir os custos operacionais. O Amazon CloudFront Origin Shield fornece os seguintes benefícios:

Melhor proporção de acertos de cache

O Origin Shield pode ajudar a melhorar a taxa de acertos do cache da distribuição do CloudFront pois fornece uma camada adicional de cache à frente da origem. Quando você usa o Origin Shield, todas as solicitações de todas as camadas de cache do CloudFront para a origem passam pelo Origin Shield, aumentando a probabilidade de um acerto de cache. O CloudFront pode recuperar cada objeto com uma única solicitação do Origin Shield de origem para a origem, e todas as outras camadas do cache do CloudFront (pontos de presença e [caches de borda regionais](#)) podem recuperar o objeto do Origin Shield.

Carga de origem reduzida

O Origin Shield pode reduzir ainda mais o número de [solicitações simultâneas](#) enviadas à sua origem para o mesmo objeto. As solicitações de um conteúdo que não está no cache do Origin Shield são consolidadas com outras solicitações para o mesmo objeto, resultando em apenas uma solicitação em direção à sua origem. Lidar com menos solicitações na sua origem pode preservar a disponibilidade dela durante picos de carga ou picos de tráfego inesperados e pode reduzir os custos de itens, como empacotamento just-in-time, transformações de imagens e DTO (transferência de dados).

Melhor performance de rede

Ao habilitar o Origin Shield na região da AWS [que tem a menor latência para sua origem](#), é possível obter uma melhor performance de rede. Para origens em uma região da AWS, o tráfego de rede do CloudFront permanece na rede do CloudFront de alta taxa de transferência até sua origem. Para origens fora da AWS, o tráfego de rede do CloudFront permanece na rede do CloudFront até chegar ao Origin Shield, que tem uma conexão de baixa latência com a sua origem.

Você incorrerá cobranças adicionais por usar o Origin Shield. Para obter mais informações, consulte [Definição de preço do CloudFront](#).

Tópicos

- [Casos de uso do Origin Shield](#)
- [Escolher a região da AWS para o Origin Shield](#)
- [Habilitar o Origin Shield](#)
- [Estimar custos do Origin Shield](#)
- [Alta disponibilidade do Origin Shield](#)
- [Como o Origin Shield interage com outros recursos do CloudFront](#)

Casos de uso do Origin Shield

O CloudFront Origin Shield pode ser vantajoso para muitos casos de uso, incluindo os seguintes:

- Visualizadores espalhados em diferentes regiões geográficas
- Origens que fornecem empacotamento just-in-time para streaming ao vivo ou processamento instantâneo de imagens

- Origens no local com restrições de capacidade ou largura de banda
- Cargas de trabalho que usam várias redes de entrega de conteúdo (CDNs)

O Origin Shield pode não ser ideal em outros casos, como conteúdo dinâmico encaminhado por proxy para a origem, conteúdo com pouca capacidade de cache ou conteúdo que é solicitado com pouca frequência.

As seções a seguir explicam os benefícios do Origin Shield para os seguintes casos de uso.

Casos de uso

- [Visualizadores em diferentes regiões geográficas](#)
- [Várias CDNs](#)

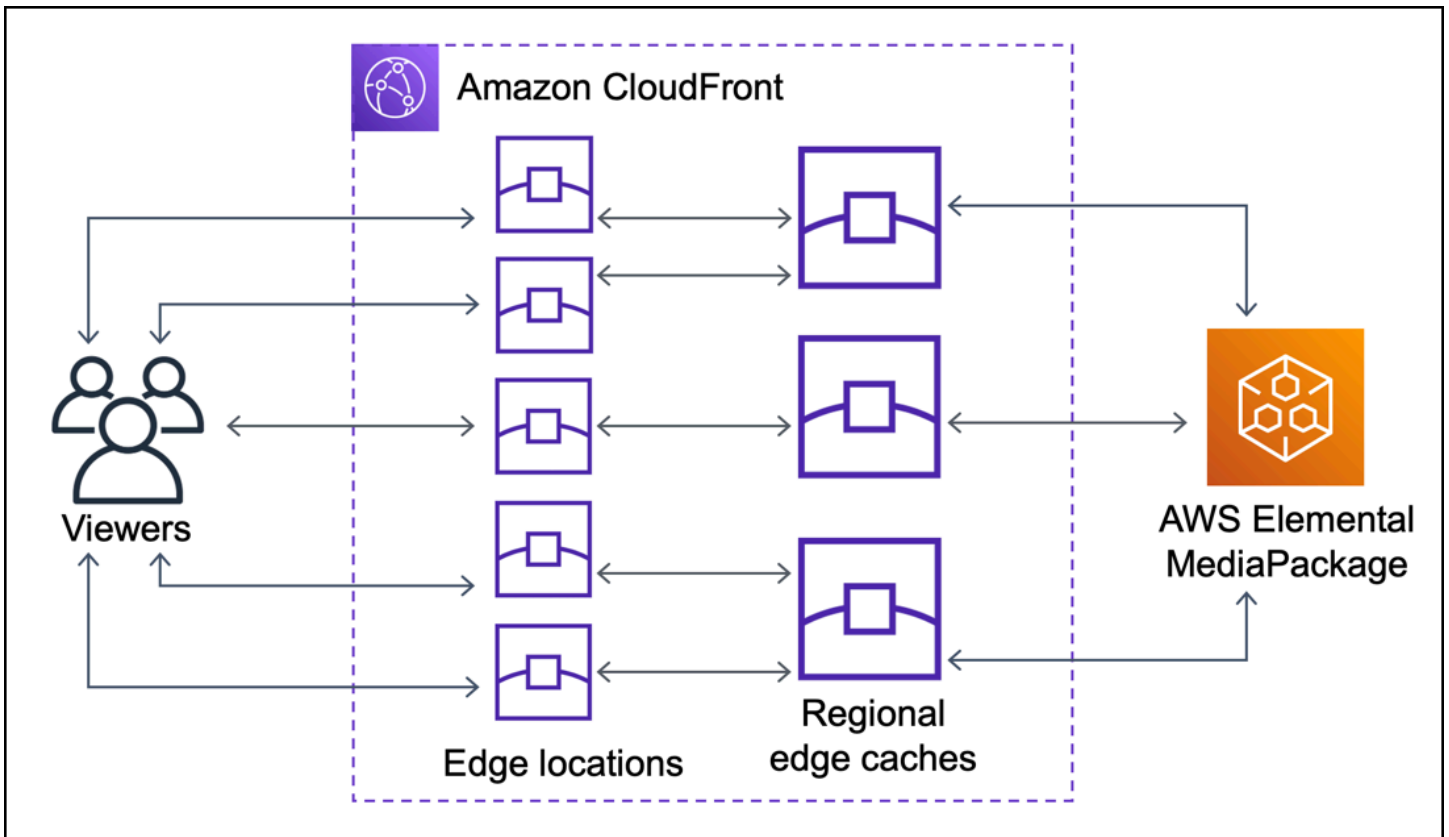
Visualizadores em diferentes regiões geográficas

Com o Amazon CloudFront, você obtém uma carga reduzida que é intrínseca à origem, pois as solicitações que o CloudFront pode atender do cache não vão para a origem. Além da [rede global de pontos de presença](#) do CloudFront, [caches de borda regionais](#) servem como uma camada intermediária de armazenamento em cache para fornecer acertos de cache e consolidar solicitações de origem para visualizadores em regiões geográficas próximas. As solicitações do visualizador são roteadas primeiro para um ponto de presença próximo do CloudFront, e, se o objeto não estiver armazenado em cache nesse local, a solicitação será enviada para um cache de borda regional.

Quando os visualizadores estão em regiões geográficas diferentes, as solicitações podem ser roteadas por diferentes caches de pontos regionais, e cada um deles pode enviar uma solicitação à sua origem para o mesmo conteúdo. Mas com o Origin Shield, você tem uma camada adicional de armazenamento em cache entre os caches de pontos regionais e sua origem. Todas as solicitações de todos os caches de pontos regionais passam pelo Origin Shield, reduzindo ainda mais a carga na sua origem. Os diagramas a seguir ilustram isso. Nos diagramas a seguir, a origem é AWS Elemental MediaPackage.

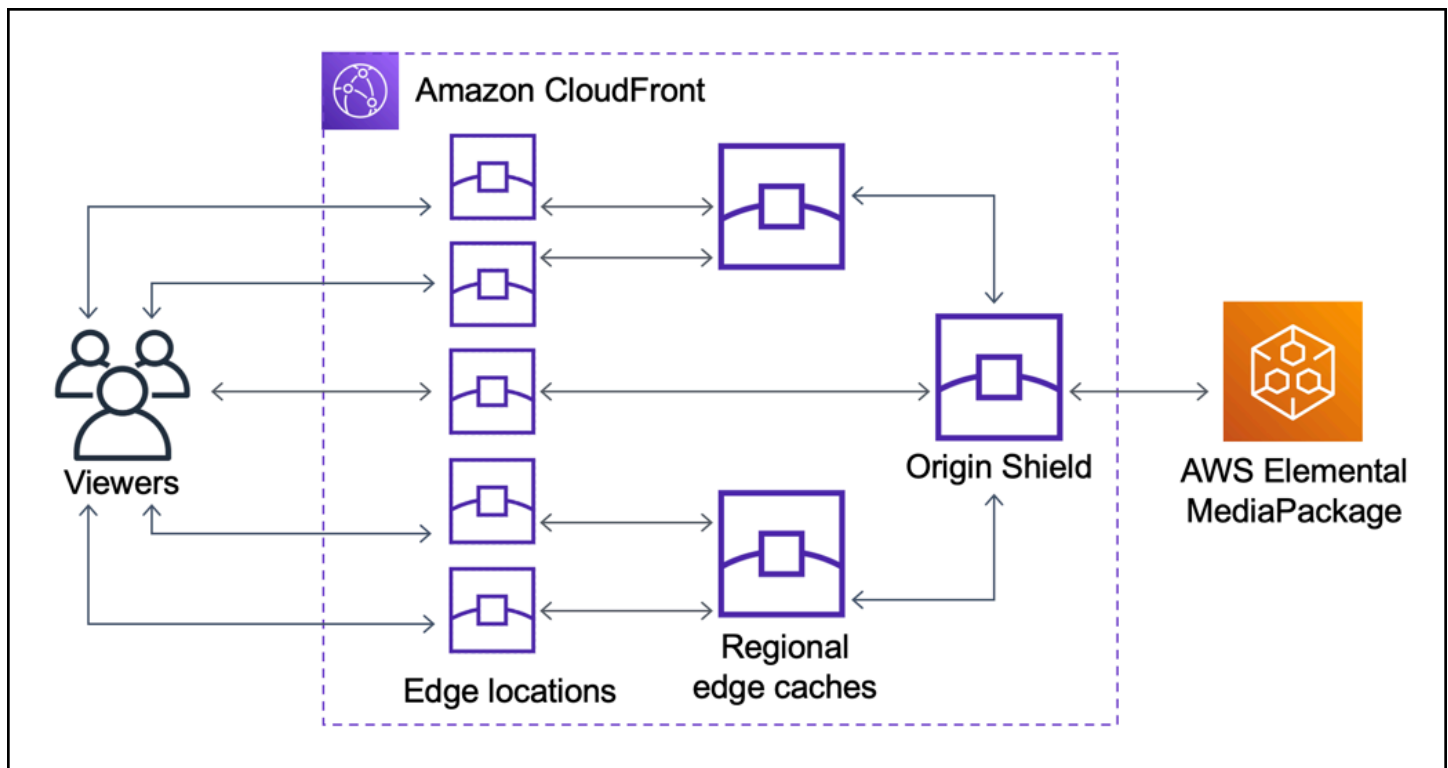
Sem o Origin Shield

Sem o Origin Shield, sua origem pode receber solicitações duplicadas para o mesmo conteúdo, conforme mostrado no diagrama a seguir.



Com o Origin Shield

Usar o Origin Shield pode ajudar a reduzir a carga na sua origem, como mostrado no diagrama a seguir.




Várias CDNs

Para veicular eventos de vídeo ao vivo ou conteúdo sob demanda popular, é possível usar várias redes de entrega de conteúdo (CDNs). Usar várias CDNs pode oferecer certas vantagens, mas também significa que sua origem pode receber muitas solicitações duplicadas para o mesmo conteúdo, cada uma proveniente de CDNs diferentes ou locais diferentes dentro da mesma CDN. Essas solicitações redundantes podem afetar negativamente a disponibilidade de sua origem ou causar custos operacionais adicionais para processos como empacotamento just-in-time ou transferência de dados (DTO) para a Internet.

Ao combinar o Origin Shield usando a distribuição do CloudFront como a origem para outras CDNs, é possível obter os seguintes benefícios:

- Menos solicitações redundantes recebidas na sua origem, o que ajuda a reduzir os efeitos negativos do uso de várias CDNs.
- Uma [chave de cache](#) comum entre CDNs e gerenciamento centralizado para recursos voltados para a origem.
- Melhor performance de rede. O tráfego de rede de outras CDNs é encerrado em um ponto de presença próximo do CloudFront, o que pode fornecer um acerto do cache local. Se o objeto solicitado não estiver no ponto de presença de cache, a solicitação para a origem permanecerá na

rede do CloudFront até chegar ao Origin Shield, o que fornece alta taxa de transferência e baixa latência para a origem. Se o objeto solicitado estiver no cache do Origin Shield, a solicitação para sua origem será totalmente evitada.

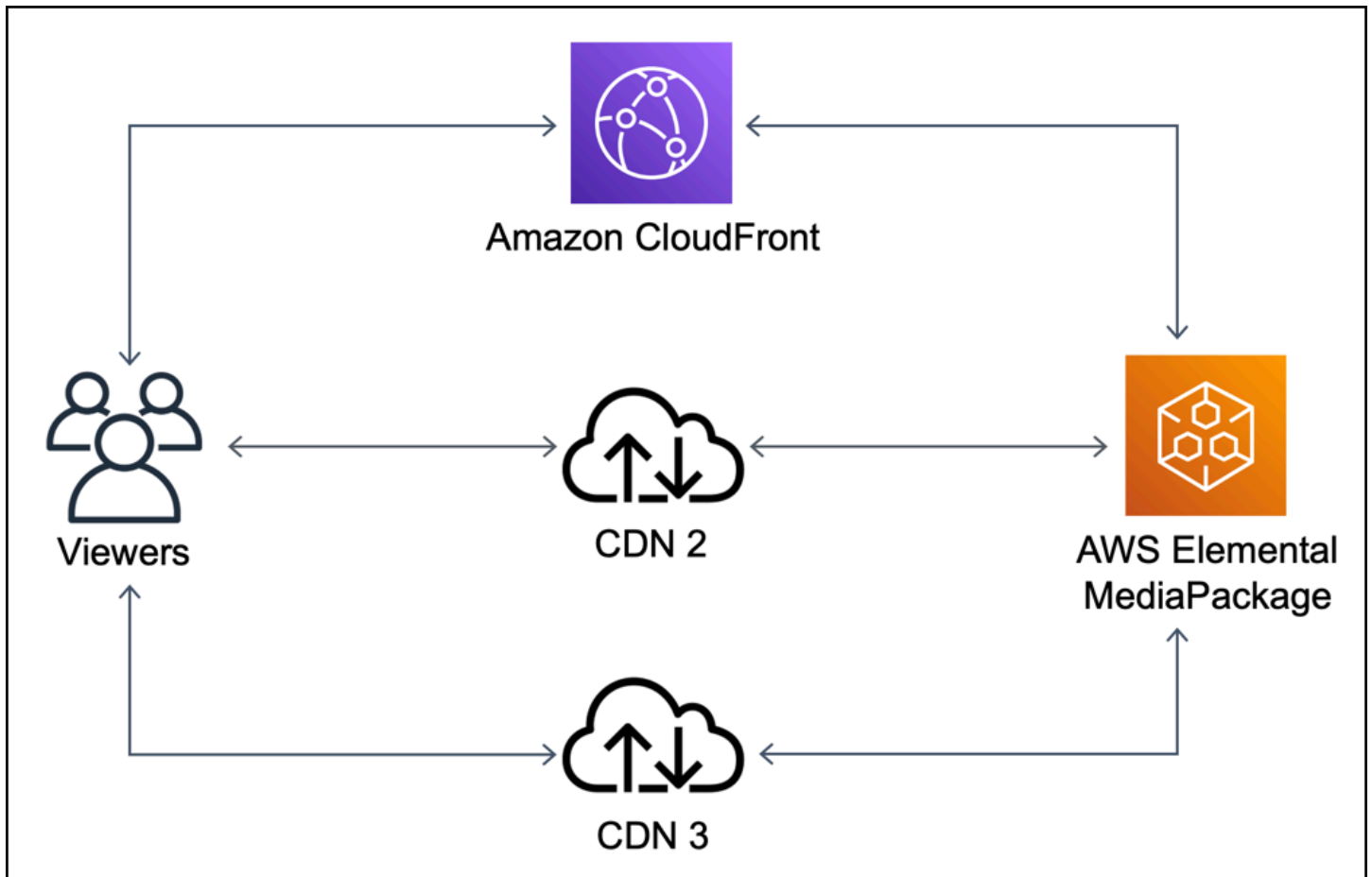
 Important

Se estiver interessado em usar o Origin Shield em uma arquitetura de várias CDNs e tiver preços com desconto, [entre em contato conosco](#) ou com seu representante de vendas da AWS para obter mais informações. Podem se aplicar cobranças adicionais.

Os diagramas a seguir mostram como essa configuração pode ajudar a minimizar a carga em sua origem quando você fornece eventos de vídeo ao vivo populares com várias CDNs. Nos diagramas a seguir, a origem é AWS Elemental MediaPackage.

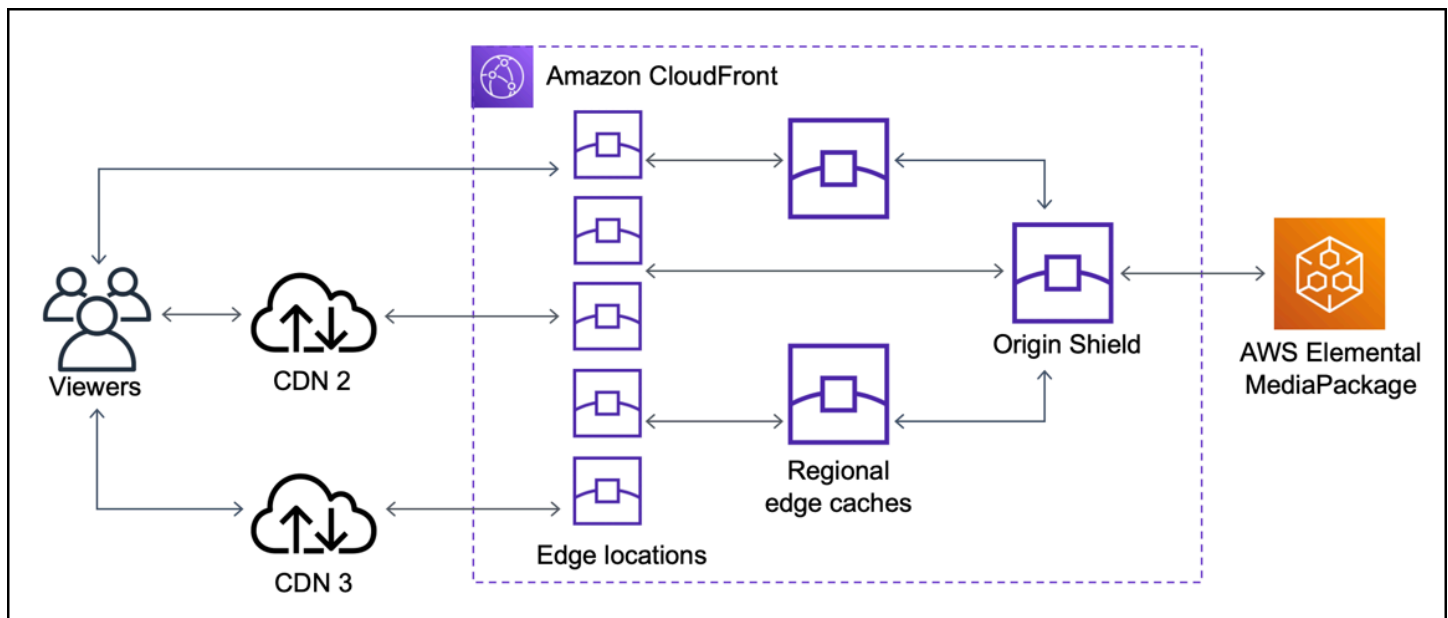
Sem o Origin Shield (várias CDNs)

Sem o Origin Shield, sua origem pode receber muitas solicitações duplicadas para o mesmo conteúdo, cada uma proveniente de uma CDN diferente, conforme mostrado no diagrama a seguir.



Com o Origin Shield (várias CDNs)

O uso do Origin Shield, com o CloudFront como a origem para as outras CDNs, pode ajudar a reduzir a carga na origem, conforme mostrado no diagrama a seguir.



Escolher a região da AWS para o Origin Shield

O Amazon CloudFront oferece o Origin Shield nas regiões da AWS em que o CloudFront tem um [cache de borda regional](#). Ao habilitar o Origin Shield, escolha a região da AWS para o Origin Shield. Recomendamos escolher a região da AWS que tem a menor latência para sua origem. É possível usar o Origin Shield com origens que estão em uma região da AWS e com origens que não estão na AWS.

Para origens em uma região da AWS

Se a sua origem estiver em uma região da AWS, primeiro determine se a origem está em uma região na qual o CloudFront oferece o Origin Shield. O CloudFront oferece o Origin Shield nas seguintes regiões da AWS.

- US East (Ohio) – us-east-2
- Leste dos EUA (Norte da Virgínia): us-east-1
- US West (Oregon) – us-west-2
- Asia Pacific (Mumbai) – ap-south-1
- Ásia-Pacífico (Seul) – ap-northeast-2
- Asia Pacific (Singapore) – ap-southeast-1
- Ásia-Pacífico (Sydney) – ap-southeast-2
- Ásia-Pacífico (Tóquio) (ap-northeast-1)

- Europe (Frankfurt) – eu-central-1
- Europe (Ireland) – eu-west-1
- Europe (London) – eu-west-2
- South America (São Paulo) – sa-east-1

Se a sua origem estiver em uma região da AWS na qual o CloudFront oferece o Origin Shield

Se a origem estiver em uma região da AWS em que o CloudFront oferece o Origin Shield (consulte a lista anterior), habilite o Origin Shield na mesma região que a sua origem.

Se a sua origem não estiver em uma região da AWS na qual o CloudFront oferece o Origin Shield

Se a sua origem não estiver em uma região da AWS em que o CloudFront oferece o Origin Shield, consulte a tabela a seguir para determinar em qual região habilitar o Origin Shield.

Se a sua origem está em...	Habilitar o Origin Shield em ...
US West (N. California) – us-west-1	US West (Oregon) – us-west-2
Africa (Cape Town) – af-south-1	Europe (Ireland) – eu-west-1
Asia Pacific (Hong Kong) – ap-east-1	Asia Pacific (Singapore) – ap-southeast-1
Canada (Central) – ca-central-1	US East (N. Virginia) – us-east-1
Europe (Milan) – eu-south-1	Europe (Frankfurt) – eu-central-1
Europe (Paris) – eu-west-3	Europe (London) – eu-west-2
Europe (Stockholm) – eu-north-1	Europe (London) – eu-west-2
Middle East (Bahrain) – me-south-1	Asia Pacific (Mumbai) – ap-south-1

Para origens fora da AWS

É possível usar o Origin Shield com uma origem no local ou não em uma região da AWS. Nesse caso, habilite o Origin Shield na região da AWS que tem a menor latência de sua origem. Se você não tiver certeza de qual região da AWS tem a menor latência de sua origem, use as sugestões a seguir para ajudá-lo a fazer uma determinação.

- É possível consultar a tabela anterior para obter uma aproximação de qual região da AWS pode ter a menor latência em relação à sua origem, com base na localização geográfica da sua origem.
- É possível executar instâncias do Amazon EC2 em algumas regiões da AWS diferentes que estão geograficamente próximas da sua origem e executar alguns testes usando ping para medir as latências de rede típicas entre essas regiões e a sua origem.

Habilitar o Origin Shield

É possível habilitar o Origin Shield para melhorar sua taxa de acertos de cache, reduzir a carga em sua origem e ajudar a melhorar a performance. Para habilitar o Origin Shield, altere as configurações de origem em uma distribuição do CloudFront. O Origin Shield é uma propriedade da origem. Para cada origem em suas distribuições do CloudFront, é possível habilitar separadamente o Origin Shield em qualquer região da AWS que forneça a melhor performance para essa origem.

É possível habilitar o Origin Shield no console do CloudFront com o AWS CloudFormation ou com a API do CloudFront.

Console

Como habilitar o Origin Shield para uma origem existente (console)

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha a distribuição que tem a origem que você deseja atualizar.
3. Escolha a guia Origins and Origin Groups (Origens e grupos de origem).
4. Escolha a origem a ser atualizada e escolha Edit (Editar).
5. Em Enable Origin Shield (Habilitar o Origin Shield), escolha Yes (Sim).
6. Para a Origin Shield Region (Região do Origin Shield), escolha a região da AWS em que você deseja habilitar o Origin Shield. Para obter ajuda na escolha de uma região, consulte [Escolher a região da AWS para o Origin Shield](#).
7. Na parte inferior da página, escolha Yes, Edit (Sim, editar).

Quando seu status de distribuição for Deployed (Implantado), o Origin Shield estará pronto. Isso leva alguns minutos.

Como habilitar o Origin Shield para uma nova origem (console)

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Para criar a nova origem em uma distribuição existente, faça o seguinte:
 1. Escolha a distribuição em que deseja criar a origem.
 2. Escolha Create Origin (Criar origem) e avance para a etapa 3.

Para criar a nova origem em uma nova distribuição, faça o seguinte:

1. Escolha Criar distribuição.
2. Na seção Web escolha Get Started (Conceitos básicos). Na seção Origin Settings (Configurações de origem), conclua as etapas a seguir, começando com a etapa 3.
3. Em Enable Origin Shield (Habilitar o escudo de origem), escolha Yes (Sim).
4. Para a Origin Shield Region (Região do Origin Shield), escolha a região da AWS em que você deseja habilitar o Origin Shield. Para obter ajuda na escolha de uma região, consulte [Escolher a região da AWS para o Origin Shield](#).

Se você estiver criando uma nova distribuição, continue configurando sua distribuição, usando as outras configurações na página. Para obter mais informações, consulte [Referência de configurações da distribuição](#).

5. Salve suas alterações escolhendo Create (Criar) (para uma nova origem em uma distribuição existente) ou Create Distribution (Criar distribuição) (para uma nova origem em uma nova distribuição).

Quando seu status de distribuição for Deployed (Implantado), o escudo de origem estará pronto. Isso leva alguns minutos.

AWS CloudFormation

Para habilitar o Origin Shield com o AWS CloudFormation, use a propriedade `OriginShield` no tipo de propriedade `Origin` em um recurso `AWS::CloudFront::Distribution`. É possível adicionar a propriedade `OriginShield` a um `Origin` existente ou incluí-la ao criar um novo `Origin`.

O exemplo a seguir mostra a sintaxe, no formato YAML, para habilitar `OriginShield` na região Oeste dos EUA (Oregon) (`us-west-2`). Para obter ajuda na escolha de uma região, consulte [the](#)

[section called “Escolher a região da AWS para o Origin Shield”](#). Este exemplo mostra apenas o tipo de propriedade `Origin`, não o recurso `AWS::CloudFront::Distribution` inteiro.

```
Origins:
- DomainName: 3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com
  Id: Example-EMP-3ae97e9482b0d011
  OriginShield:
    Enabled: true
    OriginShieldRegion: us-west-2
  CustomOriginConfig:
    OriginProtocolPolicy: match-viewer
    OriginSSLProtocols: TLSv1
```

Para obter mais informações, consulte [AWS::CloudFront::Origem da distribuição](#) na seção de referência de recursos e propriedade do Guia do usuário do AWS CloudFormation.

API

Para habilitar o Origin Shield com a API do CloudFront usando os AWS SDKs ou a AWS Command Line Interface (AWS CLI), use o tipo `OriginShield`. Especifique `OriginShield` em uma `Origin` em uma `DistributionConfig`. Para obter informações sobre o tipo `OriginShield`, consulte as seguintes informações na Referência da API do Amazon CloudFront.

- [OriginShield](#) (tipo)
- [Origin](#) (tipo)
- [DistributionConfig](#) (tipo)
- [UpdateDistribution](#) (operação)
- [CreateDistribution](#) (operação)

A sintaxe específica para usar esses tipos e operações varia com base no cliente SDK, CLI ou API. Para obter mais informações, consulte a documentação de referência do SDK, da CLI ou do cliente.

Estimar custos do Origin Shield

Você acumula cobranças do Origin Shield com base no número de solicitações que vão para o Origin Shield como camada incremental.

Para solicitações dinâmicas (não armazenáveis em cache) encaminhadas por proxy para a origem, o Origin Shield é sempre uma camada incremental. As solicitações dinâmicas usam os métodos HTTP: PUT, POST, PATCH, e DELETE.

As solicitações GET e HEAD com uma configuração de vida útil (TTL) de menos de 3600 segundos são consideradas solicitações dinâmicas. Além disso, as solicitações GET e HEAD que desabilitaram o armazenamento em cache também são consideradas solicitações dinâmicas.

Para estimar suas cobranças pelo Origin Shield para solicitações dinâmicas, use a seguinte fórmula:

Número total de solicitações dinâmicas x Origin Shield por 10.000 solicitações / 10.000

Para solicitações não dinâmicas com métodos HTTP GET, HEAD e OPTIONS, o Origin Shield às vezes é uma camada incremental. Ao habilitar o Origin Shield, escolha a Região da AWS para o Origin Shield. Para solicitações que naturalmente vão para o [cache de borda regional](#) na mesma região que o Origin Shield, o Origin Shield não é uma camada incremental. Você não acumula cobranças do Origin Shield para essas solicitações. Para solicitações que vão para um cache de borda regional em uma região diferente do Origin Shield e que, depois, vão para o Origin Shield, o Origin Shield é uma camada incremental. Você acumula cobranças do Origin Shield para essas solicitações.

Para estimar suas cobranças pelo Origin Shield para solicitações que podem ser armazenadas em cache, use a seguinte fórmula:

Número total de solicitações que podem ser armazenadas em cache x (1 - taxa de acertos do cache) x taxa de acertos do cache que acessam o Origin Shield em um cache de borda regional x carga do Origin Shield por 10.000 solicitações / 10.000

Para obter mais informações sobre a cobrança por 10.000 solicitações do Origin Shield, consulte [Definição de preço do CloudFront](#).

Alta disponibilidade do Origin Shield

O Origin Shield utiliza o atributo de [caches de borda regional](#) do CloudFront. Cada um desses caches de borda é criado em uma região da AWS usando pelo menos três [Zonas de disponibilidade](#) com frotas de instâncias do Amazon EC2 Auto Scaling. As conexões de locais do CloudFront com o Origin Shield também usam rastreamento de erro ativo para cada solicitação, para encaminhar automaticamente a solicitação para um local secundário do Origin Shield se o local primário do Origin Shield não estiver disponível.

Como o Origin Shield interage com outros recursos do CloudFront

As seções a seguir explicam como o Origin Shield interage com outros recursos do CloudFront.

Origin Shield e registro em log do CloudFront

Para ver quando o Origin Shield processou uma solicitação, é necessário habilitar uma das seguintes opções:

- [Registros em log padrão do CloudFront \(logs de acesso\)](#). Os logs padrão são fornecidos gratuitamente.
- [Logs em tempo real do CloudFront](#). Você incorrerá cobranças adicionais por usar logs em tempo real. Consulte [Definição de preço do Amazon CloudFront](#)

Os acertos de cache do Origin Shield são exibidos como `OriginShieldHit` no campo `x-edge-detailed-result-type` nos logs do CloudFront. O Origin Shield utiliza os [caches de borda](#) do Amazon CloudFront. Se uma solicitação for roteada de um ponto de presença do CloudFront para o cache de borda regional que está agindo como Origin Shield, ela será relatada como um `Hit` nos logs, não como um `OriginShieldHit`.

Origin Shield e grupos de origem

O Origin Shield é compatível com [Grupos de origem do CloudFront](#). Como o Origin Shield é uma propriedade da origem, as solicitações sempre passam pelo Origin Shield para cada origem, mesmo quando a origem faz parte de um grupo de origem. Para uma determinada solicitação, o CloudFront roteia a solicitação para a origem primária no grupo de origem por meio do Origin Shield da origem primária. Se essa solicitação falhar (de acordo com os critérios de failover do grupo da origem), o CloudFront roteará a solicitação para a origem secundária por meio do Origin Shield da origem secundária.

Origin Shield e Lambda@Edge

O Origin Shield não afeta a funcionalidade das funções do [Lambda@Edge](#) mas pode afetar a região da AWS em que essas funções são executadas.

Ao usar o Origin Shield com o Lambda@Edge, [triggers voltados para a origem](#) (solicitação e resposta de origem) são executados na região da AWS em que o Origin Shield está habilitado. Se o local primário do Origin Shield não estiver disponível e o CloudFront encaminhar as solicitações

para um local secundário do Origin Shield, os acionadores de origem do Lambda@Edge também mudarão para usar o local secundário do Origin Shield.

Os triggers voltados para o visualizador não são afetados.

Otimizar a alta disponibilidade com o failover de origem do CloudFront

Também é possível configurar o CloudFront failover de origem para cenários que exigem alta disponibilidade. Para começar, crie um grupo de origem com duas origens: uma primária e outra secundária. Se a origem primária estiver indisponível ou retornar códigos de status de resposta HTTP específicos que indiquem falha, o CloudFront alternará automaticamente para a origem secundária.

Para configurar o failover de origem, você deve ter uma distribuição com pelo menos duas origens. Depois, crie um grupo de origens para a distribuição que inclua duas origens, definindo uma delas como a primária. Por último, crie ou atualize um comportamento de cache para usar o grupo de origem.

Para consultar as etapas da configuração de grupos de origens e configurar opções específicas de failover de origem, consulte [Criar um grupo de origem](#).

Depois de você configurar o failover de origem para um comportamento de cache, o CloudFront faz o seguinte para solicitações do visualizador:

- Quando houver um acerto de cache, o CloudFront retornará o objeto solicitado.
- Quando houver um erro de cache, o CloudFront roteará a solicitação para a origem primária no grupo de origem.
- Quando a origem primária retornar um código de status que não esteja configurado para failover, como um código de status HTTP 2xx ou 3xx, o CloudFront fornecerá o objeto solicitado ao visualizador.
- Quando ocorrer uma das seguintes situações:
 - A origem primária retorna um código de status HTTP que você configurou para failover
 - O CloudFront não consegue se conectar à origem primária
 - A resposta da origem primária demora muito (atinge o tempo limite)

Assim, o CloudFront roteia a solicitação para a origem secundária do grupo de origem.

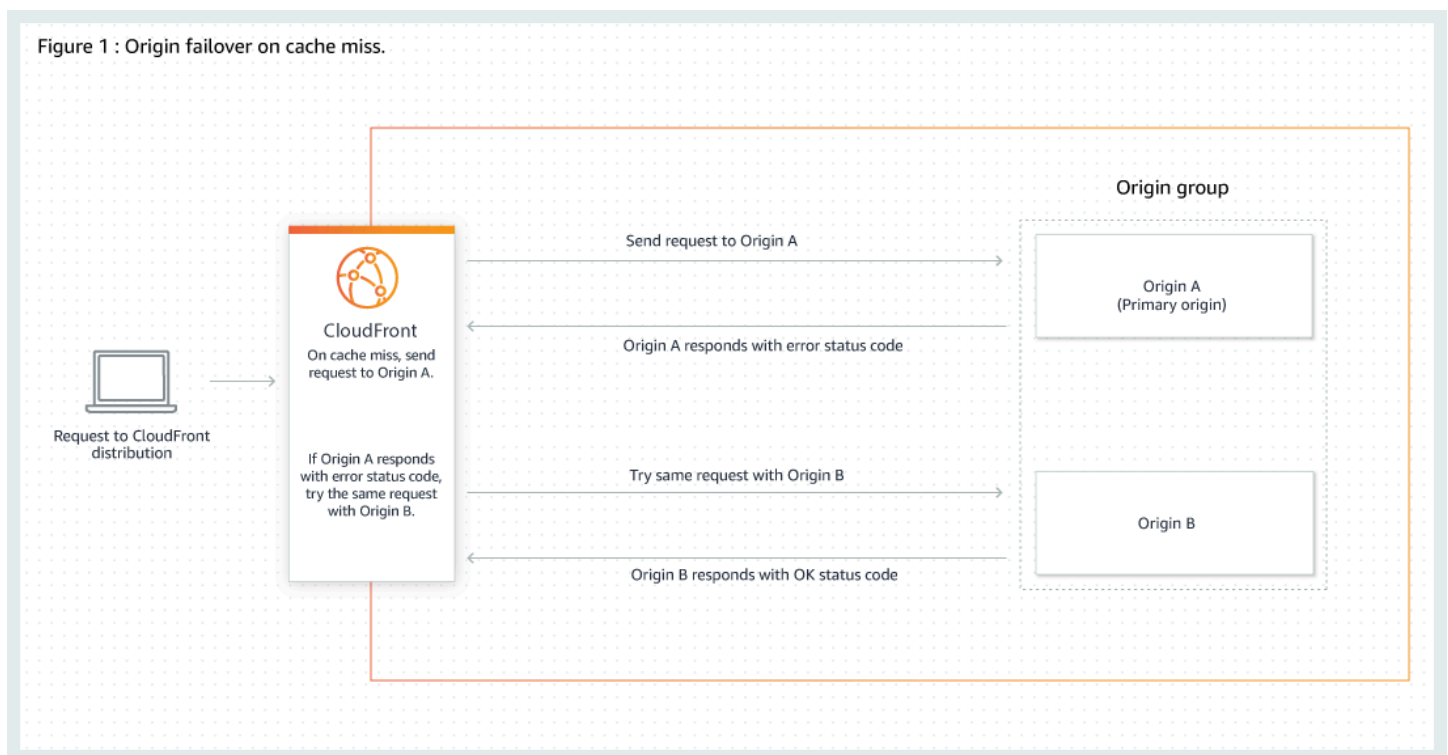
Note

Em alguns casos de uso, como conteúdo de vídeo por streaming, convém que o CloudFront faça failover para a origem secundária rapidamente. Para ajustar a rapidez com que o CloudFront faz failover para a origem secundária, consulte [Controlar tempos limite e tentativas da origem](#).

O CloudFront roteia todas as solicitações recebidas para a origem primária, mesmo quando uma solicitação anterior tenha feito failover para a origem secundária. O CloudFront só envia solicitações para a origem secundária após uma falha de uma solicitação para a origem primária.

O CloudFront faz failover para a origem secundária somente quando o método HTTP da solicitação do visualizador for GET, HEAD ou OPTIONS. O CloudFront não faz failover quando o visualizador envia um método HTTP diferente (por exemplo POST, PUT etc.).

O diagrama a seguir mostra como o failover de origem funciona.

**Tópicos**

- [Criar um grupo de origem](#)

- [Controlar tempos limite e tentativas da origem](#)
- [Uso do failover de origem com funções do Lambda@Edge](#)
- [Usar páginas de erro personalizadas com failover de origem](#)

Criar um grupo de origem

Para criar um grupo de origens

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha a distribuição para a qual deseja criar o grupo de origem.
3. Escolha a guia Origins (Origens).
4. Certifique-se de que a distribuição tenha mais de uma origem. Se ela não tiver, adicione uma segunda origem.
5. Na guia Origins (Origem), no painel Origin groups (Grupos de origens), escolha Create origin group (Criar grupo de origens).
6. Escolha as origens para o grupo de origem. Depois de adicioná-las, use as setas para definir a prioridade, ou seja, qual será a origem primária e qual será a secundária.
7. Insira um nome para o grupo de origens.
8. Escolha os códigos de status de HTTP a serem usados como critérios de failover. Você pode escolher qualquer combinação dos seguintes códigos de status: 400, 403, 404, 416, 500, 502, 503 ou 504. Quando o CloudFront recebe uma resposta com um dos códigos de status especificados, ele executa o failover para a origem secundária.

Note

O CloudFront faz failover para a origem secundária somente quando o método HTTP da solicitação do visualizador for GET, HEAD ou OPTIONS. O CloudFront não faz failover quando o visualizador envia um método HTTP diferente (por exemplo POST, PUT etc.).

9. Escolha Create origin group (Criar grupo de origens).

Certifique-se de atribuir seu grupo de origem como origem para o comportamento de cache distribuído. Para ter mais informações, consulte [Nome](#).

Controlar tempos limite e tentativas da origem

Por padrão, o CloudFront tenta se conectar à origem primária em um grupo de origem por até 30 segundos (3 tentativas de conexão de 10 segundos cada) antes de fazer failover para a origem secundária. Para alguns casos de uso, como conteúdo de vídeo por streaming, talvez você queira que o CloudFront faça failover para a origem secundária mais rapidamente. É possível ajustar as configurações a seguir para afetar a rapidez com que o CloudFront faz failover para a origem secundária. Se a origem for uma origem secundária ou uma origem que não faça parte de um grupo de origem, essas configurações afetarão a rapidez com que o CloudFront retorna uma resposta HTTP 504 ao visualizador.

Para que o failover seja feito mais rapidamente, especifique um tempo limite de conexão mais curto, menos tentativas de conexão ou ambos. Para origens personalizadas (incluindo origens de bucket do Amazon S3 que são configuradas com hospedagem de site estático), também é possível ajustar o tempo limite de resposta da origem.

Tempo limite de conexão da origem

A configuração de tempo limite de conexão da origem afeta o tempo de espera do CloudFront ao tentar estabelecer uma conexão com a origem. Por padrão, o CloudFront aguarda 10 segundos para estabelecer uma conexão, mas é possível especificar de 1 a 10 segundos (inclusive). Para obter mais informações, consulte [Tempo limite da conexão](#).

Tentativas de conexão da origem

A configuração de tentativas de conexão da origem afeta o número de vezes que o CloudFront tenta se conectar à origem. Por padrão, o CloudFront tenta se conectar três vezes, mas é possível especificar de uma a três (inclusive). Para obter mais informações, consulte [Tentativas de conexão](#).

Para uma origem personalizada (incluindo um bucket do Amazon S3 configurado com hospedagem de site estático), essa configuração também afeta o número de vezes que o CloudFront tenta obter uma resposta da origem caso o tempo limite de resposta da origem expire.

Tempo limite da e resposta da origem

Note

Isto se aplica apenas a origens personalizadas.

A configuração do tempo limite de resposta da origem afeta o tempo que o CloudFront espera para receber uma resposta (ou para receber a resposta completa) da origem. Por padrão, o CloudFront aguarda 30 segundos, mas é possível especificar de 1 a 60 segundos (inclusive).” Para obter mais informações, consulte [Tempo limite de resposta \(somente origens personalizadas\)](#).

Como alterar essas configurações

Como alterar essas configurações no [console do CloudFront](#)

- Para uma nova origem ou uma nova distribuição, especifique esses valores ao criar o recurso.
- Para uma origem existente em uma distribuição existente, especifique esses valores ao editar a origem.

Para obter mais informações, consulte [Referência de configurações da distribuição](#).

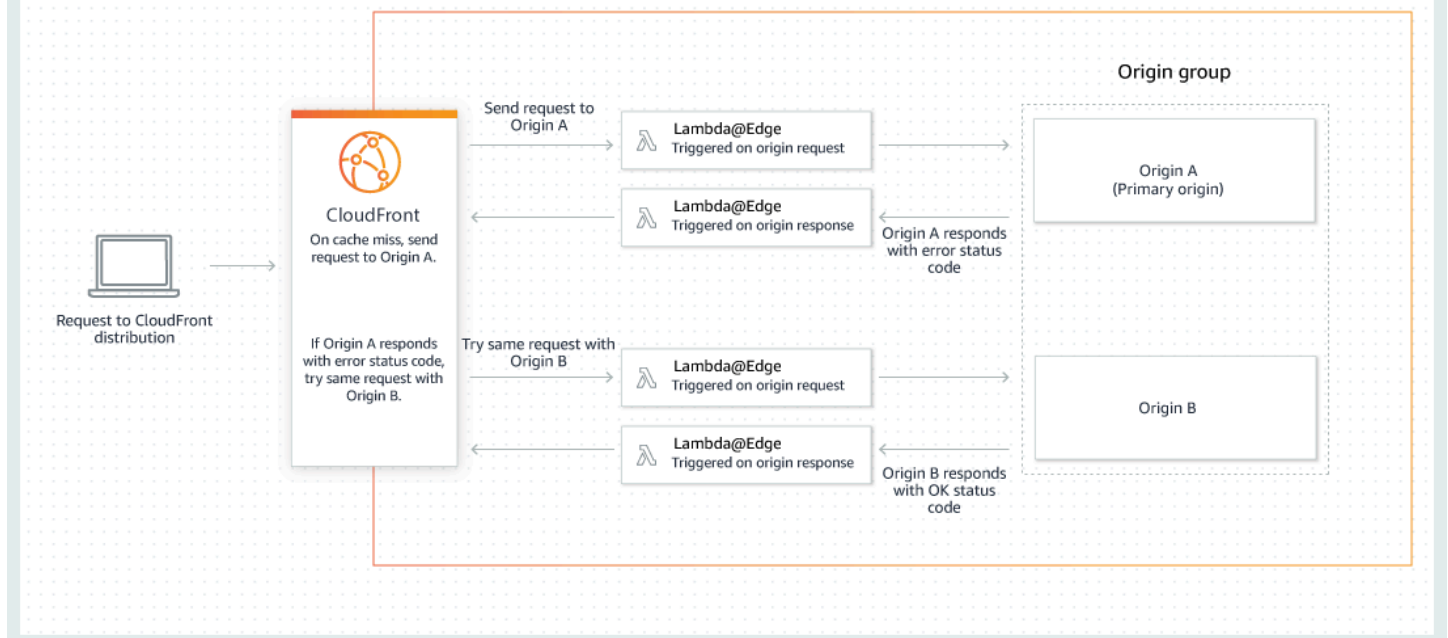
Uso do failover de origem com funções do Lambda@Edge

Você pode usar as funções do Lambda@Edge com distribuições do CloudFront que tiver configurado com grupos de origem. Para usar uma função do Lambda, especifique-a em um [trigger de solicitação origem ou resposta de origem](#) de um grupo de origem ao criar o comportamento de cache. Ao usar uma função do Lambda@Edge com um grupo de origem, ela poderá ser acionada duas vezes para uma única solicitação de visualizador. Por exemplo, considere este cenário:

1. Você cria uma função do Lambda@Edge com um trigger de solicitação de origem.
2. A função do Lambda é acionada quando o CloudFront envia uma solicitação para a origem primária (em um falha de cache).
3. A origem primária responde com um código de status de HTTP configurado para failover.
4. A função do Lambda é acionada novamente quando o CloudFront envia a mesma solicitação à origem secundária.

O diagrama a seguir ilustra como o failover de origem funciona quando você inclui uma função do Lambda@Edge em uma solicitação de origem ou trigger de resposta.

Figure 2 : Origin failover with Lambda@Edge functions triggered on origin request and response events.



Para obter mais informações sobre como usar triggers do Lambda@Edge, consulte [the section called “Adicionar acionadores para uma função do Lambda@Edge”](#).

Para ter mais informações sobre como gerenciar failover de DNS, consulte [Configurar failover de DNS](#) no Guia do desenvolvedor do Amazon Route 53.

Usar páginas de erro personalizadas com failover de origem

Você pode usar as páginas de erro personalizadas com grupos de origens de forma semelhante à forma como você usá-los com origens que não são configuradas para o failover de origem.

Ao usar o failover de origem, você pode configurar o CloudFront para retornar uma página de erro personalizada para a origem primária ou secundária (ou ambas):

- Retornar uma página de erro personalizada para a origem primária: se a origem primária retornar um código de status HTTP que não estiver configurado para failover, o CloudFront retornará a página de erro personalizada aos visualizadores.
- Retornar uma página de erro personalizada à origem secundária: se o CloudFront receber um código de status de falha da origem secundária, o CloudFront retornará a página de erro personalizada.

Para obter mais informações sobre como usar páginas de erro personalizadas com o CloudFront, consulte [Gerar respostas a erros personalizadas](#).

Gerenciar o tempo de permanência do conteúdo no cache (expiração)

É possível controlar o tempo de permanência dos arquivos em um cache do CloudFront antes de o CloudFront encaminhar outra solicitação para a origem. A diminuição da duração permite fornecer conteúdo dinâmico. O aumento da duração significa que os usuários obtêm melhor performance, pois é mais provável que seus arquivos sejam fornecidos diretamente do cache de borda. Uma duração maior também reduz a carga na origem.

Normalmente, o CloudFront fornece um arquivo de um ponto de presença até passar a duração do cache especificada, ou seja, até o arquivo expirar. Depois de expirar, na próxima vez em que o local da borda receber uma solicitação para o arquivo, o CloudFront encaminhará a solicitação à origem a fim de verificar se o cache contém a versão mais recente do arquivo. A resposta da origem depende se o arquivo foi alterado ou não:

- Se o cache do CloudFront já tiver a versão mais recente, a origem retornará um código de status `304 Not Modified`.
- Se o cache do CloudFront não tiver a versão mais recente, a origem retornará um código de status `200 OK` e a versão mais recente do arquivo.

Se um arquivo em um ponto de presença não for solicitado com frequência, o CloudFront poderá excluí-lo (removê-lo antes da data de expiração) para criar espaço para os arquivos solicitados mais recentemente.

Por padrão, cada arquivo expira automaticamente após 24 horas, mas você pode alterar o comportamento padrão de duas formas:

- Para alterar a duração do cache de todos os arquivos com o mesmo padrão de caminho, você pode alterar as configurações do CloudFront para Minimum TTL (TTL mínimo), Maximum TTL (TTL máximo) e Default TTL (TTL padrão) para um comportamento de cache. Para obter informações sobre configurações individuais, consulte [TTL mínimo](#), [TTL máximo](#) e [TTL padrão](#) em [the section called “Configurações de distribuição”](#).
- Para alterar a duração do cache de um arquivo individual, é possível configurar a origem para adicionar um cabeçalho `Cache-Control` com a diretiva `max-age` ou `s-maxage`, ou um

cabeçalho Expires ao arquivo. Para ter mais informações, consulte [Usar cabeçalhos para controlar a duração do cache para objetos individuais](#).

Para obter mais informações sobre como Minimum TTL (TTL mínimo), Default TTL (TTL padrão) e Maximum TTL (TTL máximo) interagem com as diretivas max-age e s-maxage e o campo de cabeçalho Expires, consulte [the section called “Especificar por quanto tempo o CloudFront armazena os objetos em cache”](#).

Você também pode controlar o tempo de permanência de erros (por exemplo, 404 Not Found) em um cache do CloudFront antes de o CloudFront tentar obter novamente o objeto solicitado encaminhando outra solicitação para a origem. Para ter mais informações, consulte [the section called “Como o CloudFront processa códigos de status HTTP 4xx e 5xx da origem”](#).

Tópicos

- [Usar cabeçalhos para controlar a duração do cache para objetos individuais](#)
- [Fornecer conteúdo obsoleto \(expirado\)](#)
- [Especificar por quanto tempo o CloudFront armazena os objetos em cache](#)
- [Adicionar cabeçalhos aos objetos usando o console do Amazon S3](#)

Usar cabeçalhos para controlar a duração do cache para objetos individuais

Você pode usar os cabeçalhos Cache-Control e Expires para controlar o tempo de permanência dos objetos no cache. As configurações de Minimum TTL, Default TTL e Maximum TTL também afetam a duração do cache, mas esta é uma visão geral de como os cabeçalhos podem afetar a duração do cache:

- A diretiva Cache-Control max-age permite especificar o tempo de permanência (em segundos) de um objeto no cache antes de o CloudFront obtê-lo novamente do servidor de origem. O tempo mínimo de expiração é compatível com o CloudFront é de 0 segundos. O valor máximo é de 100 anos. Especifique o valor no seguinte formato:

Cache-Control: max-age=*segundos*

Por exemplo, a seguinte diretiva solicita que o CloudFront mantenha o objeto associado no cache por 3600 segundos (uma hora):

Cache-Control: max-age=3600

Para que os objetos permaneçam nos caches de borda do CloudFront por uma duração diferente da permanência nos caches do navegador, use as diretivas `Cache-Control max-age` e `Cache-Control s-maxage` em conjunto. Para obter mais informações, consulte [Especificar por quanto tempo o CloudFront armazena os objetos em cache](#).

- O campo de cabeçalho `Expires` permite especificar uma data e hora de expiração usando o formato especificado em [RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1 Section 3.3.1, Full Date](#), por exemplo:

```
Sat, 27 Jun 2015 23:59:59 GMT
```

Recomendamos o uso da diretiva `Cache-Control max-age`, em vez do campo de cabeçalho `Expires`, para controlar o armazenamento de objetos em cache. Se você especificar os valores de `Cache-Control max-age` e de `Expires`, o CloudFront usará somente o valor de `Cache-Control max-age`.

Para obter mais informações, consulte [Especificar por quanto tempo o CloudFront armazena os objetos em cache](#).

Não é possível usar os campos de cabeçalho HTTP `Cache-Control` ou `Pragma` em uma solicitação GET de um visualizador para forçar o CloudFront a voltar ao servidor de origem para obter o objeto. O CloudFront ignora esses campos de cabeçalho em solicitações do visualizador.

Para obter mais informações sobre os campos de cabeçalho `Cache-Control` e `Expires`, consulte as seguintes seções em RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1:

- [Section 14.9 Cache Control](#)
- [Section 14.21 Expires](#)

Fornecer conteúdo obsoleto (expirado)

O CloudFront oferece suporte às diretivas de controle de cache `Stale-While-Revalidate` e `Stale-If-Error`.

- A diretiva `stale-while-revalidate` permite que o CloudFront forneça conteúdo obsoleto do cache enquanto busca de forma assíncrona uma nova versão da origem. Isso melhora a latência, pois os usuários recebem respostas imediatamente dos locais de borda do CloudFront, sem

precisar esperar pela busca em segundo plano, e novos conteúdos são carregados em segundo plano para futuras solicitações.

No exemplo a seguir, o CloudFront armazena a resposta em cache por uma hora (`max-age=3600`). Se uma solicitação for feita após esse período, o CloudFront fornecerá o conteúdo obsoleto e, ao mesmo tempo, enviará uma solicitação à origem para revalidar e atualizar o conteúdo em cache. O conteúdo obsoleto é exibido por até 10 minutos (`stale-while-revalidate=600`) enquanto o conteúdo está sendo revalidado.

```
Cache-Control: max-age=3600, stale-while-revalidate=600
```

Note

O CloudFront fornecerá o conteúdo obsoleto até o valor da diretiva `stale-while-revalidate` ou o valor de TTL máximo do CloudFront, o que for menor.

- A diretiva `stale-if-error` permite que o CloudFront forneça conteúdo obsoleto do cache se a origem estiver inacessível ou retornar o código de erro que esteja entre 500 e 600. Isso garante que os visualizadores possam acessar o conteúdo mesmo durante uma interrupção na origem.

No exemplo a seguir, o CloudFront armazena a resposta em cache por uma hora (`max-age=3600`). Se a origem estiver inativa ou retornar um erro após esse período, o CloudFront continuará veiculando o conteúdo obsoleto por até 24 horas (`stale-if-error=86400`).

```
Cache-Control: max-age=3600, stale-if-error=86400
```

Note

O CloudFront fornecerá o conteúdo obsoleto até o valor da diretiva `stale-if-error` ou o valor de TTL máximo do CloudFront, o que for menor.

Quando as respostas de erro `stale-if-error` e [personalizadas](#) estiverem configuradas, o CloudFront primeiro tentará fornecer o conteúdo obsoleto se um erro for encontrado dentro da duração especificada de `stale-if-error`. Se o conteúdo obsoleto não estiver disponível ou o conteúdo estiver além da duração de `stale-if-error`, o CloudFront exibirá as respostas de erro personalizadas configuradas para o código de status de erro correspondente.

Usar os dois juntos

`stale-while-revalidate` e `stale-if-error` são diretivas independentes de controle de cache que podem ser usadas juntas para reduzir a latência e adicionar um buffer para que sua origem responda ou se recupere.

No exemplo a seguir, o CloudFront armazena a resposta em cache por uma hora (`max-age=3600`). Se uma solicitação for feita após esse período, o CloudFront exibirá o conteúdo obsoleto por até 10 minutos (`stale-while-revalidate=600`) enquanto o conteúdo está sendo revalidado. Se o servidor de origem retornar um erro enquanto o CloudFront tenta revalidar o conteúdo, o CloudFront continuará servindo o conteúdo obsoleto por até 24 horas (`stale-if-error=86400`).

```
Cache-Control: max-age=3600, stale-while-revalidate=600, stale-if-error=86400
```

Tip

O armazenamento em cache é um equilíbrio entre desempenho e atualização. Usar diretivas como `stale-while-revalidate` e `stale-if-error` pode melhorar o desempenho e a experiência do usuário, mas certifique-se de que as configurações estejam alinhadas com a atualização do seu conteúdo. As diretivas de conteúdo obsoletas são mais adequadas para casos de uso em que o conteúdo precisa ser atualizado, mas ter a versão mais recente não é essencial. Além disso, se seu conteúdo não mudar ou raramente mudar, o `stale-while-revalidate` poderá adicionar solicitações de rede desnecessárias. Em vez disso, considere definir uma longa duração de cache.

Especificar por quanto tempo o CloudFront armazena os objetos em cache

Para controlar a quantidade de tempo que o CloudFront mantém um objeto no cache antes de enviar outra solicitação para a origem, você pode:

- Defina os valores de TTL mínimo, máximo e padrão no comportamento de cache de uma distribuição do CloudFront. Você pode definir esses valores em uma [política de cache](#) anexada ao comportamento de cache (recomendado) ou nas configurações de cache herdadas.

- Inclua os cabeçalhos `Cache-Control` ou `Expires` nas respostas da origem. Esses cabeçalhos também ajudam a determinar por quanto tempo um navegador mantém um objeto no cache do navegador antes de enviar outra solicitação para o CloudFront.

A tabela a seguir explica como os cabeçalhos `Cache-Control` e `Expires` enviados da origem funcionam em conjunto com as configurações de TTL em um comportamento de cache para afetar o cache.

Cabeçalhos de origem	TTL mínimo = 0	TTL mínimo > 0
<p>A origem adiciona uma Cache-Control: max-age diretiva ao objeto</p>	<p>Armazenamento em cache do CloudFront</p> <p>O CloudFront armazena objetos em cache pelo valor da diretiva <code>Cache-Control: max-age</code> ou pelo valor do TTL máximo do CloudFront, o que for menor.</p> <p>Armazenamento em cache no navegador</p> <p>Os navegadores armazenam em cache o objeto para o valor da diretiva <code>Cache-Control: max-age</code>.</p>	<p>Armazenamento em cache do CloudFront</p> <p>O armazenamento em cache do CloudFront depende dos valores do TTL mínimo e do TTL máximo do CloudFront e da diretiva <code>Cache-Control max-age</code>:</p> <ul style="list-style-type: none"> • Se $TTL\ mínimo < max-age < TTL\ máximo$, o CloudFront armazena em cache o objeto para o valor da diretiva <code>Cache-Control: max-age</code>. • Se $max-age < TTL\ mínimo$, o CloudFront armazena em cache o objeto para o valor do TTL mínimo do CloudFront. • Se $max-age > TTL\ máximo$, o CloudFront armazena em cache o objeto para o

Cabeçalhos de origem	TTL mínimo = 0	TTL mínimo > 0
		<p>valor do TTL máximo do CloudFront.</p> <p>Armazenamento em cache no navegador</p> <p>Os navegadores armazenam em cache o objeto para o valor da diretiva <code>Cache-Control: max-age</code> .</p>
<p>A origem não adiciona uma Cache-Control: max-age diretiva ao objeto</p>	<p>Armazenamento em cache do CloudFront</p> <p>O CloudFront armazena em cache o objeto para o valor do TTL padrão do CloudFront.</p> <p>Armazenamento em cache no navegador</p> <p>Depende do navegador.</p>	<p>Armazenamento em cache do CloudFront</p> <p>O CloudFront armazena objetos em cache pelo valor do TTL máximo do CloudFront ou TTL padrão, o que for maior.</p> <p>Armazenamento em cache no navegador</p> <p>Depende do navegador.</p>

Cabeçalhos de origem	TTL mínimo = 0	TTL mínimo > 0
<p>A origem adiciona diretivas Cache-Control: max-age e Cache-Control: s-maxage ao objeto</p>	<p>Armazenamento em cache do CloudFront</p> <p>O CloudFront armazena objetos em cache pelo valor da diretiva Cache-Control: s-maxage ou pelo valor do TTL máximo do CloudFront, o que for menor.</p> <p>Armazenamento em cache no navegador</p> <p>Os navegadores armazenam em cache o objeto para o valor da diretiva Cache-Control max-age.</p>	<p>Armazenamento em cache do CloudFront</p> <p>O armazenamento em cache do CloudFront depende dos valores do TTL mínimo e do TTL máximo do CloudFront e da diretiva Cache-Control: s-maxage:</p> <ul style="list-style-type: none"> • Se $TTL\ mínimo < s-maxage < TTL\ máximo$, o CloudFront armazena em cache o objeto para o valor da diretiva Cache-Control: s-maxage. • Se $s-maxage < TTL\ mínimo$, o CloudFront armazena em cache o objeto para o valor do TTL mínimo do CloudFront. • Se $s-maxage > TTL\ máximo$, o CloudFront armazena em cache o objeto para o valor do TTL máximo do CloudFront. <p>Armazenamento em cache no navegador</p>

Cabeçalhos de origem	TTL mínimo = 0	TTL mínimo > 0
		Os navegadores armazenam em cache o objeto para o valor da diretiva <code>Cache-Control: max-age</code> .

Cabeçalhos de origem	TTL mínimo = 0	TTL mínimo > 0
<p>A origem adiciona um cabeçalho Expires no objeto</p>	<p>Armazenamento em cache do CloudFront</p> <p>O CloudFront armazena o objeto até a data no cabeçalho Expires ou para o valor do TTL máximo do CloudFront, o que ocorrer antes.</p> <p>Armazenamento em cache no navegador</p> <p>Os navegadores armazenam o objeto em cache até a data no cabeçalho Expires.</p>	<p>Armazenamento em cache do CloudFront</p> <p>O armazenamento em cache do CloudFront depende dos valores do TTL mínimo e máximo do CloudFront e do cabeçalho Expires:</p> <ul style="list-style-type: none"> • Se $TTL\ mínimo < Expires < TTL\ máximo$, o CloudFront armazena o objeto até a data e hora no cabeçalho Expires. • Se $Expires < TTL\ mínimo$, o CloudFront armazena em cache o objeto para o valor do TTL mínimo do CloudFront. • Se $Expires > TTL\ máximo$, o CloudFront armazena em cache o objeto para o valor do TTL máximo do CloudFront. <p>Armazenamento em cache no navegador</p> <p>Os navegadores armazenam o objeto em cache até a data</p>

Cabeçalhos de origem	TTL mínimo = 0	TTL mínimo > 0
		e hora no Expires cabeçalho.
A origem adiciona as diretivas Cache-Control: no-cache, no-store e/ou private ao objeto	O CloudFront e os navegadores respeitam os cabeçalhos.	<p>Armazenamento em cache do CloudFront</p> <p>O CloudFront armazena em cache o objeto para o valor do TTL mínimo do CloudFront. Veja o aviso abaixo desta tabela.</p> <p>Armazenamento em cache no navegador</p> <p>Os navegadores respeitam os cabeçalhos.</p>

Warning

Se seu TTL mínimo for maior que 0, o CloudFront usa o TTL mínimo da política de cache, mesmo que as diretivas Cache-Control: no-cache, no-store e/ou private estejam presentes nos cabeçalhos de origem.

Se a origem for alcançável, o CloudFront obterá o objeto da origem e o retornará ao visualizador.

Se a origem estiver inacessível e o valor do TTL mínimo ou máximo for maior que 0, o CloudFront atenderá ao objeto obtido da origem anteriormente.

Para evitar esse comportamento, inclua a Cache-Control: stale-if-error=0 diretiva com o objeto retornado da origem. Isso faz com que o CloudFront retorne um erro em resposta a solicitações futuras se a origem não for alcançável, em vez de retornar o objeto obtido da origem anteriormente.

Para mais informações sobre como alterar as configurações de distribuições usando o console do CloudFront, consulte [Atualizar uma distribuição](#). Para mais informações sobre como alterar as configurações de distribuições usando a API do CloudFront, consulte [UpdateDistribution](#).

Adicionar cabeçalhos aos objetos usando o console do Amazon S3

Como adicionar um campo de cabeçalho **Cache-Control** ou **Expires** aos objetos do Amazon S3 usando o console do Amazon S3

1. Faça login no AWS Management Console e abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. Na lista de buckets, escolha o nome do bucket que contém os arquivos nos quais você está adicionando cabeçalhos.
3. Marque a caixa de seleção ao lado do nome do arquivo ou pasta em que você está adicionando cabeçalhos. Quando você adiciona cabeçalhos a uma pasta, isso afeta todos os arquivos dentro dela.
4. Escolha Actions (Ações), em seguida, Edit metadata (Editar metadados).
5. No painel Add metadata (Adicionar metadados), faça o seguinte:
 - a. Escolha Add Metadata (Adicionar metadados).
 - b. Em Type (Tipo), escolha System Defined (Definido pelo sistema).
 - c. Em Key (Chave), escolha o nome do cabeçalho que você está adicionando (Cache-Control ou Expires).
 - d. Em Value (Valor), insira um valor de cabeçalho. Por exemplo, para um cabeçalho Cache-Control, você pode inserir `max-age=86400`. Em Expires, você pode inserir uma data de expiração e hora, como `Wed, 30 Jun 2021 09:28:00 GMT`.
6. Na parte inferior da página, escolha Edit metadata (Editar metadados).

Conteúdo em cache com base em parâmetros de string de consulta

Alguns aplicativos web usam query strings para enviar informações para a origem. Uma string de consulta é a parte de uma solicitação da web que aparece após um caractere ?. Ela pode conter um ou mais parâmetros separados por caracteres &. No seguinte exemplo, a string de consulta inclui dois parâmetros, *color = red* e *size = grandes*:

<https://d1111111abcdef8.cloudfront.net/images/image.jpg?color=red&size=large>

Para distribuições, você pode escolher se deseja que o CloudFront encaminhe strings de consulta para a origem e se deve armazenar o conteúdo em cache com base em todos os parâmetros ou em parâmetros selecionados. Por que isso pode ser útil? Considere o seguinte exemplo.

Imagine que seu site esteja disponível em cinco idiomas. A estrutura do diretório e os nomes de arquivo de todas as cinco versões do site são idênticos. Quando um usuário visualiza seu site, as solicitações encaminhadas para o CloudFront incluem um parâmetro de string de consulta de idioma com base no idioma escolhido do usuário. É possível configurar o CloudFront para encaminhar strings de consulta para a origem e para armazenamento em cache com base no parâmetro de idioma. Se você configurar o servidor da Web para retornar a versão de uma página correspondente ao idioma selecionado, o CloudFront armazenará em cache a versão de cada idioma separadamente, com base no valor do parâmetro de string de consulta do idioma.

Neste exemplo, se a página principal do site for `main.html`, as cinco solicitações a seguir farão com que o CloudFront armazene `main.html` cinco vezes em cache, uma vez para cada valor do parâmetro da string de consulta do idioma:

- <https://d1111111abcdef8.cloudfront.net/main.html?language=de>
- <https://d1111111abcdef8.cloudfront.net/main.html?language=en>
- <https://d1111111abcdef8.cloudfront.net/main.html?language=es>
- <https://d1111111abcdef8.cloudfront.net/main.html?language=fr>
- <https://d1111111abcdef8.cloudfront.net/main.html?language=jp>

Observe o seguinte:

- Alguns servidores HTTP não processam parâmetros de query string e, portanto, não retornam diferentes versões de um objeto com base nos valores de parâmetro. Para essas origens, se você configurar o CloudFront para encaminhar parâmetros de string de consulta para a origem, o CloudFront armazenará em cache com base nos valores dos parâmetros, mesmo que a origem retorne versões idênticas do objeto ao CloudFront para cada valor de parâmetro.
- Para que os parâmetros de string de consulta funcionem conforme descrito no exemplo acima com os idiomas, é necessário usar o caractere `&` como delimitador entre os parâmetros da string de consulta. Se você usar um delimitador diferente, poderá obter resultados inesperados, dependendo dos parâmetros que especificar para o CloudFront usar como base para o armazenamento em cache, e a ordem em que os parâmetros serão exibidos na string de consulta.

Os exemplos a seguir mostram o que acontece quando você usa um delimitador diferente e configura o CloudFront para armazenamento em cache com base no parâmetro `color`:

- Na solicitação a seguir, o CloudFront armazena o conteúdo em cache com base no valor do parâmetro `color`, mas interpreta o valor como *red;size=large*:

```
https://d1111111abcdef8.cloudfront.net/images/  
image.jpg?color=red;size=large
```

- Na solicitação a seguir, o CloudFront armazena o conteúdo em cache, mas não baseia o armazenamento em cache nos parâmetros da string de consulta. Isso ocorre porque você configurou o CloudFront para armazenamento em cache com base no parâmetro `color`, mas ele interpreta a seguinte string como contendo apenas um parâmetro `size` com um valor de *large;color=red*:

```
https://d1111111abcdef8.cloudfront.net/images/  
image.jpg?size=large;color=red
```

É possível configurar o CloudFront para realizar uma destas ações:

- Não encaminhar query strings para a origem. Se você não encaminhar strings de consulta, o CloudFront não armazenará em cache com base nos parâmetros da string de consulta.
- Encaminhar query strings para a origem e armazenar em cache com base em todos os parâmetros na query string.
- Encaminhar query strings para a origem e armazenar em cache com base em parâmetros específicos na query string.

Para ter mais informações, consulte [the section called “Otimizar o armazenamento em cache”](#).

Tópicos

- [Configurações do console e da API para encaminhamento e armazenamento de strings de consulta em cache](#)
- [Otimizar o armazenamento em cache](#)
- [Parâmetros de string de consulta e logs padrão do CloudFront \(logs de acesso\)](#)

Configurações do console e da API para encaminhamento e armazenamento de strings de consulta em cache

Para configurar o encaminhamento e o armazenamento da string de consulta em cache no console do CloudFront, consulte as seguintes configurações em [the section called “Configurações de distribuição”](#):

- [the section called “Encaminhamento e armazenamento em cache de strings de consulta”](#)
- [the section called “Lista de permissões de strings de consulta”](#)

Para configurar o encaminhamento e o armazenamento em cache de strings de consulta com a API do CloudFront, consulte [CachePolicy](#) e [OriginRequestPolicy](#) na Referência de API do Amazon CloudFront.

Otimizar o armazenamento em cache

Ao configurar o CloudFront para armazenamento em cache com base nos parâmetros da string de consulta, siga seguir as etapas a seguir para reduzir o número de solicitações que o CloudFront encaminha para a origem. Quando os pontos de presença do CloudFront fornecem objetos, a carga no servidor de origem e a latência são reduzidas, pois os objetos são fornecidos de locais mais próximos dos usuários.

Armazenamento em cache baseado apenas em parâmetros para os quais a origem retorna diferentes versões de um objeto

Para cada parâmetro da string de consulta encaminhado pela aplicação Web para o CloudFront, o CloudFront encaminha solicitações para cada valor de parâmetro para a origem e armazena em cache uma versão separada do objeto para cada valor de parâmetro. Isso será verdadeiro mesmo se a sua origem sempre retornar o mesmo objeto, independentemente do valor do parâmetro. Para vários parâmetros, o número de solicitações e o número de objetos se multiplicam.

Recomendamos configurar o CloudFront para armazenamento em cache com base apenas nos parâmetros da string de consulta para os quais a origem retorna diferentes versões, e considerar cuidadosamente os méritos do armazenamento em cache com base em cada parâmetro. Por exemplo, imagine que você tem um site de varejo. Você tem fotos de uma jaqueta em seis cores diferentes, e ela tem 10 opções de tamanhos diferentes. As imagens que você tem da jaqueta mostram as cores diferentes, mas não os tamanhos. Para otimizar o armazenamento em cache,

configure o CloudFront para armazenamento em cache apenas com base no parâmetro de cor, não de tamanho. Isso aumenta a probabilidade de o CloudFront atender a uma solicitação do cache, melhorando a performance e reduzindo a carga na origem.

Sempre liste os parâmetros na mesma ordem

A ordem dos parâmetros é importante em query strings. No exemplo a seguir, as query strings são idênticas, exceto pelo fato de os parâmetros estarem em ordem diferente. Isso fará com que o CloudFront encaminhe duas solicitações separadas de `image.jpg` para a origem e armazene duas versões separadas do objeto em cache:

- `https://d111111abcdef8.cloudfront.net/images/image.jpg?color=red&size=large`
- `https://d111111abcdef8.cloudfront.net/images/image.jpg?size=large&color=red`

Recomendamos que você sempre indique os nomes de parâmetro na mesma ordem, como em ordem alfabética.

Sempre use a mesma letra (maiúscula ou minúscula) para os nomes e valores de parâmetro

O CloudFront diferencia letras maiúsculas de minúsculas de nomes e valores de parâmetros ao armazenar em cache com base nos parâmetros da string de consulta. No exemplo a seguir, as query strings são idênticas, exceto pelo formato das letras nos nomes e valores do parâmetro. Isso faz com que o CloudFront encaminhe quatro solicitações separadas de `image.jpg` para a origem e armazene quatro versões separadas do objeto em cache:

- `https://d111111abcdef8.cloudfront.net/images/image.jpg?color=red`
- `https://d111111abcdef8.cloudfront.net/images/image.jpg?color=Red`
- `https://d111111abcdef8.cloudfront.net/images/image.jpg?Color=red`
- `https://d111111abcdef8.cloudfront.net/images/image.jpg?Color=Red`

Recomendamos que você use letras maiúsculas/minúsculas de forma consistente nos nomes e valores de parâmetro, como todas as letras minúsculas.

Não use nomes de parâmetro em conflito com signed URLs

Se você estiver usando signed URLs para restringir o acesso ao conteúdo (se tiver adicionado assinantes confiáveis à distribuição), o CloudFront removerá os seguintes parâmetros da string de consulta antes de encaminhar o restante do URL para a origem:

- Expires
- Key-Pair-Id
- Policy
- Signature

Ao usar signed URLs e quiser configurar o CloudFront para encaminhar strings de consulta para a origem, seus próprios parâmetros da string de consulta não podem ser denominados Expires, Key-Pair-Id, Policy ou Signature.

Parâmetros de string de consulta e logs padrão do CloudFront (logs de acesso)

Se você habilitar o registro em log, o CloudFront registrará o URL completo em log, incluindo os parâmetros da string de consulta. Isso será verdadeiro independentemente de você ter configurado o CloudFront para encaminhar strings de consulta para a origem. Para obter mais informações sobre o log do CloudFront, consulte [the section called “Usar logs padrão \(logs de acesso\)”](#).

Conteúdo em cache com base em cookies

Por padrão, o CloudFront não considera cookies ao processar solicitações e respostas, nem ao armazenar os objetos em cache em pontos de presença. Se o CloudFront receber duas solicitações idênticas, exceto pelo que está no cabeçalho Cookie, por padrão, o CloudFront tratará as solicitações como idênticas e retornará o mesmo objeto para as duas solicitações.

É possível configurar o CloudFront para encaminhar à origem alguns ou todos os cookies em solicitações de visualizador e armazenar em cache versões separadas dos objetos com base nos valores de cookie encaminhados. Ao fazer isso, o CloudFront usa alguns ou todos os cookies das solicitações de visualizador (todos os que estiverem configurados para serem encaminhados) a fim de identificar exclusivamente um objeto no cache.

Por exemplo, imagine que as solicitações de `locations.html` contêm um cookie `country` com um valor de `uk` ou `fr`. Ao configurar o CloudFront para armazenar os objetos em cache com base no valor do cookie `country`, o CloudFront encaminhará as solicitações de `locations.html` para a origem e incluirá o cookie `country` e os respectivos valores. Sua origem retornará `locations.html`, e o CloudFront armazenará o objeto em cache uma vez para solicitações em que o valor do cookie `country` é `uk` e uma vez para solicitações em que o valor for `fr`.

⚠ Important

O Amazon S3 e alguns servidores HTTP não processam cookies. Não configure o CloudFront para encaminhar cookies a uma origem que não processe cookies ou que não varie a resposta com base em cookies. Isso pode fazer com que o CloudFront encaminhe mais solicitações para a origem para o mesmo objeto, o que reduz a performance e aumenta a carga na origem. Ao considerar o exemplo anterior, se sua origem não processar o cookie `country` ou sempre retornar a mesma versão de `locations.html` para o CloudFront, independentemente do valor do cookie `country`, não configure o CloudFront para encaminhar esse cookie.

Por outro lado, se a origem personalizada depender de um cookie específico ou enviar respostas diferentes com base em um cookie, configure o CloudFront para encaminhar esse cookie para a origem. Caso contrário, o CloudFront remove o cookie antes de encaminhar a solicitação para a origem.

Para configurar o encaminhamento de cookies, atualize o comportamento de cache da sua distribuição. Para obter mais informações sobre comportamentos de cache, consulte [Configurações de comportamento de cache](#), especificamente as seções [Cookies progressivos](#) e [Lista de permissões de cookies](#).

Você pode configurar cada comportamento de cache para realizar uma das seguintes ações:

- Encaminhar todos os cookies para a origem: o CloudFront inclui todos os cookies enviados pelo visualizador ao encaminhar solicitações para a origem. Quando a origem retorna uma resposta, o CloudFront armazena a resposta em cache usando os nomes e os valores de cookie na solicitação do visualizador. Se a resposta da origem incluir cabeçalhos `Set-Cookie`, o CloudFront os retornará ao visualizador com o objeto solicitado. O CloudFront também armazena os cabeçalhos em cache `Set-Cookie` com o objeto retornado da origem e envia esses cabeçalhos `Set-Cookie` para visualizadores em todos os acertos do cache.
- Encaminhar um conjunto de cookies especificados: o CloudFront remove todos os cookies enviados pelo visualizador que não estejam na lista de permissões antes de encaminhar uma solicitação para a origem. O CloudFront armazena a resposta em cache usando os nomes e valores de cookies na lista na solicitação do visualizador. Se a resposta da origem incluir cabeçalhos `Set-Cookie`, o CloudFront os retornará ao visualizador com o objeto solicitado. O CloudFront também armazena os cabeçalhos em cache `Set-Cookie` com o objeto retornado da origem e envia esses cabeçalhos `Set-Cookie` para visualizadores em todos os acertos do cache.

Para obter informações sobre como especificar curingas em nomes de cookies, consulte [Lista de permissões de cookies](#).

Para saber a cota atual do número de nomes de cookies que você pode encaminhar para cada comportamento de cache ou para solicitar uma cota maior, consulte [Cotas em cadeias de consulta \(configurações de cache herdadas\)](#).

- Não encaminhar cookies para a origem: o CloudFront não armazena os objetos em cache com base no cookie enviado pelo visualizador. Além disso, o CloudFront remove os cookies antes de encaminhar as solicitações para a origem e remove os cabeçalhos Set-Cookie das respostas antes de retorná-las aos visualizadores. Como essa não é uma maneira ideal de usar seus recursos de origem, ao selecionar esse comportamento de cache, certifique-se de que sua origem não inclua cookies nas respostas de origem por padrão.

Observações sobre como especificar os cookies que você deseja encaminhar:

Logs de acesso

Se você configurar o CloudFront para registrar solicitações e cookies em log, o CloudFront registrará todos os cookies e atributos de cookie em log, mesmo que você configure o CloudFront para não encaminhar cookies para a origem ou configure o CloudFront para encaminhar somente cookies específicos. Para obter mais informações sobre o log do CloudFront, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Diferenciação de letras maiúsculas e minúsculas

Os nomes e valores de cookie diferenciam letras maiúsculas e minúsculas. Por exemplo, se o CloudFront estiver configurado para encaminhar todos os cookies, e duas solicitações de visualizador para o mesmo objeto tiverem cookies idênticos, apenas com diferenças de maiúsculas e minúsculas, o CloudFront armazenará o objeto em cache duas vezes.

O CloudFront classifica cookies

Se o CloudFront estiver configurado para encaminhar cookies (todos ou um subconjunto), ele classificará os cookies na ordem natural pelo nome do cookie antes de encaminhar a solicitação para a origem.

Note

Nomes de cookies que comecem com o caractere \$ não são aceitos. O CloudFront removerá o cookie antes de encaminhar a solicitação para a origem. É possível remover o caractere \$ ou especificar outro caractere no início do nome do cookie.

If-Modified-Since e If-None-Match

As solicitações condicionais If-Modified-Since e If-None-Match não são compatíveis com o CloudFront quando ele estiver configurado para encaminhar cookies (todos ou um subconjunto).

O formato padrão do par nome-valor é obrigatório

O CloudFront encaminhará um cabeçalho de cookie somente se o valor estiver em conformidade com o [formato padrão do par nome-valor](#), por exemplo: "Cookie: cookie1=value1; cookie2=value2"

Desativar o armazenamento em cache de cabeçalhos Set-Cookie

Se o CloudFront estiver configurado para encaminhar cookies para a origem (sejam todos ou alguns específicos), ele também armazenará em cache os cabeçalhos Set-Cookie recebidos na resposta da origem. O CloudFront inclui esses cabeçalhos Set-Cookie em sua resposta ao visualizador original e também os inclui em respostas subsequentes que são fornecidas do cache do CloudFront.

Se você quiser receber cookies na origem, mas não quiser que o CloudFront armazene os cabeçalhos Set-Cookie nas respostas da origem, configure a origem para adicionar um cabeçalho Cache-Control com uma diretiva no-cache que especifique Set-Cookie como um nome de campo. Por exemplo: Cache-Control: no-cache="Set-Cookie". Para obter mais informações, consulte [Response Cache-Control Directives](#) no padrão Hypertext Transfer Protocol (HTTP/1.1): Caching.

Tamanho máximo dos nomes de cookies

Se você configurar o CloudFront para encaminhar cookies específicos para sua origem, o número total de bytes em todos os nomes de cookies que o CloudFront foi configurado para encaminhar não poderá exceder 512 menos o número de cookies que serão encaminhados. Por exemplo, se você configurar o CloudFront para encaminhar 10 cookies para a origem, o tamanho total dos nomes de todos eles não poderá ultrapassar 502 bytes (512 a 10).

Se você configurar o CloudFront para encaminhar todos os cookies para a origem, o tamanho dos nomes dos cookies não importará.

Para obter informações sobre como usar o console do CloudFront para atualizar uma distribuição para que o CloudFront encaminhe cookies para a origem, consulte [Atualizar uma distribuição](#). Para obter informações sobre como usar a API do CloudFront para atualizar uma distribuição, consulte [UpdateDistribution](#) na Referência da API do Amazon CloudFront.

Conteúdo em cache com base nos cabeçalhos de solicitação

O CloudFront permite que você escolha se quer que o CloudFront encaminhe cabeçalhos para a origem e armazene em cache versões distintas de um objeto especificado com base nos valores do cabeçalho nas solicitações do visualizador. Isso permite fornecer diferentes versões do seu conteúdo com base no dispositivo que o usuário estiver usando, a localização dele, o idioma usado por ele e uma variedade de outros critérios.

Tópicos

- [Visão geral de cabeçalhos e distribuições](#)
- [Selecionar os cabeçalhos para basear o armazenamento em cache](#)
- [Configurar o CloudFront para respeitar as configurações do CORS](#)
- [Configurar o armazenamento em cache com base no tipo de dispositivo](#)
- [Configurar o armazenamento em cache com base no idioma do visualizador](#)
- [Configurar o armazenamento em cache com base no local do visualizador](#)
- [Configurar o armazenamento em cache com base no protocolo da solicitação](#)
- [Configurar o armazenamento em cache para arquivos compactados](#)
- [Como o armazenamento em cache com base em cabeçalhos afeta a performance](#)
- [Como a letra e os valores do cabeçalho afetam o armazenamento em cache](#)
- [Cabeçalhos que o CloudFront retorna ao visualizador](#)

Visão geral de cabeçalhos e distribuições


Por padrão, o CloudFront não considera cabeçalhos ao armazenar seus objetos em cache em pontos de presença. Se a origem retornar dois objetos e seus valores nos cabeçalhos de solicitação forem diferentes, o CloudFront armazenará apenas uma versão do objeto em cache.

É possível configurar o CloudFront para encaminhar cabeçalhos para a origem, o que faz com que o CloudFront armazene várias versões de um objeto em cache com base nos valores de um ou mais cabeçalhos de solicitação. Para configurar o CloudFront para armazenamento em cache os objetos baseados nos valores de cabeçalhos específicos, você especifica as configurações de comportamento do cache para a sua distribuição. Para obter mais informações, consulte [Cache baseado em cabeçalhos de solicitação selecionados](#).

Por exemplo, imagine que as solicitações do visualizador de `logo.jpg` contêm um cabeçalho `Product` personalizado com um valor de `Acme` ou `Apex`. Ao configurar o CloudFront para armazenar seus objetos em cache com base no valor do cabeçalho `Product`, o CloudFront encaminha solicitações de `logo.jpg` para a origem e inclui o cabeçalho `Product` e os valores de cabeçalho. O CloudFront armazena `logo.jpg` em cache uma vez para solicitações em que o valor do cabeçalho `Product` é `Acme`, e uma vez para solicitações em que o valor é `Apex`.

Você pode configurar cada comportamento de cache em uma distribuição para executar uma das seguintes ações:

- Encaminhar todos os cabeçalhos para sua origem

 Note

Para configurações de cache herdadas: se você configurar o CloudFront para encaminhar todos os cabeçalhos para a origem, ele não armazenará em cache os objetos associados a esse comportamento de cache. Em vez disso, ele enviará todas as solicitações para a origem.

- Encaminhar uma lista específica de cabeçalhos. O CloudFront armazena seus objetos em cache com base nos valores de todos os cabeçalhos especificados. O CloudFront também encaminha os cabeçalhos encaminhados por ele por padrão, mas armazena seus objetos em cache com base apenas nos cabeçalhos especificados.
- Encaminhe somente os cabeçalhos padrão. Nessa configuração, o CloudFront não armazena seus objetos em cache com base nos valores dos cabeçalhos de solicitação.

Para saber a cota atual do número de cabeçalhos que você pode encaminhar para cada comportamento de cache ou para solicitar uma cota maior, consulte [Cotas para cabeçalhos](#).

Para obter informações sobre como usar o console do CloudFront para atualizar uma distribuição para que o CloudFront encaminhe cabeçalhos para a origem, consulte [Atualizar uma distribuição](#).

Para obter informações sobre como usar a API do CloudFront para atualizar uma distribuição existente, consulte [Atualizar distribuição](#) na Referência da API do Amazon CloudFront.

Selecionar os cabeçalhos para basear o armazenamento em cache

Os cabeçalhos que podem ser encaminhados para a origem nos quais o CloudFront baseia o armazenamento em cache dependem de se a origem é um bucket do Amazon S3 ou uma origem personalizada.

- **Amazon S3:** é possível configurar o CloudFront para encaminhar e armazenar os objetos em cache com base em vários cabeçalhos específicos (veja a lista de exceções a seguir). No entanto, recomendamos que você evite encaminhar cabeçalhos com uma origem do Amazon S3, a menos que você precise implementar o compartilhamento de recursos de origem cruzada (CORS) ou queira personalizar o conteúdo usando o Lambda@Edge em eventos voltados para a origem.
- Para configurar o CORS, você deve encaminhar cabeçalhos que permitem que o CloudFront distribua conteúdo para sites habilitados para compartilhamento de recursos de origem cruzada (CORS). Para obter mais informações, consulte [Configurar o CloudFront para respeitar as configurações do CORS](#).
- Para personalizar o conteúdo usando cabeçalhos que você encaminha para a origem do Amazon S3, escreva e adicione funções do Lambda@Edge e associe-as à sua distribuição do CloudFront a ser acionada por um evento voltado para origem. Para obter mais informações sobre como trabalhar com cabeçalhos para personalizar o conteúdo, consulte [Personalizar o conteúdo por cabeçalhos de país ou tipo de dispositivo: exemplos](#).

Evite encaminhar os cabeçalhos que não estiver usando para personalizar o conteúdo, pois o encaminhamento de cabeçalhos extras pode reduzir a proporção de acertos do cache. Ou seja, o CloudFront não poderá atender a muitas solicitações de caches de borda como uma proporção de todas as solicitações.

- **Origem personalizada:** é possível configurar o CloudFront para armazenamento em cache com base no valor de qualquer cabeçalho de solicitação, com exceção de:
 - `Connection`
 - `Cookie`: se você quiser encaminhar e armazenar em cache com base em cookies, use uma configuração separada na distribuição. Para obter mais informações, consulte [Conteúdo em cache com base em cookies](#).
 - `Host` (for Amazon S3 origins)
 - `Proxy-Authorization`

- TE
- Upgrade

É possível configurar o CloudFront para armazenar os objetos em cache com base nos valores dos cabeçalhos `Date` e `User-Agent`, mas não é recomendável fazê-lo. Esses cabeçalhos têm vários valores possíveis, e o armazenamento em cache com base nesses valores pode fazer com que o CloudFront encaminhe significativamente mais solicitações para a origem.

Para obter uma lista completa dos cabeçalhos de solicitação HTTP e a maneira como o CloudFront os processa, consulte [Cabeçalhos de solicitação HTTP e comportamento do CloudFront \(origens do Amazon S3 e personalizadas\)](#).

Configurar o CloudFront para respeitar as configurações do CORS

Se você habilitou o compartilhamento de recursos entre origens (CORS) em um bucket do Amazon S3 ou uma origem personalizada, escolha os cabeçalhos específicos a serem encaminhados, para respeitar as configurações do CORS. Os cabeçalhos que você precisa encaminhar diferem dependendo da origem (Amazon S3 ou personalizada) e de se você quer armazenar respostas `OPTIONS`.

Amazon S3

- Se quiser que as respostas `OPTIONS` sejam armazenadas em cache, faça o seguinte:
 - Escolha as opções para configurações de comportamento de cache padrão que permitem o armazenamento em cache de respostas `OPTIONS`.
 - Configure o CloudFront para encaminhar os seguintes cabeçalhos: `Origin`, `Access-Control-Request-Headers` e `Access-Control-Request-Method`.
- Se não quiser que as respostas `OPTIONS` sejam armazenadas em cache, configure o CloudFront para encaminhar o cabeçalho `Origin` junto com todos os outros cabeçalhos requeridos pela origem (por exemplo, `Access-Control-Request-Headers`, `Access-Control-Request-Method` ou outros).

Origens personalizadas: encaminhe o cabeçalho `Origin` com todos os outros cabeçalhos requeridos pela origem.

Para configurar o CloudFront para armazenamento em cache de respostas com base no CORS, configure-o para encaminhar cabeçalhos usando uma política de cache. Para ter mais informações, consulte [Controlar a chave de cache com uma política](#).

Para obter mais informações sobre o CORS e o Amazon S3, consulte [Uso de compartilhamento de recursos entre origens \(CORS\)](#) no Guia do usuário do Amazon Simple Storage Service.

Configurar o armazenamento em cache com base no tipo de dispositivo

Para que o CloudFront armazene em cache diferentes versões de seus objetos com base no dispositivo que um usuário está usando para visualizar seu conteúdo, configure o CloudFront para encaminhar os cabeçalhos aplicáveis para a origem personalizada:

- `CloudFront-Is-Desktop-Viewer`
- `CloudFront-Is-Mobile-Viewer`
- `CloudFront-Is-SmartTV-Viewer`
- `CloudFront-Is-Tablet-Viewer`

Com base no valor do cabeçalho `User-Agent`, o CloudFront define o valor desses cabeçalhos como `true` ou `false` antes de encaminhar a solicitação para a origem. Se o dispositivo se encaixar em mais de uma categoria, mais de um valor poderá ser `true`. Por exemplo, para alguns tablets, o CloudFront pode definir tanto `CloudFront-Is-Mobile-Viewer` quanto `CloudFront-Is-Tablet-Viewer` como `true`.

Configurar o armazenamento em cache com base no idioma do visualizador

Se você quiser que o CloudFront armazene diferentes versões dos objetos em cache com base no idioma especificado na solicitação, configure o CloudFront para encaminhar o cabeçalho `Accept-Language` para a origem.

Configurar o armazenamento em cache com base no local do visualizador

Se você quiser que o CloudFront armazene em cache diferentes versões de seus objetos com base no país de origem da solicitação, configure-o para encaminhar o cabeçalho `CloudFront-Viewer-Country` para a origem. O CloudFront converte automaticamente o endereço IP da origem da solicitação em um código de país de duas letras. Para obter uma lista de códigos de país fácil de usar, classificável por código e nome do país, consulte a entrada da Wikipédia [ISO 3166-1 alfa-2](#).

Configurar o armazenamento em cache com base no protocolo da solicitação

Se você quiser que o CloudFront armazene em cache diferentes versões de seus objetos com base no protocolo da solicitação, HTTP ou HTTPS, configure o CloudFront para encaminhar o cabeçalho `CloudFront-Forwarded-Proto` para a origem.

Configurar o armazenamento em cache para arquivos compactados

Se a origem for compatível com compactação Brotli, você poderá armazenar em cache com base no cabeçalho `Accept-Encoding`. Configure o armazenamento em cache com base em `Accept-Encoding` somente se a origem fornecer conteúdo diferente com base no cabeçalho.

Como o armazenamento em cache com base em cabeçalhos afeta a performance

Ao configurar o CloudFront para armazenamento em cache com base em um ou mais cabeçalhos, e os cabeçalhos tiverem mais de um valor possível, o CloudFront encaminhará mais solicitações para o servidor de origem para o mesmo objeto. Isso reduz a performance e aumenta a carga no servidor de origem. Se o servidor de origem retornar o mesmo objeto, independentemente do valor de um determinado cabeçalho, recomendamos não configurar o CloudFront para armazenamento em cache com base nesse cabeçalho.

Se você configurar o CloudFront para encaminhar mais de um cabeçalho, a ordem dos cabeçalhos nas solicitações do visualizador não afetará o armazenamento em cache, desde que os valores sejam os mesmos. Por exemplo, se uma solicitação contiver os cabeçalhos `A:1,B:2` e outra solicitação contiver `B:2,A:1`, o CloudFront armazenará em cache apenas uma cópia do objeto.

Como a letra e os valores do cabeçalho afetam o armazenamento em cache

Quando o CloudFront armazena em cache com base nos valores de cabeçalhos, ele não diferencia maiúsculas e minúsculas do nome do cabeçalho, mas diferencia maiúsculas e minúsculas do valor do cabeçalho:

- Se as solicitações do visualizador incluírem `Product:Acme` e `product:Acme`, o CloudFront armazenará um objeto em cache apenas uma vez. A única diferença entre eles é a letra (maiúscula/minúscula) do nome de cabeçalho, que não afeta o armazenamento em cache.

- Se as solicitações do visualizador incluírem `Product : Acme` e `Product : acme`, o CloudFront armazenará um objeto duas vezes em cache, pois o valor será Acme em algumas solicitações e acme em outras.

Cabeçalhos que o CloudFront retorna ao visualizador

Configurar o CloudFront para encaminhar e armazenar cabeçalhos em cache não afeta quais cabeçalhos o CloudFront retorna ao visualizador. O CloudFront retorna todos os cabeçalhos obtidos da origem, com algumas exceções. Para obter mais informações, consulte o tópico aplicável:

- Origens do Amazon S3: consulte [Cabeçalhos de resposta HTTP removidos ou atualizados pelo CloudFront](#).
- Origens personalizadas: consulte [Cabeçalhos de resposta HTTP que o CloudFront remove ou substitui](#).

Controlar a chave de cache com uma política

Com uma política de cache do CloudFront, é possível especificar os cabeçalhos HTTP, cookies e strings de consulta que o CloudFront inclui na chave de cache para objetos armazenados em cache em locais de borda do CloudFront. A chave de cache é o identificador exclusivo de cada objeto no cache e determina se a solicitação HTTP de um visualizador gera um acerto de cache.

Um acerto de cache ocorre quando uma solicitação de visualizador gera a mesma chave de cache de uma solicitação anterior, e o objeto dessa chave de cache está no cache do local da borda e é válido. Quando há um acerto de cache, o objeto é fornecido ao visualizador de um local da borda do CloudFront, o que inclui os seguintes benefícios:

- Carga reduzida no servidor de origem
- Latência reduzida para o visualizador

Incluir menos valores na chave de cache aumenta a probabilidade de um acerto de cache. Isso pode gerar melhor performance do site ou da aplicação porque há uma taxa de acertos de cache maior (uma proporção maior de solicitações do visualizador que ocasiona um acerto de cache). Para ter mais informações, consulte [Noções básicas sobre a chave de cache](#).

Para controlar a chave de cache, use uma política de cache do CloudFront. Anexe uma política de cache para um ou mais comportamentos de cache em uma distribuição do CloudFront.

Você também pode usar a política de cache para especificar configurações de vida útil (TTL) para objetos no cache do CloudFront e habilitar o CloudFront para solicitar e armazenar em cache objetos compactados.

Tópicos

- [Noções básicas sobre políticas de cache](#)
- [Criar políticas de cache](#)
- [Usar políticas de cache gerenciadas](#)
- [Noções básicas sobre a chave de cache](#)

Noções básicas sobre políticas de cache

É possível usar uma política de cache para melhorar a taxa de acertos do cache controlando os valores (strings de consulta de URL, cabeçalhos HTTP e cookies) incluídos na chave de cache. O CloudFront fornece algumas políticas de cache predefinidas, conhecidas como políticas gerenciadas, para casos de uso comuns. É possível usar essas políticas gerenciadas ou criar sua própria política de cache específica para suas necessidades. Para obter mais informações sobre políticas gerenciadas, consulte [Usar políticas de cache gerenciadas](#).

Uma política de cache contém as seguintes configurações, que são categorizadas em informações de política, configurações de vida útil (TTL) e configurações chave de cache.

Informações de política

Nome

Um nome exclusivo para identificar a política de cache. No console, você usa o nome para anexar a política de cache a um comportamento de cache.

Descrição

Um comentário para descrever a política de cache. Isso é opcional, mas pode ajudar a identificar a finalidade da política de cache.

Configurações de vida útil (TTL)

As configurações de vida útil (TTL) funcionam em conjunto com os cabeçalhos HTTP `Cache-Control` e `Expires` (se eles estiverem na resposta da origem) para determinar por quanto tempo os objetos permanecem válidos no cache do CloudFront.

Minimum TTL

O tempo mínimo, em segundos, que você quer que os objetos permaneçam no cache do CloudFront antes que o CloudFront verifique se o objeto foi atualizado na origem. Para ter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Maximum TTL

O tempo máximo, em segundos, que os objetos permanecem no cache do CloudFront antes que o CloudFront envie outra solicitação à origem para verificar se o objeto foi atualizado.

O CloudFront usa essa configuração somente quando a origem envia cabeçalhos Cache-Control ou Expires com o objeto. Para ter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

TTL padrão

O tempo padrão, em segundos, que você quer que os objetos permaneçam no cache do CloudFront antes que o CloudFront verifique se o objeto foi atualizado na origem. O CloudFront usa esse valor de configuração como a TTL do objeto somente quando a origem não envia cabeçalhos Cache-Control ou Expires com o objeto. Para ter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Note

Se as configurações Minimum TTL (TTL mínimo), Maximum TTL (TTL máximo) e Default TTL (TTL padrão) estiverem todas definidas como 0, isso desativará o armazenamento em cache do CloudFront.

Configurações da chave de cache

As configurações da chave de cache especificam os valores nas solicitações do visualizador que o CloudFront inclui na chave de cache. Os valores podem incluir strings de consulta de URL, cabeçalhos HTTP e cookies. Os valores que você inclui na chave de cache são automaticamente incluídos nas solicitações que o CloudFront envia à origem, conhecidas como solicitações de origem. Para obter informações sobre como controlar solicitações de origem sem afetar a chave de cache, consulte [Controlar as solicitações de origem com uma política](#).

As configurações de chave de cache incluem:

- [Cabeçalhos](#)
- [Cookies](#)
- [Strings de consulta](#)
- [Suporte à compactação](#)

Cabeçalhos

Os cabeçalhos HTTP em solicitações do visualizador que o CloudFront inclui na chave de cache e nas solicitações da origem. Para cabeçalhos, é possível escolher uma das seguintes configurações:

- **None (Nenhum):** os cabeçalhos HTTP nas solicitações do visualizador não são incluídos na chave de cache e não são incluídos automaticamente nas solicitações da origem.
- **Include the following headers (Incluir os seguintes cabeçalhos):** você especifica quais dos cabeçalhos HTTP nas solicitações do visualizador serão incluídos na chave de cache e incluídos automaticamente nas solicitações da origem.

Ao usar a configuração **Include the following headers (Incluir os seguintes cabeçalhos)**, você especifica cabeçalhos HTTP pelo nome, não pelo valor. Por exemplo, considere o seguinte cabeçalho HTTP:

```
Accept-Language: en-US,en;q=0.5
```

Nesse caso, você especifica o cabeçalho como `Accept-Language`, não como `Accept-Language: en-US,en;q=0.5`. No entanto, o CloudFront inclui o cabeçalho completo, incluindo seu valor, na chave de cache e nas solicitações da origem.

Também é possível incluir determinados cabeçalhos gerados pelo CloudFront na chave de cache. Para ter mais informações, consulte [the section called “Adicionar cabeçalhos de solicitação do CloudFront”](#).

Cookies

Os cookies em solicitações do visualizador que o CloudFront inclui na chave de cache e nas solicitações da origem. Para cookies, é possível escolher uma das seguintes configurações:

- **None (Nenhum):** os cookies nas solicitações do visualizador não são incluídos na chave de cache e não são incluídos automaticamente nas solicitações da origem.
- **All (Todos):** todos os cookies em solicitações do visualizador são incluídos na chave de cache e incluídos automaticamente nas solicitações da origem.
- **Include specified cookies (Incluir os cookies especificados):** você especifica quais dos cookies nas solicitações do visualizador serão incluídos na chave de cache e incluídos automaticamente nas solicitações da origem.

- **Include all cookies except (Incluir todos os cookies, exceto):** você especifica quais dos cookies nas solicitações do visualizador não serão incluídos na chave de cache e não serão incluídos automaticamente nas solicitações da origem. Todos os outros cookies, exceto os que você especificar, são incluídos na chave de cache e incluídos automaticamente nas solicitações da origem.

Ao usar a configuração **Include specified cookies (Incluir os cookies especificados)** ou **Include all cookies except (Incluir todos os cookies, exceto)**, você especifica cookies pelo nome, não pelo valor. Por exemplo, considere o seguinte cabeçalho `Cookie`:

```
Cookie: session_ID=abcd1234
```

Nesse caso, você especifica o cookie como `session_ID`, não como `session_ID=abcd1234`. No entanto, o CloudFront inclui o cookie completo, incluindo o seu valor, na chave de cache e nas solicitações da origem.

Strings de consulta

As strings de consulta de URL nas solicitações do visualizador que o CloudFront inclui na chave de cache e nas solicitações de origem. Para strings de consulta, é possível escolher uma das seguintes configurações:

- **None (Nenhuma):** as strings de consulta nas solicitações do visualizador não são incluídas na chave do cache e não são incluídas automaticamente nas solicitações de origem.
- **All (Todas):** todas as strings de consulta em solicitações do visualizador são incluídas na chave de cache e também são incluídas automaticamente nas solicitações de origem.
- **Include specified query strings (Incluir as strings de consulta especificadas):** você especifica quais strings de consulta nas solicitações do visualizador serão incluídas na chave de cache e incluídas automaticamente nas solicitações de origem.
- **Include all query strings except (Incluir todas as strings de consulta, exceto):** você especifica quais das strings de consulta nas solicitações do visualizador não serão incluídas na chave de cache e não serão incluídas automaticamente nas solicitações da origem. Todas as outras strings de consulta, com exceção das que você especificar, são incluídas na chave de cache e incluídas automaticamente nas solicitações da origem.

Ao usar a configuração **Include specified query strings (Incluir as strings de consulta especificadas)** ou **Include all query strings except (Incluir todas as strings de consulta, exceto)**,

você especifica as strings de consulta pelo nome, não pelo valor. Por exemplo, considere o seguinte caminho do URL:

```
/content/stories/example-story.html?split-pages=false
```

Nesse caso, você especifica a string de consulta como `split-pages`, não como `split-pages=false`. No entanto, o CloudFront inclui a string de consulta completa, incluindo seu valor, na chave de cache e nas solicitações de origem.

Suporte à compactação

Essas configurações permitem que o CloudFront solicite e armazene em cache objetos compactados nos formatos de compactação Gzip ou Brotli, quando o visualizador for compatível com eles. Essas configurações também permitem que a [compactação do CloudFront](#) funcione. Os visualizadores indicam sua compatibilidade com esses formatos de compactação com o cabeçalho HTTP `Accept-Encoding`.


Note

Os navegadores da Web Chrome e Firefox são compatíveis com a compactação Brotli somente quando a solicitação é enviada usando HTTPS. Esses navegadores não são compatíveis com o Brotli com solicitações HTTP.

Habilite essas configurações quando qualquer uma das seguintes situações for verdadeira:

- Sua origem retorna objetos compactados Gzip quando os visualizadores são compatíveis com ele (as solicitações contêm o cabeçalho HTTP `Accept-Encoding` com `gzip` como um valor). Nesse caso, use a configuração `Gzip enabled` (Habilitada para Gzip) (defina `EnableAcceptEncodingGzip` como `true` na API do CloudFront, nos AWS SDKs, na AWS CLI ou no AWS CloudFormation).
- A origem retorna objetos compactados Brotli quando os visualizadores são compatíveis com ele (as solicitações contêm o cabeçalho HTTP `Accept-Encoding` com `br` como um valor). Nesse caso, use a configuração `Brotli enabled` (Habilitada para Brotli) (defina `EnableAcceptEncodingBrotli` como `true` na API do CloudFront, nos AWS SDKs, na AWS CLI ou no AWS CloudFormation).
- O comportamento do cache ao qual esta política de cache está anexada é configurado com [Compactação do CloudFront](#). Nesse caso, é possível habilitar o cache para Gzip ou Brotli ou

ambos. Quando a compactação do CloudFront está habilitada, habilitar o cache para os dois formatos pode ajudar a reduzir os custos de transferência de dados para a internet.

 Note

Se você habilitar o armazenamento em cache para um ou os dois formatos de compactação, não inclua o cabeçalho `Accept-Encoding` em uma [política de solicitação de origem](#) associada ao mesmo comportamento de cache. O CloudFront sempre inclui esse cabeçalho em solicitações da origem quando o cache está habilitado para qualquer um desses formatos, portanto, incluir `Accept-Encoding` em uma política de solicitação de origem não tem efeito.

Se o servidor de origem não retornar objetos compactados por Gzip ou Brotli, ou o comportamento do cache não estiver configurado com compactação do CloudFront, não habilite o cache para objetos compactados. Se você o habilitar, a [taxa de acertos do cache](#) poderá diminuir.

A explicação a seguir mostra como essas configurações afetam uma distribuição do CloudFront. Todas as situações descritas a seguir partem do pressuposto de que a solicitação do visualizador contém o cabeçalho `Accept-Encoding`. Quando a solicitação do visualizador não inclui o cabeçalho `Accept-Encoding`, o CloudFront não inclui esse cabeçalho na chave de cache e não o inclui na solicitação de origem correspondente.

Quando o armazenamento em cache de objetos compactados está habilitado para os dois formatos de compactação

Se o visualizador for compatível com Gzip e Brotli, ou seja, se os valores `gzip` e `br` estiverem no cabeçalho `Accept-Encoding` na solicitação do visualizador, o CloudFront fará o seguinte:

- Normaliza o cabeçalho como `Accept-Encoding: br, gzip` e inclui o cabeçalho normalizado na chave de cache. A chave de cache não inclui outros valores que estavam no cabeçalho `Accept-Encoding` enviado pelo visualizador.
- Se o ponto de presença tiver um objeto compactado Brotli ou Gzip no cache que corresponda à solicitação e não tiver expirado, o ponto de presença retornará o objeto ao visualizador.
- Se o local da borda não tiver um objeto compactado por Brotli ou Gzip no cache, que corresponda à solicitação e não esteja expirado, o CloudFront incluirá o cabeçalho normalizado (`Accept-Encoding: br, gzip`) na solicitação de origem correspondente.

A solicitação de origem não incluirá outros valores que estavam no cabeçalho `Accept-Encoding` enviado pelo visualizador.

Se o visualizador for compatível com um formato de compactação, mas não com o outro, por exemplo, se `gzip` for um valor no cabeçalho `Accept-Encoding` na solicitação do visualizador, mas `br` não for, o CloudFront fará o seguinte:

- Normaliza o cabeçalho como `Accept-Encoding: gzip` e inclui o cabeçalho normalizado na chave de cache. A chave de cache não inclui outros valores que estavam no cabeçalho `Accept-Encoding` enviado pelo visualizador.
- Se o ponto de presença tiver um objeto compactado Gzip no cache que corresponda à solicitação e não tiver expirado, o ponto de presença retornará o objeto ao visualizador.
- Se o local da borda não tiver um objeto compactado por Gzip no cache, que corresponda à solicitação e não esteja expirado, o CloudFront incluirá o cabeçalho normalizado (`Accept-Encoding: gzip`) na solicitação de origem correspondente. A solicitação de origem não incluirá outros valores que estavam no cabeçalho `Accept-Encoding` enviado pelo visualizador.

Para entender o que o CloudFront fará se o visualizador for compatível com Brotli, mas não com Gzip, substitua os dois formatos de compactação um pelo outro no exemplo anterior.

Se o visualizador não for compatível com o Brotli ou com o Gzip, ou seja, o cabeçalho `Accept-Encoding` na solicitação do visualizador não contiver `br` ou `gzip` como valores, o CloudFront:

- Não incluirá o cabeçalho `Accept-Encoding` na chave de cache.
- Incluirá `Accept-Encoding: identity` na solicitação de origem correspondente. A solicitação de origem não incluirá outros valores que estavam no cabeçalho `Accept-Encoding` enviado pelo visualizador.

Quando o armazenamento em cache de objetos compactados está habilitado para um formato de compactação, mas não o outro

Se o visualizador for compatível com o formato para o qual o cache está habilitado, por exemplo, se o armazenamento em cache de objetos compactados estiver habilitado para Gzip, e o visualizador for compatível com o Gzip (`gzip` for um dos valores no cabeçalho `Accept-Encoding` na solicitação do visualizador), o CloudFront fará o seguinte:

- Normaliza o cabeçalho como `Accept-Encoding: gzip` e inclui o cabeçalho normalizado na chave de cache.

- Se o ponto de presença tiver um objeto compactado Gzip no cache que corresponda à solicitação e não tiver expirado, o ponto de presença retornará o objeto ao visualizador.
- Se o local da borda não tiver um objeto compactado por Gzip no cache, que corresponda à solicitação e não esteja expirado, o CloudFront incluirá o cabeçalho normalizado (`Accept-Encoding: gzip`) na solicitação de origem correspondente. A solicitação de origem não incluirá outros valores que estavam no cabeçalho `Accept-Encoding` enviado pelo visualizador.

Esse comportamento é o mesmo quando o visualizador é compatível com o Gzip e o Brotli (o cabeçalho `Accept-Encoding` na solicitação do visualizador inclui `gzip` e `br` como valores), porque nesse cenário, o armazenamento em cache de objetos compactados para Brotli não está habilitado.

Para entender o que o CloudFront fará se o armazenamento em cache de objetos compactados estiver habilitado para Brotli, mas não para Gzip, substitua os dois formatos de compactação um pelo outro no exemplo anterior.

Se o visualizador não for compatível com o formato de compactação para o qual o cache está habilitado (o cabeçalho `Accept-Encoding` na solicitação do visualizador não contiver o valor desse formato), o CloudFront:

- Não incluirá o cabeçalho `Accept-Encoding` na chave de cache.
- Incluirá `Accept-Encoding: identity` na solicitação de origem correspondente. A solicitação de origem não incluirá outros valores que estavam no cabeçalho `Accept-Encoding` enviado pelo visualizador.

Quando o armazenamento em cache de objetos compactados está desabilitado para os dois formatos de compactação

Quando o armazenamento em cache de objetos compactados está desabilitado para os dois formatos de compactação, o CloudFront trata o cabeçalho `Accept-Encoding` da mesma forma como qualquer outro cabeçalho HTTP na solicitação do visualizador. Por padrão, ele não está incluído na chave de cache e não está incluído nas solicitações de origem. É possível incluí-lo na lista de cabeçalhos em uma política de cache ou em uma política de solicitação de origem como qualquer outro cabeçalho HTTP.

Criar políticas de cache

É possível usar uma política de cache para melhorar a taxa de acertos do cache controlando os valores (strings de consulta de URL, cabeçalhos HTTP e cookies) incluídos na chave de cache. Você pode criar uma política de cache no console do CloudFront com a AWS Command Line Interface (AWS CLI) ou a API do CloudFront.

Depois de criar uma política de cache, anexe-a a um ou mais comportamentos de cache em uma distribuição do CloudFront.

Console

Como criar uma política de cache (console)

1. Faça login no AWS Management Console e abra a página Políticas (Políticas) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home?#/policies>.
2. Escolha Create cache policy (Criar política de cache).
3. Escolha a configuração desejada para esta política de cache. Para obter mais informações, consulte [Noções básicas sobre políticas de cache](#).
4. Quando terminar, escolha Create (Criar).

Depois de criar uma política de cache, é possível anexá-la a um comportamento de cache.

Como anexar uma política de cache a uma distribuição existente (console)

1. Abra a página Distributions (Distribuições) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/distributions>.
2. Escolha a distribuição a ser atualizada e escolha a guia Behaviors (Comportamentos).
3. Escolha o comportamento de cache a ser atualizado e escolha Edit (Editar).

Ou, para criar um novo comportamento de cache, escolha Create behavior (Criar comportamento).

4. Na seção Cache key and origin requests (Solicitações da chave de cache e de origem), verifique se a opção Cache policy and origin request policy (Política de cache e política de solicitação de origem) está selecionada.
5. Em Cache policy (Política de cache), escolha a política de cache a ser anexada a esse comportamento de cache.

6. Na parte inferior da página, escolha Save changes (Salvar alterações).

Como anexar uma política de cache a uma nova distribuição (console)

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha Create distribution (Criar distribuição).
3. Na seção Cache key and origin requests (Solicitações da chave de cache e de origem), verifique se a opção Cache policy and origin request policy (Política de cache e política de solicitação de origem) está selecionada.
4. Em Cache policy (Política de cache), escolha a política de cache a ser anexada ao comportamento de cache padrão dessa distribuição.
5. Escolha as configurações desejadas para a origem, o comportamento padrão do cache e outras configurações de distribuição. Para obter mais informações, consulte [Referência de configurações da distribuição](#).
6. Ao concluir, escolha Create distribution (Criar distribuição).

CLI

Para criar uma política de cache com a AWS Command Line Interface (AWS CLI), use o comando `aws cloudfront create-cache-policy`. É possível usar um arquivo de entrada para fornecer os parâmetros de entrada do comando, em vez de especificar cada parâmetro individual como entrada na linha de comando.

Como criar uma política de cache (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `cache-policy.yaml` que contém todos os parâmetros de entrada para o comando `create-cache-policy`.

```
aws cloudfront create-cache-policy --generate-cli-skeleton yaml-input > cache-policy.yaml
```

2. Abra o arquivo chamado `cache-policy.yaml` que você acabou de criar. Edite o arquivo para especificar as configurações de política de cache desejadas e salve o arquivo. É possível remover campos opcionais do arquivo, mas não remover os campos obrigatórios.

Para obter mais informações sobre as configurações de política de cache, consulte [Noções básicas sobre políticas de cache](#).

- Use o seguinte comando para criar a política de cache usando parâmetros de entrada do arquivo `cache-policy.yaml`.

```
aws cloudfront create-cache-policy --cli-input-yaml file://cache-policy.yaml
```

Anote o valor de `Id` na saída do comando. Esse é o ID da política de cache que será necessário para anexar a política de cache ao comportamento de cache de uma distribuição do CloudFront.

Como anexar uma política de cache a uma distribuição existente (CLI com arquivo de entrada)

- Use o comando a seguir para salvar a configuração da distribuição do CloudFront que você deseja atualizar. Substitua *distribution_ID* pelo ID da distribuição.

```
aws cloudfront get-distribution-config --id distribution_ID --output yaml > dist-config.yaml
```

- Abra o arquivo chamado `dist-config.yaml` que você acabou de criar. Edite o arquivo, fazendo as seguintes alterações em cada comportamento de cache que você está atualizando para usar uma política de cache.
 - No comportamento de cache, adicione um campo chamado `CachePolicyId`. Para o valor do campo, use o ID da política de cache que você anotou depois de criar a política.
 - Remova os campos `MinTTL`, `MaxTTL`, `DefaultTTL` e `ForwardedValues` do comportamento de cache. Essas configurações são especificadas na política de cache, portanto você não pode incluir esses campos e uma política de cache no mesmo comportamento de cache.
 - Renomeie o campo `ETag` para `IfMatch`, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

- Use o comando a seguir para atualizar a distribuição para utilizar a política de cache. Substitua *distribution_ID* pelo ID da distribuição.

```
aws cloudfront update-distribution --id distribution_ID --cli-input-yaml file://  
dist-config.yaml
```

Como anexar uma política de cache a uma nova distribuição (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `distribution.yaml` que contém todos os parâmetros de entrada para o comando `create-distribution`.

```
aws cloudfront create-distribution --generate-cli-skeleton yaml-input >  
distribution.yaml
```

2. Abra o arquivo chamado `distribution.yaml` que você acabou de criar. No comportamento de cache padrão, no campo `CachePolicyId`, insira o ID da política de cache que você anotou após criar a política. Continue editando o arquivo para especificar as configurações de distribuição desejadas e salve o arquivo ao concluir.

Para obter mais informações sobre as configurações de distribuição, consulte [Referência de configurações da distribuição](#).

3. Use o seguinte comando para criar a distribuição usando parâmetros de entrada do arquivo `distribution.yaml`.

```
aws cloudfront create-distribution --cli-input-yaml file://distribution.yaml
```

API

Para criar uma política de cache com a API do CloudFront, use [CreateCachePolicy](#). Para obter mais informações sobre os campos especificados nessa chamada de API, consulte [Noções básicas sobre políticas de cache](#) e a documentação de referência de API do seu SDK da AWS ou de outro cliente de API.

Depois de criar uma política de cache, é possível anexá-la a um comportamento de cache, usando uma das seguintes chamadas de API:

- Para anexá-la a um comportamento de cache em uma distribuição existente, use [UpdateDistribution](#).
- Para anexá-la a um comportamento de cache em uma nova distribuição, use [CreateDistribution](#).

Para as duas chamadas de API, forneça o ID da política de cache no campo `CachePolicyId`, dentro de um comportamento de cache. Para mais informações sobre os outros campos especificados nessas chamadas de API, consulte [Referência de configurações da distribuição](#) e a documentação de referência da API do AWS SDK ou de outro cliente de API.

Usar políticas de cache gerenciadas

O CloudFront fornece um conjunto de políticas de cache gerenciadas que é possível anexar a qualquer um dos comportamentos de cache da distribuição. Com uma política de cache gerenciada, você não precisa gravar ou manter sua própria política de cache. As políticas gerenciadas usam configurações que são otimizadas para casos de uso específicos.

Para usar uma política de cache gerenciada, anexe-a a um comportamento de cache em sua distribuição. O processo é o mesmo que o da criação de uma política de cache, mas em vez de criar uma, basta anexar uma das políticas de cache gerenciadas. Você anexa a política por nome (com o console) ou por ID (com a AWS CLI ou os SDKs). Os nomes e IDs são listados na seção a seguir.

Para ter mais informações, consulte [Criar políticas de cache](#).

Os tópicos a seguir descrevem as políticas de cache gerenciadas que você pode usar.

Tópicos

- [Amplify](#)
- [CachingDisabled](#)
- [CachingOptimized](#)
- [CachingOptimizedForUncompressedObjects](#)
- [Elemental-MediaPackage](#)
- [UseOriginCacheControlHeaders](#)
- [UseOriginCacheControlHeaders-QueryString](#)

Amplify

[Visualizar essa política no console do CloudFront](#)

Esta política foi projetada para uso com uma origem que é uma aplicação Web do [AWS Amplify](#).

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

```
2e54312d-136d-493c-8eb9-b001f22f67d2
```

Essa política tem as seguintes configurações:

- TTL mínimo: 2 segundos
- TTL máximo: 600 segundos (10 minutos)
- TTL padrão: 2 segundos
- Cabeçalhos incluídos na chave de cache:
 - Authorization
 - CloudFront-Viewer-Country
 - Host

O cabeçalho normalizado Accept-Encoding também é incluído porque a configuração de objetos compactados de cache está habilitada. Para obter mais informações, consulte o [Suporte à compactação](#).


- Cookies included in cache key (Cookies incluídos na chave de cache): todos os cookies serão incluídos.
- Query strings included in cache key (Strings de consulta incluídas na chave de cache): todas as strings de consulta serão incluídas.
- Configuração de armazenamento de objetos compactados em cache: habilitado. Para obter mais informações, consulte o [Suporte à compactação](#).

Políticas de cache de hospedagem do AWS Amplify

O Amplify usa as seguintes políticas de cache gerenciadas para otimizar a configuração de cache padrão das aplicações dos clientes:

- [Amplify-Default](#)
- [Amplify-DefaultNoCookies](#)

- [Amplify-ImageOptimization](#)
- [Amplify-StaticContent](#)

 Note

Essas políticas são usadas apenas pelo Amplify. Não recomendamos usá-las para suas distribuições.

Para ter mais informações sobre como gerenciar a configuração de cache para a aplicação hospedada no Amplify, consulte [Managing cache configuration](#) no Guia do usuário de hospedagem do Amplify.

CachingDisabled

[Visualizar essa política no console do CloudFront](#)

Esta política desabilita o armazenamento em cache. Essa política é útil para conteúdo dinâmico e para solicitações que não podem ser armazenadas em cache.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

```
4135ea2d-6df8-44a3-9df3-4b5a84be39ad
```

Essa política tem as seguintes configurações:

- TTL mínimo: 0 segundo
- TTL máximo: 0 segundo
- TTL padrão: 0 segundo
- Cabeçalhos incluídos na chave de cache: nenhum
- Cookies incluídos na chave de cache: nenhum
- Strings de consulta incluídas na chave de cache: nenhuma
- Configuração de armazenamento de objetos compactados em cache: desabilitado

CachingOptimized

[Visualizar essa política no console do CloudFront](#)

Essa política foi projetada para otimizar a eficiência do cache minimizando os valores incluídos pelo CloudFront na chave de cache. O CloudFront não inclui strings de consulta ou cookies na chave de cache e inclui apenas o cabeçalho Accept-Encoding normalizado. Isso permite que o CloudFront armazene objetos em cache separadamente nos formatos de compactação Gzip e Brotli quando a origem os retorna ou quando a [Compactação de borda do CloudFront](#) está habilitada.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

```
658327ea-f89d-4fab-a63d-7e88639e58f6
```

Essa política tem as seguintes configurações:

- TTL mínimo: 1 segundo.
- TTL máximo: 31.536.000 segundos (365 dias).
- TTL padrão: 86.400 segundos (24 horas).
- Headers included in the cache key (Cabeçalhos incluídos na chave de cache): nenhum será explicitamente incluído. O cabeçalho normalizado Accept-Encoding é incluído porque a configuração de objetos compactados de cache está habilitada. Para obter mais informações, consulte o [Suporte à compactação](#).
- Cookies incluídos na chave de cache: nenhum.
- Strings de consulta incluídas na chave de cache: nenhuma.
- Configuração de armazenamento de objetos compactados em cache: habilitado. Para obter mais informações, consulte o [Suporte à compactação](#).

CachingOptimizedForUncompressedObjects

[Visualizar essa política no console do CloudFront](#)

Essa política foi projetada para otimizar a eficiência do cache minimizando os valores incluídos na chave de cache. Nenhuma string de consulta, cabeçalho ou cookie é incluído. Essa política é idêntica à anterior, mas desabilita a configuração de armazenamento de objetos compactados em cache.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

```
b2884449-e4de-46a7-ac36-70bc7f1ddd6d
```

Essa política tem as seguintes configurações:

- TTL mínimo: 1 segundo
- TTL máximo: 31.536.000 segundos (365 dias)
- TTL padrão: 86.400 segundos (24 horas)
- Cabeçalhos incluídos na chave de cache: nenhum
- Cookies incluídos na chave de cache: nenhum
- Strings de consulta incluídas na chave de cache: nenhuma
- Configuração de armazenamento de objetos compactados em cache: desabilitado

Elemental-MediaPackage

[Visualizar essa política no console do CloudFront](#)

Esta política foi projetada para uso com uma origem que é um endpoint do AWS Elemental MediaPackage.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

08627262-05a9-4f76-9ded-b50ca2e3a84f

Essa política tem as seguintes configurações:

- TTL mínimo: 0 segundo
- TTL máximo: 31.536.000 segundos (365 dias)
- TTL padrão: 86.400 segundos (24 horas)
- Cabeçalhos incluídos na chave de cache:
 - `Origin`

O cabeçalho normalizado `Accept-Encoding` também será incluído porque a configuração de objetos compactados de cache está habilitada para Gzip. Para obter mais informações, consulte o [Suporte à compactação](#).

- Cookies incluídos na chave de cache: nenhum
- Strings de consulta incluídas na chave de cache:
 - `aws.manifestfilter`
 - `start`

- end
- m
- Configuração de armazenamento de objetos compactados em cache: habilitado para Gzip. Para obter mais informações, consulte o [Suporte à compactação](#).

UseOriginCacheControlHeaders

[Visualizar essa política no console do CloudFront](#)

Essa política foi projetada para uso com uma origem que exibe cabeçalhos de resposta HTTP `Cache-Control` e não veicula conteúdo diferente com base nos valores presentes na string de consulta. Se a origem fornecer conteúdo diferente com base em valores presentes na string de consulta, pense em usar [UseOriginCacheControlHeaders-QueryString](#).

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

```
83da9c7e-98b4-4e11-a168-04f0df8e2c65
```

Essa política tem as seguintes configurações:

- TTL mínimo: 0 segundo
- TTL máximo: 31.536.000 segundos (365 dias)
- TTL padrão: 0 segundo
- Cabeçalhos incluídos na chave de cache:
 - Host
 - Origin
 - X-HTTP-Method-Override
 - X-HTTP-Method
 - X-Method-Override

O cabeçalho normalizado `Accept-Encoding` também é incluído porque a configuração de objetos compactados de cache está habilitada. Para obter mais informações, consulte o [Suporte à compactação](#).

- Cookies incluídos na chave de cache: todos os cookies serão incluídos.
- Strings de consulta incluídas na chave de cache: nenhuma.

- Configuração de armazenamento de objetos compactados em cache: habilitado. Para obter mais informações, consulte o [Suporte à compactação](#).

UseOriginCacheControlHeaders-QueryString

[Visualizar essa política no console do CloudFront](#)

Essa política foi projetada para uso com uma origem que exibe cabeçalhos de resposta HTTP `Cache-Control` e veicula conteúdo diferente com base em valores presentes na string de consulta. Se a origem não fornecer conteúdo diferente com base em valores presentes na string de consulta, pense em usar [UseOriginCacheControlHeaders](#).

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

```
4cc15a8a-d715-48a4-82b8-cc0b614638fe
```

Essa política tem as seguintes configurações:

- TTL mínimo: 0 segundo
- TTL máximo: 31.536.000 segundos (365 dias)
- TTL padrão: 0 segundo
- Cabeçalhos incluídos na chave de cache:
 - Host
 - Origin
 - X-HTTP-Method-Override
 - X-HTTP-Method
 - X-Method-Override

O cabeçalho normalizado `Accept-Encoding` também é incluído porque a configuração de objetos compactados de cache está habilitada. Para obter mais informações, consulte o [Suporte à compactação](#).

- Cookies incluídos na chave de cache: todos os cookies serão incluídos.
- Strings de consulta incluídas na chave de cache: todas as strings de consulta serão incluídas.
- Configuração de armazenamento de objetos compactados em cache: habilitado. Para obter mais informações, consulte o [Suporte à compactação](#).

Noções básicas sobre a chave de cache

A chave de cache determina se a solicitação de um visualizador para um ponto de presença do CloudFront resulta em um acerto de cache. A chave de cache é o identificador exclusivo de um objeto no cache. Cada objeto no cache tem uma chave de cache exclusiva.

Um acerto de cache ocorre quando uma solicitação de visualizador gera a mesma chave de cache de uma solicitação anterior, e o objeto dessa chave de cache está no cache do local da borda e é válido. Quando há um acerto de cache, o objeto solicitado é fornecido ao visualizador de um ponto de presença do CloudFront, o que traz os seguintes benefícios:

- Carga reduzida no servidor de origem
- Latência reduzida para o visualizador

É possível obter melhor performance do site ou da aplicação quando você tem uma taxa de acertos de cache maior (uma proporção maior de solicitações do visualizador que resultam em um acerto de cache). Uma maneira de melhorar a taxa de acertos do cache é incluir apenas os valores mínimos necessários na chave de cache. Para obter mais informações, consulte as seções a seguir.

É possível modificar os valores (strings de consulta de URL, cabeçalhos HTTP e cookies) na chave de cache usando uma [política de cache](#). (Também é possível modificar a chave de cache usando uma [função do Lambda@Edge](#).) Antes de modificar a chave de cache, é importante entender como a aplicação foi projetada e quando e como ela pode fornecer respostas diferentes com base nas características da solicitação do visualizador. Quando um valor na solicitação do visualizador determinar a resposta retornada pela origem, inclua esse valor na chave de cache. Mas se você incluir um valor na chave de cache que não afete a resposta retornada pela origem, poderá acabar armazenando objetos duplicados em cache.

Chave de cache padrão

Por padrão, a chave de cache de uma distribuição do CloudFront inclui as seguintes informações:

- O nome de domínio da distribuição do CloudFront (por exemplo, d111111abcdef8.cloudfront.net)
- O caminho do URL do objeto solicitado (por exemplo, /content/stories/example-story.html)

Note

O método OPTIONS está incluído na chave de cache para solicitações de OPTIONS. Isto significa que respostas para as solicitações de OPTIONS são armazenadas em cache separadamente das respostas para as solicitações de GET e HEAD.

Outros valores da solicitação do visualizador não são incluídos na chave de cache, por padrão. Considere a seguinte solicitação HTTP de um navegador da Web.

```
GET /content/stories/example-story.html?ref=0123abc&split-pages=false
HTTP/1.1
Host: d111111abcdef8.cloudfront.net
User-Agent: Mozilla/5.0 Gecko/20100101 Firefox/68.0
Accept: text/html,*/*
Accept-Language: en-US,en
Cookie: session_id=01234abcd
Referer: https://news.example.com/
```

Quando uma solicitação de visualizador como essa do exemplo chega a um local da borda do CloudFront, o CloudFront usa a chave de cache para determinar se há um acerto de cache. Por padrão, somente os seguintes componentes da solicitação são incluídos na chave de cache: /content/stories/example-story.html e d111111abcdef8.cloudfront.net. Se o objeto solicitado não estiver no cache (uma falha de cache), o CloudFront enviará uma solicitação à origem para obter o objeto. Depois de obter o objeto, o CloudFront o retorna ao visualizador e o armazena no cache do ponto de presença.

Quando o CloudFront recebe outra solicitação para o mesmo objeto, conforme determinado pela chave de cache, o CloudFront fornece o objeto armazenado em cache para o visualizador imediatamente, sem enviar uma solicitação à origem. Por exemplo, considere a seguinte solicitação HTTP que é recebida após a solicitação anterior.

```
GET /content/stories/example-story.html?ref=xyz987&split-pages=true
HTTP/1.1
Host: d111111abcdef8.cloudfront.net
User-Agent: Mozilla/5.0 AppleWebKit/537.36 Chrome/83.0.4103.116
Accept: text/html,*/*
```

```
Accept-Language: en-US,en  
Cookie: session_id=wxyz9876  
Referer: https://rss.news.example.net/
```

Essa solicitação é para o mesmo objeto que a solicitação anterior, embora seja diferente da solicitação anterior. Ela tem uma string de consulta de URL diferente, cabeçalhos `User-Agent` e `Referer` diferentes, e outro cookie de `session_id`. No entanto, nenhum desses valores faz parte da chave de cache por padrão, portanto, essa segunda solicitação resulta em um acerto de cache.

Personalizar a chave de cache

Em alguns casos, convém incluir mais informações na chave de cache, mesmo que isso possa resultar em menos acertos de cache. Você especifica o que incluir na chave de cache usando uma [política de cache](#).

Por exemplo, se o servidor de origem usar o cabeçalho HTTP `Accept-Language` em solicitações do visualizador para retornar conteúdo diferente com base no idioma do visualizador, convém incluir esse cabeçalho na chave de cache. Quando você faz isso, o CloudFront usa esse cabeçalho para determinar os acertos de cache e inclui o cabeçalho nas solicitações de origem (solicitações que o CloudFront envia à origem quando há uma falha de cache).

Uma consequência potencial da inclusão de valores adicionais na chave de cache é que o CloudFront pode acabar armazenando objetos duplicados em cache devido à variação que pode ocorrer em solicitações do visualizador. Por exemplo, os visualizadores podem enviar um dos seguintes valores para o cabeçalho `Accept-Language`:

- en-US, en
- en, en-US
- en-US, en
- en-US

Todos esses valores diferentes indicam que o idioma do visualizador é o inglês, mas a variação pode fazer com que o CloudFront armazene o mesmo objeto em cache várias vezes. Isso pode reduzir os acertos de cache e aumentar o número de solicitações de origem. Para evitar essa duplicação, não coloque o cabeçalho `Accept-Language` na chave de cache. Em vez disso, configure o site ou a aplicação de modo que eles usem URLs diferentes para conteúdo em diferentes idiomas (por exemplo, `/en-US/content/stories/example-story.html`).

Para qualquer valor específico que você pretenda incluir na chave de cache, entenda quantas variações diferentes desse valor podem aparecer nas solicitações do visualizador. Para determinados valores de solicitação, raramente faz sentido incluí-los na chave de cache. Por exemplo, o cabeçalho `User-Agent` pode ter milhares de variações exclusivas, portanto, geralmente ele não é um bom candidato para inclusão na chave de cache. Os cookies que têm valores específicos do usuário ou específicos da sessão e são exclusivos em milhares (ou mesmo milhões) de solicitações também não são bons candidatos para inclusão na chave de cache. Se você incluir esses valores na chave de cache, cada variação exclusiva resultará em outra cópia do objeto no cache. Se essas cópias do objeto não forem exclusivas, ou se você acabar com um número tão grande de objetos ligeiramente diferentes que cada objeto obtém apenas um pequeno número de acertos de cache, convém considerar uma abordagem diferente. É possível excluir esses valores altamente variáveis da chave de cache ou marcar objetos como não armazenáveis em cache.

Tenha cuidado ao personalizar a chave de cache. Às vezes, isso é desejável, mas pode ter consequências não intencionais, como armazenar em cache objetos duplicados, reduzir a taxa de acertos do cache e aumentar o número de solicitações de origem. Se o site ou a aplicação de origem precisar receber determinados valores de solicitações do visualizador para análise, telemetria ou outros usos, mas esses valores não alterarem o objeto retornado pela origem, use uma [política de solicitação de origem](#) para incluir esses valores em solicitações de origem, mas não os inclua na chave de cache.

Controlar as solicitações de origem com uma política

Quando uma solicitação do visualizador para o CloudFront resulta em uma falha de cache (o objeto solicitado não está armazenado em cache no ponto de presença), o CloudFront envia uma solicitação à origem para recuperar o objeto. É o que chamamos de solicitação de origem. A solicitação de origem sempre inclui as seguintes informações da solicitação do visualizador:

- O caminho da URL (somente o caminho, sem strings de consulta de URL ou o nome de domínio)
- O corpo da solicitação (se houver)
- Os cabeçalhos HTTP que o CloudFront inclui automaticamente em cada solicitação de origem, incluindo `Host`, `User-Agent` e `X-Amz-Cf-Id`

Outras informações da solicitação do visualizador, como strings de consulta de URL, cabeçalhos HTTP e cookies, não são incluídas na solicitação de origem por padrão. (Exceção: com configurações de cache herdadas, o CloudFront encaminha os cabeçalhos para a origem por padrão.) No entanto, é recomendável receber algumas dessas outras informações na origem, por exemplo, para coletar dados para análise ou telemetria. É possível usar uma política de solicitação de origem para controlar as informações incluídas em uma solicitação de origem.

As políticas de solicitação de origem são separadas das [políticas de cache](#) que controlam a chave de cache. Dessa forma, é possível receber informações adicionais na origem e também manter uma boa taxa acertos do cache (a proporção de solicitações do visualizador que ocasionam um acerto de cache). Você faz isso controlando separadamente quais informações são incluídas nas solicitações de origem (usando a política de solicitação de origem) e são incluídas na chave de cache (usando a política de cache).

Embora os dois tipos de política sejam separados, eles estão relacionados. Todas as strings de consulta de URL, cabeçalhos HTTP e cookies que você inclui na chave de cache (usando uma política de cache) são automaticamente incluídos nas solicitações de origem. Use a política de solicitação de origem para especificar as informações que você deseja incluir nas solicitações de origem, mas não incluir na chave de cache. Assim como uma política de cache, você anexa uma política de solicitação de origem a um ou mais comportamentos de cache em uma distribuição do CloudFront.

Você também pode usar uma política de solicitação de origem para adicionar cabeçalhos HTTP adicionais a uma solicitação de origem que não foram incluídos na solicitação do visualizador. Esses cabeçalhos adicionais são adicionados pelo CloudFront antes de enviar a solicitação de origem,

com valores de cabeçalho determinados automaticamente com base na solicitação do visualizador. Para ter mais informações, consulte [the section called “Adicionar cabeçalhos de solicitação do CloudFront”](#).

Tópicos

- [Noções básicas sobre políticas de solicitação de origem](#)
- [Criar políticas de solicitação de origem](#)
- [Usar políticas de solicitação de origem gerenciadas](#)
- [Adicionar cabeçalhos de solicitação do CloudFront](#)
- [Noções básicas sobre como as políticas de solicitação de origem e as políticas de cache funcionam juntas](#)

Noções básicas sobre políticas de solicitação de origem

O CloudFront fornece algumas políticas de solicitação de origem predefinidas, conhecidas como políticas gerenciadas, para casos de uso comuns. É possível usar essas políticas gerenciadas ou criar sua própria política de solicitação de origem específica para suas necessidades. Para obter mais informações sobre políticas gerenciadas, consulte [Usar políticas de solicitação de origem gerenciadas](#).

Uma política de solicitação de origem contém as seguintes configurações, que são categorizadas em informações de política e configurações de solicitação de origem.

Informações de política

Nome

Um nome exclusivo para identificar a política da solicitação de origem. No console, você usa o nome para anexar a política de solicitação de origem a um comportamento de cache.

Descrição

Um comentário para descrever a política da solicitação de origem. Isso é opcional.

Configurações da solicitação de origem

As configurações de solicitação de origem especificam os valores nas solicitações do visualizador que estão incluídos nas solicitações que o CloudFront envia para a origem (conhecidas como

solicitações de origem). Os valores podem incluir strings de consulta de URL, cabeçalhos HTTP e cookies. Os valores especificados são incluídos nas solicitações de origem, mas não estão incluídos na chave de cache. Para obter informações sobre como controlar a chave de cache, consulte [Controlar a chave de cache com uma política](#).

Cabeçalhos

Os cabeçalhos HTTP em solicitações do visualizador que o CloudFront inclui em solicitações de origem. Para cabeçalhos, é possível escolher uma das seguintes configurações:

- None (Nenhum): os cabeçalhos HTTP nas solicitações do visualizador não são incluídos nas solicitações de origem.
- All viewer headers (Todos os cabeçalhos do visualizador): todos os cabeçalhos HTTP nas solicitações do visualizador são incluídos nas solicitações de origem.
- All viewer headers and the following CloudFront headers (Todos os cabeçalhos do visualizador e os seguintes cabeçalhos do CloudFront): todos os cabeçalhos HTTP nas solicitações do visualizador serão incluídos nas solicitações de origem. Além disso, você especifica quais dos cabeçalhos do CloudFront você deseja adicionar às solicitações de origem. Para obter mais informações sobre os cabeçalhos do CloudFront, consulte [the section called “Adicionar cabeçalhos de solicitação do CloudFront”](#).
- Include the following headers (Incluir os seguintes cabeçalhos): você especifica quais cabeçalhos HTTP serão incluídos nas solicitações de origem.

Note

Não especifique um cabeçalho que já esteja incluído nas configurações de cabeçalhos personalizados de origem. Para ter mais informações, consulte [Configurar o CloudFront para adicionar cabeçalhos personalizados às solicitações de origem](#).

- Todos os cabeçalhos do visualizador, exceto: especifique quais cabeçalhos HTTP não estão incluídos nas solicitações de origem. Todos os outros cabeçalhos HTTP nas solicitações do visualizador, exceto os especificados, estão incluídos.

Ao usar a configuração Todos os cabeçalhos do visualizador e os seguintes cabeçalhos do CloudFront, Incluir todos os seguintes cabeçalhos ou Todos os cabeçalhos do visualizador, exceto especifique os cabeçalhos HTTP pelo nome, não pelo valor. O CloudFront inclui o cabeçalho completo, incluindo seu valor, nas solicitações de origem.

Note

Quando você usa a configuração Todos os cabeçalhos do visualizador, exceto para remover o cabeçalho do Host do visualizador, o CloudFront adiciona um novo cabeçalho do Host com o nome de domínio da origem à solicitação de origem.

Cookies

Os cookies em solicitações do visualizador que o CloudFront inclui em solicitações de origem. Para cookies, é possível escolher uma das seguintes configurações:

- **None (Nenhum):** os cookies nas solicitações do visualizador não são incluídos nas solicitações de origem.
- **All (Todos):** todos os cookies em solicitações do visualizador são incluídos em solicitações de origem.
- **Incluir cookies especificados:** especifique quais dos cookies nas solicitações do visualizador serão incluídos nas solicitações de origem.
- **Todos os cookies, exceto:** especifique quais dos cookies nas solicitações do visualizador não serão incluídos nas solicitações de origem. Todos os outros cookies nas solicitações do visualizador estão incluídos.

Ao usar a configuração Incluir os cookies especificados ou Incluir todos os cookies, exceto, especifique os cookies pelo nome, não pelo valor. O CloudFront inclui o cookie completo, incluindo seu valor, nas solicitações de origem.

Strings de consulta

As strings de consulta de URL em solicitações do visualizador que o CloudFront inclui em solicitações de origem. Para strings de consulta, é possível escolher uma das seguintes configurações:

- **None (Nenhuma):** as strings de consulta nas solicitações do visualizador não são incluídas nas solicitações de origem.
- **All (Todas):** todas as strings de consulta em solicitações do visualizador são incluídas em solicitações de origem.
- **Incluir as strings de consulta especificadas:** especifique quais das strings de consulta nas solicitações do visualizador serão incluídas nas solicitações de origem.

- Todas as consultas, exceto: especifique quais das strings de consulta nas solicitações do visualizador não serão incluídas nas solicitações de origem. Todas as outras strings de consulta estão incluídas.

Ao usar a configuração Incluir as strings de consulta especificadas ou (Incluir todas as strings de consulta, exceto, especifique as strings de consulta pelo nome, não pelo valor. O CloudFront inclui a string de consulta completa, incluindo seu valor, nas solicitações de origem.

Criar políticas de solicitação de origem

É possível usar uma política de solicitação de origem para controlar os valores (strings de consulta de URL, cabeçalhos HTTP e cookies) incluídos em solicitações que o CloudFront envia para a origem. Você pode criar uma política de solicitação de origem no console do CloudFront com a AWS Command Line Interface (AWS CLI) ou a API do CloudFront.

Depois de criar uma política de solicitação de origem, anexe-a a um ou mais comportamentos de cache em uma distribuição do CloudFront.

As políticas de solicitação de origem não são necessárias. Quando um comportamento de cache não tem uma política de solicitação de origem anexada, a solicitação de origem inclui todos os valores especificados na [política de cache](#), mas nada a mais.

Note

Para usar uma política de solicitação de origem, o comportamento de cache também deve usar uma [política de cache](#). Não é possível usar uma política de solicitação de origem em um comportamento de cache sem uma política de cache.

Console

Como criar uma política de solicitação de origem (console)

1. Faça login no AWS Management Console e abra a página Políticas (Políticas) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home?#/policies>.
2. Escolha Origin request (Solicitação de origem) e escolha Create origin request policy (Criar política de solicitação de origem).

3. Escolha a configuração desejada para esta política de solicitação de origem. Para obter mais informações, consulte [Noções básicas sobre políticas de solicitação de origem](#).
4. Quando terminar, escolha Create (Criar).

Depois de criar uma política de solicitação de origem, é possível anexá-la a um comportamento de cache.

Como anexar uma política de solicitação de origem a uma distribuição existente (console)

1. Abra a página Distributions (Distribuições) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/distributions>.
2. Escolha a distribuição a ser atualizada e escolha a guia Behaviors (Comportamentos).
3. Escolha o comportamento de cache a ser atualizado e escolha Edit (Editar).

Ou, para criar um novo comportamento de cache, escolha Create behavior (Criar comportamento).

4. Na seção Cache key and origin requests (Solicitações da chave de cache e de origem), verifique se a opção Cache policy and origin request policy (Política de cache e política de solicitação de origem) está selecionada.
5. Para Origin request policy (Política de solicitação de origem), escolha a política de solicitação de origem a ser anexada a esse comportamento de cache.
6. Na parte inferior da página, escolha Save changes (Salvar alterações).

Como anexar uma política de solicitação de origem a uma nova distribuição (console)

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha Create distribution (Criar distribuição).
3. Na seção Cache key and origin requests (Solicitações da chave de cache e de origem), verifique se a opção Cache policy and origin request policy (Política de cache e política de solicitação de origem) está selecionada.
4. Em Origin request policy (Política de solicitação de origem), escolha a política de solicitação de origem a ser anexada ao comportamento de cache padrão dessa distribuição.
5. Escolha as configurações desejadas para a origem, o comportamento padrão do cache e outras configurações de distribuição. Para obter mais informações, consulte [Referência de configurações da distribuição](#).

6. Ao concluir, escolha Create distribution (Criar distribuição).

CLI

Para criar uma política de solicitação de origem com a AWS Command Line Interface (AWS CLI), use o comando `aws cloudfront create-origin-request-policy`. É possível usar um arquivo de entrada para fornecer os parâmetros de entrada do comando, em vez de especificar cada parâmetro individual como entrada na linha de comando.

Como criar uma política de solicitação de origem (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `origin-request-policy.yaml` que contém todos os parâmetros de entrada para o comando `create-origin-request-policy`.

```
aws cloudfront create-origin-request-policy --generate-cli-skeleton yaml-input >
origin-request-policy.yaml
```

2. Abra o arquivo chamado `origin-request-policy.yaml` que você acabou de criar. Edite o arquivo para especificar as configurações de política de solicitação de origem desejadas e salve o arquivo. É possível remover campos opcionais do arquivo, mas não remover os campos obrigatórios.

Para obter mais informações sobre as configurações de política de solicitação de origem, consulte [Noções básicas sobre políticas de solicitação de origem](#).

3. Use o comando a seguir para criar a política de solicitação de origem usando parâmetros de entrada do arquivo `origin-request-policy.yaml`.

```
aws cloudfront create-origin-request-policy --cli-input-yaml file://origin-
request-policy.yaml
```

Anote o valor de `Id` na saída do comando. Esse é o ID da política de solicitação de origem, que será necessário para anexar essa política ao comportamento de cache de uma distribuição do CloudFront.

Como anexar uma política de solicitação de origem a uma distribuição existente (CLI com arquivo de entrada)

1. Use o comando a seguir para salvar a configuração da distribuição do CloudFront que você deseja atualizar. Substitua *distribution_ID* pelo ID da distribuição.

```
aws cloudfront get-distribution-config --id distribution_ID --output yaml >
dist-config.yaml
```

2. Abra o arquivo chamado `dist-config.yaml` que você acabou de criar. Edite o arquivo, fazendo as seguintes alterações em cada comportamento de cache que você está atualizando para usar uma política de solicitação de origem.
 - No comportamento de cache, adicione um campo chamado `OriginRequestPolicyId`. Para o valor do campo, use o ID da política de solicitação de origem que você anotou após criar a política.
 - Renomeie o campo `ETag` para `IfMatch`, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

3. Use o comando a seguir para atualizar a distribuição para usar a política de solicitação de origem. Substitua *distribution_ID* pelo ID da distribuição.

```
aws cloudfront update-distribution --id distribution_ID --cli-input-yaml file://
dist-config.yaml
```

Para anexar uma política de solicitação de origem a uma nova distribuição (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `distribution.yaml` que contém todos os parâmetros de entrada para o comando `create-distribution`.

```
aws cloudfront create-distribution --generate-cli-skeleton yaml-input >
distribution.yaml
```

2. Abra o arquivo chamado `distribution.yaml` que você acabou de criar. No comportamento de cache padrão, no campo `OriginRequestPolicyId`, insira o ID da política de solicitação de origem que você anotou após criar a política. Continue editando o arquivo para especificar as configurações de distribuição desejadas e salve o arquivo ao concluir.

Para obter mais informações sobre as configurações de distribuição, consulte [Referência de configurações da distribuição](#).

3. Use o seguinte comando para criar a distribuição usando parâmetros de entrada do arquivo `distribution.yaml`.

```
aws cloudfront create-distribution --cli-input-yaml file://distribution.yaml
```

API

Para criar uma política de solicitação de origem com a API do CloudFront, use [CreateOriginRequestPolicy](#). Para obter mais informações sobre os campos especificados nessa chamada de API, consulte [Noções básicas sobre políticas de solicitação de origem](#) e a documentação de referência de API do seu SDK da AWS ou de outro cliente de API.

Depois de criar uma política de solicitação de origem, é possível anexá-la a um comportamento de cache usando uma das seguintes chamadas de API:

- Para anexá-la a um comportamento de cache em uma distribuição existente, use [UpdateDistribution](#).
- Para anexá-la a um comportamento de cache em uma nova distribuição, use [CreateDistribution](#).

Para as duas chamadas de API, forneça o ID da política de solicitação de origem no campo `OriginRequestPolicyId`, dentro de um comportamento de cache. Para mais informações sobre os outros campos especificados nessas chamadas de API, consulte [Referência de configurações da distribuição](#) e a documentação de referência da API do AWS SDK ou de outro cliente de API.

Usar políticas de solicitação de origem gerenciadas

O CloudFront fornece um conjunto de políticas de solicitação de origem gerenciadas que é possível anexar a qualquer um dos comportamentos de cache da distribuição. Com uma política de solicitação de origem gerenciada, você não precisa escrever ou manter sua própria política de solicitação de origem. As políticas gerenciadas usam configurações que são otimizadas para casos de uso específicos.

Para usar uma política de solicitação de origem gerenciada, anexe-a a um comportamento de cache em sua distribuição. O processo é o mesmo que quando você cria uma política de solicitação de origem, mas em vez de criar uma, basta anexar uma das políticas de solicitação de origem gerenciadas. Você anexa a política por nome (com o console) ou por ID (com a AWS CLI ou os SDKs). Os nomes e IDs são listados na seção a seguir.

Para ter mais informações, consulte [Criar políticas de solicitação de origem](#).

Os tópicos a seguir descrevem as políticas de solicitação de origem gerenciadas que você pode usar.

Tópicos

- [AllViewer](#)
- [AllViewerAndCloudFrontHeaders-2022-06](#)
- [AllViewerExceptHostHeader](#)
- [CORS-CustomOrigin](#)
- [CORS-S3Origin](#)
- [Elemental-MediaTailor-PersonalizedManifests](#)
- [UserAgentRefererHeaders](#)

AllViewer

[Visualizar essa política no console do CloudFront](#)

Essa política inclui todos os valores (cabeçalhos, cookies e strings de consulta) da solicitação do visualizador.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

216adef6-5c7f-47e4-b989-5492eafa07d3

Essa política tem as seguintes configurações:

- Cabeçalhos incluídos nas solicitações de origem: todos os cabeçalhos na solicitação do visualizador
- Cookies incluídos nas solicitações de origem: todos
- Strings de consulta incluídas em solicitações de origem: todas

AllViewerAndCloudFrontHeaders-2022-06

[Visualizar essa política no console do CloudFront](#)

Essa política inclui todos os valores (cabeçalhos, cookies e strings de consulta) da solicitação do visualizador e todos os [cabeçalhos do CloudFront](#) que foram lançados até junho de 2022 (os cabeçalhos do CloudFront lançados depois de junho de 2022 não estão incluídos).

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

33f36d7e-f396-46d9-90e0-52428a34d9dc

Essa política tem as seguintes configurações:

- Cabeçalhos incluídos nas solicitações de origem: todos os cabeçalhos na solicitação do visualizador, e os seguintes cabeçalhos do CloudFront:
 - CloudFront-Forwarded-Proto
 - CloudFront-Is-Android-Viewer
 - CloudFront-Is-Desktop-Viewer
 - CloudFront-Is-IOS-Viewer
 - CloudFront-Is-Mobile-Viewer
 - CloudFront-Is-SmartTV-Viewer
 - CloudFront-Is-Tablet-Viewer
 - CloudFront-Viewer-Address
 - CloudFront-Viewer-ASN
 - CloudFront-Viewer-City
 - CloudFront-Viewer-Country

- CloudFront-Viewer-Country-Name
- CloudFront-Viewer-Country-Region
- CloudFront-Viewer-Country-Region-Name
- CloudFront-Viewer-Http-Version
- CloudFront-Viewer-Latitude
- CloudFront-Viewer-Longitude
- CloudFront-Viewer-Metro-Code
- CloudFront-Viewer-Postal-Code
- CloudFront-Viewer-Time-Zone
- CloudFront-Viewer-TLS
- Cookies incluídos nas solicitações de origem: todos
- Strings de consulta incluídas em solicitações de origem: todas

AllViewerExceptHostHeader

[Visualizar essa política no console do CloudFront](#)

Essa política não inclui o cabeçalho de Host da solicitação do visualizador, mas inclui todos os outros valores (cabeçalhos, cookies e strings de consulta) da solicitação do visualizador.

Essa política também inclui cabeçalhos de [solicitação adicionais do CloudFront](#) para protocolo HTTP, versão HTTP, versão TLS e todos os cabeçalhos de tipo de dispositivo e localização do visualizador.

Essa política é destinada para uso com o Amazon API Gateway e as origens do URL da função do AWS Lambda. Essas origens esperam que o cabeçalho de Host contenha o nome do domínio de origem, não o nome de domínio da distribuição do CloudFront. Encaminhar o cabeçalho de Host da solicitação do visualizador para essas origens pode impedir que elas funcionem.

Note

Quando você usa essa política de solicitação de origem gerenciada para remover o cabeçalho de Host do visualizador, o CloudFront adiciona um novo cabeçalho de Host com o nome de domínio da origem à solicitação de origem.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

b689b0a8-53d0-40ab-baf2-68738e2966ac

Essa política tem as seguintes configurações:

- Cabeçalhos incluídos nas solicitações de origem: todos os cabeçalhos na solicitação do visualizador, exceto para o cabeçalho de Host
- Cookies incluídos nas solicitações de origem: todos
- Strings de consulta incluídas em solicitações de origem: todas

CORS-CustomOrigin

[Visualizar essa política no console do CloudFront](#)

Esta política inclui o cabeçalho que permite solicitações de compartilhamento de recursos de origem cruzada (CORS) quando a origem é uma origem personalizada.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

59781a5b-3903-41f3-afcb-af62929ccde1

Essa política tem as seguintes configurações:

- Cabeçalhos incluídos nas solicitações de origem:
 - Origin
- Cookies incluídos nas solicitações de origem: nenhum
- Strings de consulta incluídas em solicitações de origem: nenhuma

CORS-S3Origin

[Visualizar essa política no console do CloudFront](#)

Esta política inclui os cabeçalhos que permitem solicitações de compartilhamento de recursos de origem cruzada (CORS) quando a origem é um bucket do Amazon S3.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

88a5eaf4-2fd4-4709-b370-b4c650ea3fcf

Essa política tem as seguintes configurações:

- Cabeçalhos incluídos nas solicitações de origem:
 - `Origin`
 - `Access-Control-Request-Headers`
 - `Access-Control-Request-Method`
- Cookies incluídos nas solicitações de origem: nenhum
- Strings de consulta incluídas em solicitações de origem: nenhuma

Elemental-MediaTailor-PersonalizedManifests

[Visualizar essa política no console do CloudFront](#)

Esta política deve ser usada com uma origem que seja um endpoint do AWS Elemental MediaTailor.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

775133bc-15f2-49f9-abea-afb2e0bf67d2

Essa política tem as seguintes configurações:

- Cabeçalhos incluídos nas solicitações de origem:
 - `Origin`
 - `Access-Control-Request-Headers`
 - `Access-Control-Request-Method`
 - `User-Agent`
 - `X-Forwarded-For`
- Cookies incluídos nas solicitações de origem: nenhum
- Strings de consulta incluídas em solicitações de origem: todas

UserAgentRefererHeaders

[Visualizar essa política no console do CloudFront](#)

Esta política inclui apenas os cabeçalhos `User-Agent` e `Referer`. Não inclui nenhuma string de consulta ou cookie.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

```
acba4595-bd28-49b8-b9fe-13317c0390fa
```

Essa política tem as seguintes configurações:

- Cabeçalhos incluídos nas solicitações de origem:
 - `User-Agent`
 - `Referer`
- Cookies incluídos nas solicitações de origem: nenhum
- Strings de consulta incluídas em solicitações de origem: nenhuma

Adicionar cabeçalhos de solicitação do CloudFront

É possível configurar o CloudFront para adicionar cabeçalhos HTTP específicos às solicitações que o CloudFront recebe dos visualizadores e encaminha para sua origem ou [função de borda](#). Os valores desses cabeçalhos HTTP são baseados nas características do visualizador ou da solicitação do visualizador. Os cabeçalhos fornecem informações sobre o tipo de dispositivo, o endereço IP, a localização geográfica, o protocolo de solicitação (HTTP ou HTTPS), a versão HTTP, os detalhes da conexão TLS, a [impressão digital JA3](#) e a impressão JA4 do visualizador.

Com esses cabeçalhos, sua origem ou função de borda pode receber informações sobre o visualizador sem a necessidade de escrever seu próprio código para determiná-las. Se a origem retornar respostas diferentes com base nas informações nesses cabeçalhos, você poderá incluí-las na chave de cache para que o CloudFront armazene as respostas em cache separadamente. Por exemplo, sua origem pode responder com conteúdo em um idioma específico com base no país em que o visualizador está ou com conteúdo personalizado para um tipo de dispositivo específico. Sua origem também pode gravar esses cabeçalhos em arquivos de log, que você pode usar para determinar informações sobre onde seus visualizadores estão, em quais tipos de dispositivo eles estão e muito mais.

Para incluir cabeçalhos na chave de cache, use uma política de cache. Para obter mais informações, consulte [Controlar a chave de cache com uma política](#) e [the section called “Noções básicas sobre a chave de cache”](#).

Para receber esses cabeçalhos em sua origem, mas não incluí-los na chave de cache, use uma política de solicitação de origem. Para ter mais informações, consulte [Controlar as solicitações de origem com uma política](#).

Tópicos

- [Cabeçalhos para determinar o tipo de dispositivo do visualizador](#)
- [Cabeçalhos para determinar a localização do visualizador](#)
- [Cabeçalhos para determinar a estrutura de cabeçalho do visualizador](#)
- [Outros cabeçalhos do CloudFront](#)

Cabeçalhos para determinar o tipo de dispositivo do visualizador


Você pode adicionar os cabeçalhos a seguir para determinar o tipo de dispositivo do visualizador. Com base no valor do cabeçalho `User-Agent`, o CloudFront define o valor desses cabeçalhos como `true` ou `false`. Se o dispositivo se encaixar em mais de uma categoria, mais de um valor poderá ser `true`. Por exemplo, para alguns tablets, o CloudFront pode definir tanto `CloudFront-Is-Mobile-Viewer` quanto `CloudFront-Is-Tablet-Viewer` como `true`.

- `CloudFront-Is-Android-Viewer`: definido como `true` quando o CloudFront determina que o visualizador é um dispositivo com o sistema operacional Android.
- `CloudFront-Is-Desktop-Viewer`: definido como `true` quando o CloudFront determina que o visualizador é um dispositivo desktop.
- `CloudFront-Is-IOS-Viewer`: definido como `true` quando o CloudFront determina que o visualizador é um dispositivo com um sistema operacional móvel Apple, como iPhone, iPod touch e alguns dispositivos iPad.
- `CloudFront-Is-Mobile-Viewer`: definido como `true` quando o CloudFront determina que o visualizador é um dispositivo móvel.
- `CloudFront-Is-SmartTV-Viewer`: definido como `true` quando o CloudFront determina que o visualizador é uma Smart TV.
- `CloudFront-Is-Tablet-Viewer`: definido como `true` quando o CloudFront determina que o visualizador é um tablet.

Cabeçalhos para determinar a localização do visualizador

Você pode adicionar os cabeçalhos a seguir para determinar a localização do visualizador. O CloudFront determina os valores para esses cabeçalhos com base no endereço IP do visualizador. Para caracteres não ASCII nos valores desses cabeçalhos, o CloudFront codifica o caractere de porcentagem de acordo com a [seção 1.2 da RFC 3986](#).

- `CloudFront-Viewer-Address`: contém o endereço IP do visualizador e a porta de origem da solicitação. Por exemplo, um valor de cabeçalho de `198.51.100.10:46532` significa que o endereço IP do visualizador é `198.51.100.10` e a porta da fonte da solicitação é `46532`.
- `CloudFront-Viewer-ASN`: contém o número de sistema autônomo (ASN) do visualizador.


 Note

É possível adicionar `CloudFront-Viewer-Address` e `CloudFront-Viewer-ASN` a uma política de solicitação de origem, mas não a uma política de cache.

- `CloudFront-Viewer-Country`: contém o código de duas letras do país do visualizador. Para obter uma lista de códigos de país, consulte [ISO 3166-1 alfa-2](#).
- `CloudFront-Viewer-City`: contém o nome da cidade do visualizador.

Quando você adiciona os cabeçalhos a seguir, o CloudFront os aplica a todas as solicitações exceto aquelas que se originam da rede da AWS:

- `CloudFront-Viewer-Country-Name`: contém o nome do país do visualizador.
- `CloudFront-Viewer-Country-Region`: contém um código (de até três caracteres) que representa a região do visualizador. A região é a subdivisão de primeiro nível (a mais ampla ou menos específica) do código [ISO 3166-2](#).
- `CloudFront-Viewer-Country-Region-Name`: contém o nome da região do visualizador. A região é a subdivisão de primeiro nível (a mais ampla ou menos específica) do código [ISO 3166-2](#).
- `CloudFront-Viewer-Latitude`: contém a latitude aproximada do visualizador.
- `CloudFront-Viewer-Longitude`: contém a longitude aproximada do visualizador.
- `CloudFront-Viewer-Metro-Code`: contém o código metro do visualizador. Esse código só está presente quando o visualizador está nos Estados Unidos.
- `CloudFront-Viewer-Postal-Code`: contém o CEP do visualizador.
- `CloudFront-Viewer-Time-Zone` contém o fuso horário do visualizador, no [formato de banco de dados de fuso horário da IANA](#) (por exemplo, `America/Los_Angeles`).

 Note

`CloudFront-Viewer-City`, `CloudFront-Viewer-Metro-Code` e `CloudFront-Viewer-Postal-Code` podem não estar disponíveis para todos os endereços IP. Alguns

endereços IP não podem ser geolocalizados com especificidade suficiente para receber essas informações.

Cabeçalhos para determinar a estrutura de cabeçalho do visualizador

Agora você pode adicionar os cabeçalhos a seguir para ajudar a identificar o visualizador com base nos cabeçalhos que ele envia. Por exemplo, navegadores diferentes podem enviar cabeçalhos HTTP em determinada ordem. Se o navegador especificado no cabeçalho `User-Agent` não corresponder à ordem de cabeçalho esperada desse navegador, você poderá negar a solicitação. Além disso, se o valor `CloudFront-Viewer-Header-Count` não corresponder ao número de cabeçalhos em `CloudFront-Viewer-Header-Order`, você poderá negar a solicitação.

- `CloudFront-Viewer-Header-Order`: contém os nomes dos cabeçalhos do visualizador na ordem solicitada, separados por um sinal de dois pontos. Por exemplo: `CloudFront-Viewer-Header-Order: Host:User-Agent:Accept:Accept-Encoding`. Os cabeçalhos além do limite de caracteres de 7.680 são truncados.
- `CloudFront-Viewer-Header-Count`: contém o número total de cabeçalhos do visualizador.

Outros cabeçalhos do CloudFront

É possível adicionar os seguintes cabeçalhos para determinar o URI, os parâmetros da string de consulta, bem como os valores, o protocolo, a versão, a impressão digital JA3 e os detalhes da conexão TLS da solicitação original do visualizador:

- `CloudFront-Error-Args`: contém o URI da solicitação original recebida do visualizador.
- `CloudFront-Error-Uri`: contém os parâmetros e os valores da string de consulta da solicitação original.
- `CloudFront-Forwarded-Proto`: contém o protocolo da solicitação do visualizador (HTTP ou HTTPS).
- `CloudFront-Viewer-Http-Version`: contém a versão HTTP da solicitação do visualizador.
- `CloudFront-Viewer-JA3-Fingerprint`: contém a [impressão digital JA3](#) do visualizador. A impressão digital JA3 pode ajudar você a determinar se a solicitação vem de um cliente conhecido, se é um malware ou um bot malicioso, ou se é uma aplicação esperada (na lista de permissões). Esse cabeçalho depende do pacote `Client Hello SSL/TLS` do visualizador e está presente apenas em solicitações HTTPS.

Note

Você pode adicionar CloudFront-Viewer-JA3-Fingerprint em uma [política de solicitação de origem](#), mas não em uma [política de cache](#).

- CloudFront-Viewer-TLS: contém a versão SSL/TLS da cifra e as informações sobre o handshake de SSL/TLS que foi usado para a conexão entre o visualizador e o CloudFront. O valor do cabeçalho está no seguinte formato:

```
SSL/TLS_version:cipher:handshake_information
```

Para *handshake_information*, o cabeçalho contém um dos seguintes valores:

- fullHandshake: foi realizado um handshake completo para a sessão SSL/TLS.
- sessionResumed: uma sessão anterior de SSL/TLS foi retomada.
- connectionReused: uma conexão SSL/TLS anterior foi reutilizada.

Veja a seguir alguns exemplos de valores para esse cabeçalho:

```
TLSv1.3:TLS_AES_128_GCM_SHA256:sessionResumed
```

```
TLSv1.2:ECDHE-ECDSA-AES128-GCM-SHA256:connectionReused
```

```
TLSv1.1:ECDHE-RSA-AES128-SHA256:fullHandshake
```

```
TLSv1:ECDHE-RSA-AES256-SHA:fullHandshake
```

Para obter a lista completa de possíveis versões e criptografias SSL/TLS que podem estar nesse valor de cabeçalho, consulte [the section called “Protocolos e cifras compatíveis entre visualizadores e o CloudFront”](#).

Note

Você pode adicionar CloudFront-Viewer-TLS em uma [política de solicitação de origem](#), mas não em uma [política de cache](#).

Noções básicas sobre como as políticas de solicitação de origem e as políticas de cache funcionam juntas

É possível usar uma [política de solicitação de origem](#) do CloudFront para controlar as solicitações que o CloudFront envia para a origem, que são chamadas de solicitações de origem. Para usar uma política de solicitação de origem, é necessário anexar uma [política de cache](#) ao mesmo comportamento de cache. Não é possível usar uma política de solicitação de origem em um comportamento de cache sem uma política de cache. Para ter mais informações, consulte [Controlar as solicitações de origem com uma política](#).

As políticas de solicitação de origem e as políticas de cache funcionam juntas para determinar os valores que o CloudFront inclui nas solicitações de origem. Todas as strings de consulta de URL, cabeçalhos HTTP e cookies que você especificar na chave de cache (usando uma política de cache) são automaticamente incluídos nas solicitações de origem. Quaisquer strings de consulta, cabeçalhos e cookies adicionais especificados em uma política de solicitação de origem também serão incluídos nas solicitações de origem (mas não na chave de cache).

As políticas de solicitação de origem e as políticas de cache têm configurações que podem parecer mutuamente conflitantes. Por exemplo, uma política pode permitir determinados valores enquanto outra os bloqueia. A tabela a seguir explica quais valores o CloudFront inclui nas solicitações de origem quando você usa as configurações de uma política de solicitação de origem e uma política de cache em conjunto. Essas configurações geralmente se aplicam a todos os tipos de valores (strings de consulta, cabeçalhos e cookies), com a exceção de que não é possível especificar todos os cabeçalhos ou usar uma lista de bloqueio de cabeçalhos em uma política de cache.

	Política de solicitação de origem			
	Nenhum	Todos	Lista de permissões	Lista de bloqueios

Política de cache

Nenhum	Nenhum valor da solicitação do visualizador é incluído na solicitação de origem, exceto	Todos os valores da solicitação do visualizador são incluídos na solicitação de origem.	Somente os valores especificados na política de solicitação de origem são incluídos na	Todos os valores da solicitação do visualizador, exceto aqueles especificados na política de
--------	---	---	--	--

	Política de solicitação de origem			
	Nenhum	Todos	Lista de permissões	Lista de bloqueios
	os padrões, que estão incluídos em todas as solicitações de origem. Para ter mais informações, consulte Controlar as solicitações de origem com uma política .		solicitação de origem.	solicitação de origem, são incluídos na solicitação de origem.

	Política de solicitação de origem			
	Nenhum	Todos	Lista de permissões	Lista de bloqueios
<p>Todos</p> <p>Nota: não é possível especificar todos os cabeçalhos em uma política de cache.</p>	<p>Todas as strings de consulta e cookies da solicitação do visualizador são incluídos na solicitação de origem.</p>	<p>Todos os valores da solicitação do visualizador são incluídos na solicitação de origem.</p>	<p>Todas as strings de consulta e cookies da solicitação do visualizador e quaisquer cabeçalhos especificados na política de solicitação de origem são incluídos na solicitação de origem.</p>	<p>Todas as strings de consulta e cookies da solicitação do visualizador são incluídos na solicitação de origem, mesmo aqueles especificados na lista de bloqueio da política de solicitação de origem. A configuração da política de cache substitui a lista de bloqueio da política de solicitação de origem.</p>

	Política de solicitação de origem			
	Nenhum	Todos	Lista de permissões	Lista de bloqueios
Lista de permissões	Somente os valores especificados na solicitação do visualizador são incluídos na solicitação de origem.	Todos os valores da solicitação do visualizador são incluídos na solicitação de origem.	Todos os valores especificados na política de cache ou na política de solicitação de origem são incluídos na solicitação de origem.	Os valores especificados na política de cache são incluídos na solicitação de origem, mesmo que esses valores estejam especificados na lista de bloqueio da política de solicitação de origem. A lista de permissões da política de cache substitui a lista de bloqueio da política de solicitação de origem.

	Política de solicitação de origem			
	Nenhum	Todos	Lista de permissões	Lista de bloqueios
<p>Lista de bloqueios</p> <p>Nota: não é possível especificar cabeçalhos em uma lista de bloqueio da política de cache.</p>	<p>Todas as cadeias de caracteres de consulta e cookies da solicitação do visualizador, exceto aqueles especificados, são incluídos na solicitação de origem.</p>	<p>Todos os valores da solicitação do visualizador são incluídos na solicitação de origem.</p>	<p>Os valores especificados na política de solicitação de origem são incluídos na solicitação de origem, mesmo que esses valores estejam especificados na lista de bloqueio da política de cache. A lista de permissões da política de solicitação de origem substitui a lista de bloqueio da política de cache.</p>	<p>Todos os valores da solicitação do visualizador, exceto aqueles especificados na política de cache ou na política de solicitação de origem, são incluídos na solicitação de origem.</p>

Adicionar ou remover cabeçalhos HTTP em respostas do CloudFront com uma política

É possível configurar o CloudFront para modificar os cabeçalhos HTTP nas respostas que ele envia aos visualizadores (navegadores Web e outros clientes). O CloudFront pode remover cabeçalhos recebidos da origem ou adicionar cabeçalhos à resposta antes de enviá-la aos espectadores. Fazer essas alterações não requer escrever código nem alterar a origem.

Por exemplo, você pode remover cabeçalhos como `X-Powered-By` e `Vary` para que o CloudFront não inclua esses cabeçalhos nas respostas que envia aos espectadores. Ou você pode adicionar cabeçalhos HTTP, como os seguintes:

- Um cabeçalho `Cache-Control` para controlar o cache do navegador.
- Um cabeçalho `Access-Control-Allow-Origin` para habilitar o compartilhamento de recursos entre origens (CORS). Você também pode adicionar outros cabeçalhos de CORS.
- Um conjunto de cabeçalhos de segurança comuns, como `Strict-Transport-Security`, `Content-Security-Policy` e `X-Frame-Options`.
- Um cabeçalho `Server-Timing` para ver informações relacionadas à performance e ao roteamento da solicitação e da resposta por meio do CloudFront.

Para especificar os cabeçalhos que o CloudFront adiciona ou remove em respostas HTTP, use uma política de cabeçalhos de resposta. Anexe uma política de cabeçalhos de resposta a um ou mais comportamentos de cache. O CloudFront modifica os cabeçalhos nas respostas que envia para solicitações que correspondem ao comportamento de cache. O CloudFront modifica os cabeçalhos nas respostas que encaminha do cache e aquelas que encaminha da origem. Se a resposta da origem incluir um ou mais cabeçalhos que estejam adicionados em uma política de cabeçalhos de resposta, a política poderá especificar se o CloudFront usará o cabeçalho recebido da origem ou substituirá o cabeçalho por aquele na política de cabeçalhos da resposta.

O CloudFront fornece algumas políticas de cache predefinidas, conhecidas como políticas gerenciadas, para casos de uso comuns. É possível [usar essas políticas gerenciadas](#) ou criar suas próprias políticas. Você pode anexar uma única política de cabeçalhos de resposta a vários comportamentos de cache em várias distribuições em sua Conta da AWS.

Para obter mais informações, consulte os tópicos a seguir.

Tópicos

- [Noções básicas das políticas de cabeçalhos de resposta](#)
- [Criar políticas de cabeçalhos de resposta](#)
- [Usar políticas de cabeçalhos de resposta gerenciadas](#)

Noções básicas das políticas de cabeçalhos de resposta

Você pode usar uma política de cabeçalhos de resposta para especificar os cabeçalhos HTTP que o Amazon CloudFront remove ou adiciona às respostas enviadas aos visualizadores. Para obter mais informações sobre políticas de cabeçalhos de respostas e os motivos para usá-las, consulte [Adicionar ou remover cabeçalhos de resposta com uma política](#).

Os tópicos a seguir explicam as configurações em uma política de cabeçalhos de resposta. As configurações são agrupadas em categorias, que são representadas nos tópicos a seguir.

Tópicos

- [Detalhes da política \(metadados\)](#)
- [Cabeçalhos de CORS](#)
- [Cabeçalhos de segurança](#)
- [Cabeçalhos personalizados](#)
- [Remover cabeçalhos](#)
- [Cabeçalho de temporização do servidor](#)

Detalhes da política (metadados)

As configurações de detalhes da política contêm metadados sobre uma política de cabeçalhos de resposta.

- **Name (Nome):** um nome para identificar a política dos cabeçalhos de resposta. No console, você usa o nome para anexar a política a um comportamento de cache.
- **Description (Descrição, opcional):** um comentário para descrever a política dos cabeçalhos de resposta. Isso é opcional, mas pode ajudar a identificar a finalidade da política.

Cabeçalhos de CORS

As configurações de compartilhamento de recursos entre origens (CORS) permitem adicionar e configurar cabeçalhos de CORS em uma política de cabeçalhos de resposta.

Essa lista se concentra em como especificar as configurações e os valores válidos em uma política de cabeçalhos de resposta. Para obter mais informações sobre cada um desses cabeçalhos e como eles são usados para solicitações e respostas de CORS do mundo real, consulte [compartilhamento de recursos entre origens](#) no MDN Web Docs e a [Especificação do protocolo CORS](#).

Access-Control-Allow-Credentials

Essa é uma configuração booleana (`true` ou `false`) que determina se o CloudFront adiciona o cabeçalho `Access-Control-Allow-Credentials` em respostas às solicitações de CORS. Quando essa configuração for definida como `true`, o CloudFront adicionará o cabeçalho `Access-Control-Allow-Credentials: true` em respostas às solicitações do CORS. Caso contrário, o CloudFront não adicionará esse cabeçalho às respostas.

Access-Control-Allow-Headers

Especifica os nomes de cabeçalho que o CloudFront usa como valores para o cabeçalho `Access-Control-Allow-Headers` em respostas às solicitações de comprovação de CORS. Os valores válidos para essa configuração incluem nomes de cabeçalho HTTP ou o caractere curinga (*), que indica que todos os cabeçalhos são permitidos.

Note

O cabeçalho `Authorization` não pode usar um caractere curinga e deve ser listado explicitamente.

Exemplos de uso válido do caractere curinga

Exemplo	Corresponderá	Não corresponderá
<code>x-amz-*</code>	<code>x-amz-test</code> <code>x-amz-</code>	<code>x-amz</code>
<code>x-*-amz</code>	<code>x-test-amz</code>	

Exemplo	Corresponderá	Não corresponderá
	x -- amz	
*	Todos os cabeçalhos, exceto Authorization .	Authorization

Access-Control-Allow-Methods

Especifica os métodos HTTP que o CloudFront usa como valores para o cabeçalho Access-Control-Allow-Methods em respostas às solicitações de comprovação de CORS. Os valores válidos são GET, DELETE, HEAD, OPTIONS, PATCH, POST, PUT e ALL. ALL é um valor especial que inclui todos os métodos de HTTP listados.

Access-Control-Allow-Origin

Especifica os valores que o CloudFront pode usar no cabeçalho de resposta Access-Control-Allow-Origin. Os valores válidos para essa configuração incluem uma origem específica (como `http://www.example.com`) ou o caractere curinga (*), o que indica que todas as origens são permitidas.

Observações

- O caractere curinga (*) é permitido como a parte mais à esquerda do subdomínio (*.example.org).
- O caractere curinga (*) não é permitido nas seguintes posições:
 - Domínios de nível superior (example.*)
 - À direita dos subdomínios (test.*.example.org) ou dentro de qualquer subdomínio (*test.example.org)
 - Dentro dos termos (exa*mples.org)

Para ver exemplos de como usar o caractere curinga, consulte a tabela a seguir.

Exemplo	Corresponderá	Não corresponderá
<code>http://*.example.org</code>	<code>http://www.example.org</code>	<code>https://test.example.org</code>

Exemplo	Corresponderá	Não corresponderá
	<pre>http://test.example.org</pre> <pre>http://test.example.org:123</pre>	<pre>https://test.example.org:123</pre>
<code>*.example.org</code>	<pre>test.example.org</pre> <pre>test.test.example.org</pre> <pre>.example.org</pre> <pre>http://test.example.org</pre> <pre>https://test.example.org</pre> <pre>http://test.example.org:123</pre> <pre>https://test.example.org:123</pre>	
<code>example.org</code>	<pre>http://example.org</pre> <pre>https://example.org</pre>	
<code>http://example.org</code>		<pre>https://example.org</pre> <pre>http://example.org:123</pre>
<code>http://example.org:*</code>	<pre>http://example.org:123</pre> <pre>http://example.org</pre>	

Exemplo	Corresponderá	Não corresponderá
<code>http://example.org:1*3</code>	<code>http://example.org:123</code> <code>http://example.org:1893</code> <code>http://example.org:13</code>	
<code>*.example.org:1*</code>	<code>test.example.org:123</code>	

Access-Control-Expose-Headers

Especifica os nomes de cabeçalho que o CloudFront usa como valores para o cabeçalho `Access-Control-Expose-Headers` em respostas às solicitações de CORS. Os valores válidos para essa configuração incluem nomes de cabeçalho de HTTP ou o caractere curinga (*).

Access-Control-Max-Age

Um número de segundos, que o CloudFront usa como valor para o cabeçalho `Access-Control-Max-Age` em respostas às solicitações de comprovação de CORS.

Substituição de origem

Uma configuração booliana que determina como o CloudFront se comporta quando a resposta da origem contém um dos cabeçalhos de CORS que também está na política.

- Quando definida como `true` e a resposta da origem contiver um cabeçalho de CORS que também esteja na política, o CloudFront adicionará o cabeçalho de CORS da política à resposta. Depois, o CloudFront envia essa resposta ao visualizador. O CloudFront ignora o cabeçalho que recebeu da origem.
- Quando definido como `false` e a resposta de origem contém um cabeçalho de CORS (independentemente de esse cabeçalho estar na política), o CloudFront inclui na resposta o cabeçalho de CORS que recebeu da origem. O CloudFront não adicionará cabeçalhos de CORS da política à resposta que é enviada ao visualizador.

Cabeçalhos de segurança

Você pode usar as configurações de cabeçalhos de segurança para adicionar e configurar vários cabeçalhos de resposta de HTTP relacionados à segurança em uma política de cabeçalhos de resposta.

Essa lista descreve como você pode especificar as configurações e valores válidos em uma política de cabeçalhos de resposta. Para obter mais informações sobre cada um desses cabeçalhos e como eles são usados em respostas de HTTP do mundo real, consulte os links para o MDN Web Docs.

Content-Security-Policy

Especifica as diretivas de política de segurança de conteúdo que o CloudFront usa como valores para o cabeçalho de resposta `Content-Security-Policy`.

Para obter mais informações sobre esse cabeçalho e diretivas de políticas válidas, consulte [Content-Security-Policy](#) no MDN Web Docs.

Note

O valor do cabeçalho `Content-Security-Policy` é limitado a 1783 caracteres.

Política de referenciador

Especifica a diretiva de política do referenciador que o CloudFront usa como valor para o cabeçalho de resposta `Referrer-Policy`. Os valores válidos para essa configuração são `no-referrer`, `no-referrer-when-downgrade`, `origin`, `origin-when-cross-origin`, `same-origin`, `strict-origin`, `strict-origin-when-cross-origin` e `unsafe-url`.

Para obter mais informações sobre esse cabeçalho e essas diretivas, consulte [Referrer-Policy](#) no MDN Web Docs.

Strict-Transport-Security

Especifica as diretivas e as configurações que o CloudFront usa como valor para o cabeçalho de resposta `Strict-Transport-Security`. Para essa configuração, você especifica separadamente:

- Um número de segundos, que o CloudFront usa como valor para a diretiva `max-age` desse cabeçalho

- Uma configuração booliana (`true` ou `false`) para `preload`, que determina se o CloudFront inclui a diretiva `preload` no valor desse cabeçalho
- Uma configuração booliana (`true` ou `false`) para `includeSubDomains`, que determina se o CloudFront inclui a diretiva `includeSubDomains` no valor desse cabeçalho

Para obter mais informações sobre esse cabeçalho e essas diretivas, consulte [Strict-Transport-Security](#) no MDN Web Docs.

X-Content-Type-Options

Essa é uma configuração booliana (`true` ou `false`) que determina se o CloudFront adiciona o cabeçalho `X-Content-Type-Options` às respostas. Quando essa configuração for `true`, o CloudFront adicionará o cabeçalho `X-Content-Type-Options: nosniff` às respostas. Caso contrário, o CloudFront não adicionará esse cabeçalho.

Para obter mais informações sobre esse cabeçalho, consulte [X-Content-Type-Options](#) no MDN Web Docs.

X-Frame-Options

Especifica a diretiva que o CloudFront usa como valor para o cabeçalho de resposta `X-Frame-Options`. Os valores válidos para essa configuração são `DENY` ou `SAMEORIGIN`.

Para obter mais informações sobre esse cabeçalho e essas diretivas, consulte [X-Frame-Options](#) no MDN Web Docs.

X-XSS-Protection

Especifica as diretivas e as configurações que o CloudFront usa como valor para o cabeçalho de resposta `X-XSS-Protection`. Para essa configuração, você especifica separadamente:

- Uma configuração `X-XSS-Protection` de `0` (desabilita a filtragem de XSS) ou `1` (habilita a filtragem de XSS)
- Uma configuração booliana (`true` ou `false`) para `block`, que determina se o CloudFront inclui a diretiva `mode=block` no valor desse cabeçalho
- Um URI de relatório, que determina se o CloudFront deve incluir a diretiva `report=reporting URI` no valor desse cabeçalho

Você pode especificar `true` para `block`, ou pode especificar um URI de relatório, mas não pode especificar os dois juntos. Para obter mais informações sobre esse cabeçalho e essas diretivas, consulte [X-XSS-Protection](#) no MDN Web Docs.

Substituição de origem

Cada uma dessas configurações de cabeçalhos de segurança contém uma configuração booleana (`true` ou `false`) que determina como o CloudFront se comporta quando a resposta da origem contém esse cabeçalho.

Quando essa configuração for definida como `true` e a resposta da origem contiver o cabeçalho, o CloudFront adicionará o cabeçalho na política à resposta que envia ao visualizador. Ele ignora o cabeçalho que recebeu da origem.

Quando essa configuração for definida como `false` e a resposta da origem contiver o cabeçalho, o CloudFront incluirá o cabeçalho recebido da origem na resposta que envia ao visualizador.

Quando a resposta da origem não contiver o cabeçalho, o CloudFront adicionará o cabeçalho da política à resposta que envia ao visualizador. O CloudFront faz isso quando essa configuração está definida como `true` ou `false`.

Cabeçalhos personalizados

Você pode usar as configurações de cabeçalhos personalizados para adicionar e configurar cabeçalhos de HTTP personalizados em uma política de cabeçalhos de resposta. O CloudFront adiciona esses cabeçalhos a cada resposta que ele retorna aos espectadores. Para cada cabeçalho personalizado, você também especifica o valor do cabeçalho, embora a especificação de um valor seja opcional. Isso ocorre porque o CloudFront pode adicionar um cabeçalho de resposta sem valor.

Cada cabeçalho personalizado também tem sua própria configuração de substituição de origem:

- Quando essa configuração for definida como `true` e a resposta da origem contiver o cabeçalho personalizado que está na política, o CloudFront adicionará o cabeçalho personalizado da política à resposta que envia ao visualizador. Ele ignora o cabeçalho que recebeu da origem.
- Quando essa configuração for `false` e a resposta da origem contiver o cabeçalho personalizado que está na política, o CloudFront incluirá o cabeçalho personalizado recebido da origem na resposta que envia ao visualizador.
- Quando a resposta da origem não contiver o cabeçalho personalizado que está na política, o CloudFront adicionará o cabeçalho personalizado na política à resposta que envia ao visualizador. O CloudFront faz isso quando essa configuração está definida como `true` ou `false`.

Remover cabeçalhos

Você pode especificar cabeçalhos que deseja que o CloudFront remova das respostas que recebe da origem para que os cabeçalhos não sejam incluídos nas respostas que o CloudFront envia aos visualizadores. O CloudFront remove os cabeçalhos de cada resposta que envia aos visualizadores, independentemente de os objetos serem fornecidos do cache do CloudFront ou da origem. Por exemplo, você pode remover cabeçalhos que não são úteis para navegadores, como `X-Powered-By` ou `Vary`, para que o CloudFront remova esses cabeçalhos das respostas que envia aos visualizadores.

Quando você especifica cabeçalhos a serem removidos usando uma política de cabeçalhos de resposta, primeiro o CloudFront remove os cabeçalhos, depois adiciona os cabeçalhos especificados em outras seções da política de cabeçalhos de resposta (cabeçalhos de CORS, cabeçalhos de segurança, cabeçalhos personalizados etc.). Se você especificar um cabeçalho para remover, mas também adicionar o mesmo cabeçalho em outra seção da política, o CloudFront incluirá o cabeçalho nas respostas que ele envia aos visualizadores.

Note

Você pode usar uma política de cabeçalhos de resposta para remover os cabeçalhos `Server` e `Date` que o CloudFront recebeu da origem, para que esses cabeçalhos (conforme recebidos da origem) não sejam incluídos nas respostas que o CloudFront envia aos visualizadores. No entanto, se você fizer isso, o CloudFront adicionará sua própria versão desses cabeçalhos às respostas enviadas aos visualizadores. Para o cabeçalho `Server` que o CloudFront adiciona, o valor é `CloudFront`.

Cabeçalhos que não podem ser removidos

Você não pode remover os cabeçalhos a seguir usando uma política de cabeçalhos de resposta. Se você especificar esses cabeçalhos na seção `Remove headers` (Remover cabeçalhos) de uma política de cabeçalhos de resposta (`ResponseHeadersPolicyRemoveHeadersConfig` na API), receberá um erro.

- `Connection`
- `Content-Encoding`
- `Content-Length`

- Expect
- Host
- Keep-Alive
- Proxy-Authenticate
- Proxy-Authorization
- Proxy-Connection
- Trailer
- Transfer-Encoding
- Upgrade
- Via
- Warning
- X-Accel-Buffering
- X-Accel-Charset
- X-Accel-Limit-Rate
- X-Accel-Redirect
- X-Amz-Cf-.*
- X-Amzn-Auth
- X-Amzn-Cf-Billing
- X-Amzn-Cf-Id
- X-Amzn-Cf-Xff
- X-Amzn-ErrorType
- X-Amzn-Fle-Profile
- X-Amzn-Header-Count
- X-Amzn-Header-Order
- X-Amzn-Lambda-Integration-Tag
- X-Amzn-RequestId
- X-Cache
- X-Edge-.*
- X-Forwarded-Proto
- X-Real-IP

Cabeçalho de temporização do servidor

Use a configuração de cabeçalho `Server-Timing` para habilitar o cabeçalho `Server-Timing` em respostas HTTP enviadas do CloudFront. Você pode usar esse cabeçalho para visualizar métricas que podem ajudar a obter insights sobre o comportamento e a performance do CloudFront e sua origem. Por exemplo, você pode ver qual camada de cache forneceu um acerto de cache. Ou, você poderá ver a latência do primeiro byte da origem se houver uma falha de cache. As métricas no cabeçalho `Server-Timing` podem ajudar a solucionar problemas ou testar a eficiência da configuração do CloudFront ou da origem.

Para obter mais informações sobre como usar o cabeçalho de `Server-Timing` com o CloudFront, consulte os tópicos a seguir.

Para habilitar o cabeçalho `Server-Timing`, [crie \(ou edite\) uma política de cabeçalhos de resposta](#).

Tópicos

- [Taxa de amostragem e cabeçalho de solicitação `Pragma`](#)
- [Cabeçalho de `Server-Timing` da origem](#)
- [Métricas de cabeçalho de temporização do servidor](#)
- [Exemplos de cabeçalho de temporização do servidor](#)

Taxa de amostragem e cabeçalho de solicitação `Pragma`

Ao habilitar o cabeçalho `Server-Timing` em uma política de cabeçalhos de resposta, você também especifica a taxa de amostragem. A taxa de amostragem é um número de 0 a 100 (incluído) que especifica a porcentagem de respostas às quais você deseja que o CloudFront adicione o cabeçalho `Server-Timing`. Quando você define a taxa de amostragem como 100, o CloudFront adiciona o cabeçalho `Server-Timing` às respostas HTTP de todas as solicitações que corresponderem ao comportamento de cache ao qual a política de cabeçalhos de resposta está anexada. Quando você o define como 50, o CloudFront adiciona o cabeçalho a 50% das respostas das solicitações que corresponderem ao comportamento do cache. Você pode definir a taxa de amostragem para qualquer número de 0 a 100 com até quatro casas decimais.

Quando a taxa de amostragem é definida como um número menor que 100, você não pode controlar a quais respostas o CloudFront adiciona o cabeçalho `Server-Timing`, apenas a porcentagem. No entanto, você pode adicionar o cabeçalho `Pragma` com um valor definido como `server-timing` em uma solicitação HTTP para receber o cabeçalho `Server-Timing` na resposta a essa solicitação.

Isso funciona independentemente da taxa de amostragem definida. Mesmo quando a taxa de amostragem estiver definida como zero (0), o CloudFront adicionará o cabeçalho `Server-Timing` à resposta se a solicitação contiver o cabeçalho `Pragma: server-timing`.

Cabeçalho de Server-Timing da origem

Quando há uma falha no cache e o CloudFront encaminha a solicitação à origem, a origem pode incluir um cabeçalho de `Server-Timing` em sua resposta ao CloudFront. Nesse caso, o CloudFront adiciona suas [métricas](#) ao cabeçalho de `Server-Timing` que recebeu da origem. A resposta que o CloudFront envia ao visualizador contém um único cabeçalho de `Server-Timing` que inclui o valor que veio da origem e as métricas que o CloudFront adicionou. O valor do cabeçalho da origem pode estar no final ou entre dois conjuntos de métricas que o CloudFront adiciona ao cabeçalho.

Quando há uma ocorrência de cache, a resposta que o CloudFront envia ao visualizador contém um único cabeçalho de `Server-Timing` que inclui apenas as métricas do CloudFront no valor do cabeçalho (o valor da origem não está incluído).

Métricas de cabeçalho de temporização do servidor

Quando o CloudFront adiciona o cabeçalho `Server-Timing` a uma resposta HTTP, o valor do cabeçalho contém uma ou mais métricas que podem ajudar a obter insights sobre o comportamento e a performance do CloudFront. A lista a seguir contém todas as métricas e seus possíveis valores. Um cabeçalho `Server-Timing` contém apenas algumas dessas métricas, dependendo da natureza da solicitação e da resposta por meio do CloudFront.

Algumas dessas métricas estão incluídas no cabeçalho `Server-Timing` com apenas um nome (sem valor). Outras são um nome e um valor. Quando uma métrica tem um valor, o nome e valor são separados por um ponto e vírgula (;). Quando o cabeçalho contém mais de uma métrica, as métricas são separadas por uma vírgula (,).

`cdn-cache-hit`

O CloudFront forneceu uma resposta do cache sem fazer uma solicitação para a origem.

`cdn-cache-refresh`

O CloudFront forneceu uma resposta do cache depois de enviar uma solicitação para a origem a fim de verificar se o objeto armazenado em cache ainda é válido. Nesse caso, o CloudFront não recuperou o objeto completo da origem.

cdn-cache-miss

O CloudFront não forneceu a resposta do cache. Nesse caso, o CloudFront solicitou o objeto completo da origem antes de retornar a resposta.

cdn-pop

Contém um valor que descreve qual ponto de presença (POP) do CloudFront processou a solicitação.

cdn-rid

Contém um valor com o identificador exclusivo do CloudFront para a solicitação. Você pode usar esse identificador de solicitação (RID) ao solucionar problemas com AWS Support.

cdn-hit-layer

Esta métrica está presente quando o CloudFront fornece uma resposta do cache sem fazer uma solicitação à origem. Ela contém um dos seguintes valores:

- EDGE: o CloudFront forneceu a resposta em cache de um local do POP.
- REC: o CloudFront forneceu a resposta em cache de um local do [cache de borda regional](#) (REC).
- Origin Shield: o CloudFront forneceu a resposta em cache do REC que está agindo como o [escudo de origem](#).

cdn-upstream-layer

Quando o CloudFront solicita o objeto completo da origem, essa métrica está presente e contém um dos seguintes valores:

- EDGE: um local do POP enviou a solicitação diretamente para a origem.
- REC: um local do REC enviou a solicitação diretamente à origem.
- Origin Shield: o REC que está agindo como [escudo de origem](#) enviou a solicitação diretamente à origem.

cdn-upstream-dns

Contém um valor com o número de milissegundos que foram gastos recuperando o registro DNS para a origem. Um valor de zero (0) indica que o CloudFront usou um resultado de DNS em cache ou reutilizou uma conexão existente.

cdn-upstream-connect

Contém um valor com o número de milissegundos entre quando a solicitação DNS de origem foi concluída e uma conexão TCP (e TLS, se aplicável) com a origem foi concluída. Um valor de zero (0) indica que o CloudFront reutilizou uma conexão existente.

cdn-upstream-fbl

Contém um valor com o número de milissegundos entre quando a solicitação HTTP de origem foi concluída e quando o primeiro byte foi recebido na resposta da origem (latência do primeiro byte).

cdn-downstream-fbl

Contém um valor com o número de milissegundos entre quando o local da borda terminou de receber a solicitação e quando enviou o primeiro byte da resposta ao visualizador.

Exemplos de cabeçalho de temporização do servidor

Veja a seguir exemplos de um cabeçalho `Server-Timing` que um visualizador pode receber do CloudFront quando a configuração do cabeçalho `Server-Timing` está habilitada.

Example – cache miss

O exemplo a seguir mostra um cabeçalho `Server-Timing` que um visualizador pode receber quando o objeto solicitado não está no cache do CloudFront.

```
Server-Timing: cdn-upstream-layer;desc="EDGE",cdn-upstream-dns;dur=0,cdn-upstream-connect;dur=114,cdn-upstream-fbl;dur=177,cdn-cache-miss,cdn-pop;desc="PHX50-C2",cdn-rid;desc="yNPsyYn7skvTzwWkq3Wcc8Nj_foxUjQe9H1ifslzWhb0w7aLbFvGg==",cdn-downstream-fbl;dur=436
```

Esse cabeçalho de `Server-Timing` indica o seguinte:

- A solicitação de origem foi enviada de um local de ponto de presença (POP) do CloudFront (`cdn-upstream-layer;desc="EDGE"`).
- O CloudFront usou um resultado DNS armazenado em cache para a origem (`cdn-upstream-dns;dur=0`).
- Foram necessários 114 milissegundos para o CloudFront concluir a conexão TCP (e TLS, se aplicável) com a origem (`cdn-upstream-connect;dur=114`).
- Foram necessários 177 milissegundos para o CloudFront receber o primeiro byte da resposta da origem, após concluir a solicitação (`cdn-upstream-fbl;dur=177`).

- O objeto solicitado não estava no cache do CloudFront (`cdn-cache-miss`).
- A solicitação foi recebida no local da borda identificado pelo código `PHX50-C2` (`cdn-pop;desc="PHX50-C2"`).
- O ID exclusivo do CloudFront para essa solicitação era `yNPsyYn7skvTzwWkq3Wcc8Nj_foxUjQe9H1ifslzWhb0w7aLbFvGg==` (`cdn-rid;desc="yNPsyYn7skvTzwWkq3Wcc8Nj_foxUjQe9H1ifslzWhb0w7aLbFvGg=="`).
- Foram necessários 436 milissegundos para o CloudFront enviar o primeiro byte da resposta ao visualizador, depois de receber a solicitação do visualizador (`cdn-downstream-fb1;dur=436`).

Example – cache hit

O exemplo a seguir mostra um cabeçalho `Server-Timing` que um visualizador pode receber quando o objeto solicitado está no cache do CloudFront.

```
Server-Timing: cdn-cache-hit,cdn-pop;desc="SEA19-C1",cdn-rid;desc="nQBz4aJU2kP9iC3KHEq7vFxfMoZu-VYBwGzkW9di0peVc7xsrlKj-g==",cdn-hit-layer;desc="REC",cdn-downstream-fb1;dur=137
```

Esse cabeçalho de `Server-Timing` indica o seguinte:

- O objeto solicitado estava no cache (`cdn-cache-hit`).
- A solicitação foi recebida no local da borda identificado pelo código `SEA19-C1` (`cdn-pop;desc="SEA19-C1"`).
- O ID exclusivo do CloudFront para essa solicitação era `nQBz4aJU2kP9iC3KHEq7vFxfMoZu-VYBwGzkW9di0peVc7xsrlKj-g==` (`cdn-rid;desc="nQBz4aJU2kP9iC3KHEq7vFxfMoZu-VYBwGzkW9di0peVc7xsrlKj-g=="`).
- O objeto solicitado foi armazenado em cache em um local de cache de borda regional (`REC`) (`cdn-hit-layer;desc="REC"`).
- Foram necessários 137 milissegundos para CloudFront enviar o primeiro byte da resposta ao visualizador, depois de receber a solicitação do visualizador (`cdn-downstream-fb1;dur=137`).

Criar políticas de cabeçalhos de resposta

Você pode usar uma política de cabeçalhos para especificar os cabeçalhos HTTP que o Amazon CloudFront adiciona ou remove em respostas HTTP. Para obter mais informações sobre políticas de

cabeçalhos de respostas e os motivos para usá-las, consulte [Adicionar ou remover cabeçalhos de resposta com uma política](#).

Você pode criar uma política de cabeçalhos de resposta no console do CloudFront. Ou, você pode criar uma usando o AWS CloudFormation, a AWS Command Line Interface (AWS CLI) ou a API do CloudFront. Depois de criar uma política de cabeçalhos de resposta, anexe-a a um ou mais comportamentos de cache em uma distribuição do CloudFront.

Antes de criar uma política de cabeçalhos de resposta personalizada, verifique se uma das [políticas de cabeçalhos de resposta gerenciadas](#) se encaixa no seu caso de uso. Caso contrário, você pode anexá-la ao comportamento de cache. Dessa forma, você não precisa criar nem gerenciar sua própria política de cabeçalhos de resposta.

Console

Para criar uma política de cabeçalhos de resposta (console)

1. Faça login no AWS Management Console, depois vá para a guia Response headers (Cabeçalhos de resposta) na página Policies (Políticas) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/policies/responseHeaders>.
2. Selecione Create response headers policy (Criar política de cabeçalhos de resposta).
3. No formulário Create response headers policy (Criar política de cabeçalhos de resposta), faça o seguinte:
 - a. No painel Details (Detalhes), insira um Name (Nome) para a política de cabeçalhos de resposta e (opcionalmente) uma Description (Descrição) que explique para que serve a política.
 - b. No painel Cross-origin resource sharing (CORS) (Compartilhamento de recursos entre origens, CORS), escolha a chave Configure CORS (Configurar CORS) e configure todos os cabeçalhos do CORS que você deseja adicionar à política. Se você quiser que os cabeçalhos configurados substituam os cabeçalhos que o CloudFront recebe da origem, marque a caixa de seleção Origin override (Substituir origem).

Para obter mais informações sobre as configurações dos cabeçalhos do CORS, consulte [the section called “Cabeçalhos de CORS”](#).

- c. No painel Security headers (Cabeçalhos de segurança), escolha a chave e configure cada um dos cabeçalhos de segurança que deseja adicionar à política.

Para obter mais informações sobre as configurações dos cabeçalhos de segurança, consulte [the section called “Cabeçalhos de segurança”](#).

- d. No painel Custom headers (Cabeçalhos personalizados), adicione quaisquer cabeçalhos personalizados que você deseja incluir na política.

Para obter mais informações sobre as configurações dos cabeçalhos personalizados, consulte [the section called “Cabeçalhos personalizados”](#).

- e. No painel Remove headers (Remover cabeçalhos), adicione os nomes de todos os cabeçalhos que deseja que o CloudFront remova da resposta da origem e não inclua na resposta que o CloudFront enviará aos visualizadores.

Para obter mais informações sobre as configurações de remoção de cabeçalhos, consulte [the section called “Remover cabeçalhos”](#).

- f. No painel Server-Timing header (Cabeçalho Server-Timing), escolha a opção Enable (Habilitar) e insira uma taxa de amostragem (um número entre 0 e 100, ambos incluídos).

Para obter mais informações sobre o cabeçalho Server-Timing, consulte [the section called “Cabeçalho de temporização do servidor”](#).

4. Escolha Create (Criar) para criar a política.

Depois de criar uma política de cabeçalhos de resposta, você pode anexá-la a um comportamento de cache em uma distribuição do CloudFront.

Para anexar uma política de cabeçalhos de resposta a uma distribuição existente (console)

1. Abra a página Distributions (Distribuições) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/distributions>.
2. Escolha a distribuição a ser atualizada e escolha a guia Behaviors (Comportamentos).
3. Selecione o comportamento de cache a ser atualizado e escolha Edit (Editar).

Ou, para criar um novo comportamento de cache, escolha Create behavior (Criar comportamento).

4. Em Response headers policy (Políticas de cabeçalhos de resposta), escolha a política a ser adicionada ao comportamento do cache.

5. Escolha Save changes (Salvar alterações) para atualizar o comportamento do cache. Se você estiver criando um comportamento de cache, escolha Create behavior (Criar comportamento).

Para anexar uma política de cabeçalhos de resposta a uma nova distribuição (console)

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha Create distribution (Criar distribuição).
3. Em Response headers policy (Políticas de cabeçalhos de resposta), escolha a política a ser adicionada ao comportamento do cache.
4. Escolha as outras configurações para sua distribuição. Para ter mais informações, consulte [the section called “Configurações de distribuição”](#).
5. Escolha Create distribution (Criar distribuição) para criar a distribuição.

AWS CloudFormation

Para criar uma política de cabeçalhos de resposta com AWS CloudFormation, use o tipo de recurso `AWS::CloudFront::ResponseHeadersPolicy`. O exemplo a seguir mostra a sintaxe do modelo de AWS CloudFormation, no formato YAML, para criar uma política de cabeçalhos de resposta.

```
Type: AWS::CloudFront::ResponseHeadersPolicy
Properties:
  ResponseHeadersPolicyConfig:
    Name: EXAMPLE-Response-Headers-Policy
    Comment: Example response headers policy for the documentation
  CorsConfig:
    AccessControlAllowCredentials: false
    AccessControlAllowHeaders:
      Items:
        - '*'
    AccessControlAllowMethods:
      Items:
        - GET
        - OPTIONS
    AccessControlAllowOrigins:
      Items:
        - https://example.com
        - https://docs.example.com
```

```
AccessControlExposeHeaders:
  Items:
    - '*'
AccessControlMaxAgeSec: 600
OriginOverride: false
CustomHeadersConfig:
  Items:
    - Header: Example-Custom-Header-1
      Value: value-1
      Override: true
    - Header: Example-Custom-Header-2
      Value: value-2
      Override: true
SecurityHeadersConfig:
  ContentSecurityPolicy:
    ContentSecurityPolicy: default-src 'none'; img-src 'self'; script-src
'self'; style-src 'self'; object-src 'none'; frame-ancestors 'none'
    Override: false
  ContentTypeOptions: # You don't need to specify a value for 'X-Content-Type-
Options'.
                        # Simply including it in the template sets its value to
'nosniff'.
    Override: false
  FrameOptions:
    FrameOption: DENY
    Override: false
  ReferrerPolicy:
    ReferrerPolicy: same-origin
    Override: false
  StrictTransportSecurity:
    AccessControlMaxAgeSec: 63072000
    IncludeSubdomains: true
    Preload: true
    Override: false
  XSSProtection:
    ModeBlock: true # You can set ModeBlock to 'true' OR set a value for
ReportUri, but not both
    Protection: true
    Override: false
ServerTimingHeadersConfig:
  Enabled: true
  SamplingRate: 50
RemoveHeadersConfig:
  Items:
```

- Header: Vary
- Header: X-Powered-By

Para obter mais informações, consulte [AWS::CloudFront::ResponseHeadersPolicy](#) no Guia do usuário do AWS CloudFormation.

CLI

Para criar uma política de cabeçalhos de resposta com a AWS Command Line Interface (AWS CLI), use o comando `aws cloudfront create-response-headers-policy`. É possível usar um arquivo de entrada para fornecer os parâmetros de entrada do comando, em vez de especificar cada parâmetro individual como entrada na linha de comando.

Como criar uma política de cabeçalhos de resposta (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `response-headers-policy.yaml`. Esse arquivo contém todos os parâmetros de entrada para o comando `create-response-headers-policy`.

```
aws cloudfront create-response-headers-policy --generate-cli-skeleton yaml-input  
> response-headers-policy.yaml
```

2. Abra o arquivo `response-headers-policy.yaml` que você acabou de criar. Edite o arquivo para especificar um nome de política e a configuração desejada de política de cabeçalhos de resposta, depois salve o arquivo.

Para obter mais informações sobre as configurações de política de cabeçalhos de resposta, consulte [the section called “Noções básicas das políticas de cabeçalhos de resposta”](#).

3. Use o comando a seguir para criar a política de cabeçalhos de resposta. A política que você cria usa os parâmetros de entrada do arquivo `response-headers-policy.yaml`.

```
aws cloudfront create-response-headers-policy --cli-input-yaml file://response-headers-policy.yaml
```

Anote o valor do Id na saída do comando. Esse é o ID da política de cabeçalhos de resposta. Você precisa dele para anexar a política ao comportamento de cache de uma distribuição do CloudFront.

Como anexar uma política de cabeçalhos de resposta a uma distribuição existente (CLI com arquivo de entrada)

1. Use o comando a seguir para salvar a configuração da distribuição do CloudFront que você deseja atualizar. Substitua *distribution_ID* pelo ID da distribuição.

```
aws cloudfront get-distribution-config --id distribution_ID --output yaml >
dist-config.yaml
```

2. Abra o arquivo chamado `dist-config.yaml` que você acabou de criar. Edite o arquivo, fazendo as seguintes alterações no comportamento de cache para que ele use a política de cabeçalhos de resposta.
 - No comportamento de cache, adicione um campo chamado `ResponseHeadersPolicyId`. Para o valor do campo, use o ID da política de cabeçalhos de resposta que você anotou depois de criar a política.
 - Renomeie o campo `Etag` para `IfMatch`, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

3. Use o comando a seguir para atualizar a distribuição para utilizar a política de cabeçalhos de resposta. Substitua *distribution_ID* pelo ID da distribuição.

```
aws cloudfront update-distribution --id distribution_ID --cli-input-yaml file://
dist-config.yaml
```

Como anexar uma política de cabeçalhos de resposta a uma nova distribuição (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `distribution.yaml`. Esse arquivo contém todos os parâmetros de entrada para o comando `create-distribution`.

```
aws cloudfront create-distribution --generate-cli-skeleton yaml-input >
distribution.yaml
```

2. Abra o arquivo `distribution.yaml` que você acabou de criar. No comportamento de cache padrão, no campo `ResponseHeadersPolicyId`, insira o ID da política de cabeçalhos de resposta que você anotou após criar a política. Continue editando o arquivo para especificar as configurações de distribuição desejadas e salve o arquivo ao concluir.

Para obter mais informações sobre as configurações de distribuição, consulte [Referência de configurações da distribuição](#).

3. Use o seguinte comando para criar a distribuição usando parâmetros de entrada do arquivo `distribution.yaml`.

```
aws cloudfront create-distribution --cli-input-yaml file://distribution.yaml
```

API

Para criar uma política de cabeçalhos de resposta com a API do CloudFront, use [CreateResponse HeadersPolicy](#). Para obter mais informações sobre os campos especificados nessa chamada de API, consulte [the section called “Noções básicas das políticas de cabeçalhos de resposta”](#) e a documentação de referência de API do seu SDK da AWS ou de outro cliente de API.

Depois de criar uma política de cabeçalhos de resposta, é possível anexá-la a um comportamento de cache, usando uma das seguintes chamadas de API:

- Para anexá-la a um comportamento de cache em uma distribuição existente, use [UpdateDistribution](#).
- Para anexá-la a um comportamento de cache em uma nova distribuição, use [CreateDistribution](#).

Para as duas chamadas de API, forneça o ID da política de cabeçalhos de resposta no campo `ResponseHeadersPolicyId`, dentro de um comportamento de cache. Para mais informações sobre os outros campos especificados nessas chamadas de API, consulte [Referência de configurações da distribuição](#) e a documentação de referência da API do AWS SDK ou de outro cliente de API.

Usar políticas de cabeçalhos de resposta gerenciadas

Com uma política de cabeçalhos de resposta do CloudFront, você pode especificar os cabeçalhos de HTTP que o Amazon CloudFront remove ou adiciona às respostas enviadas aos visualizadores. Para obter mais informações sobre políticas de cabeçalhos de respostas e os motivos para usá-las, consulte [Adicionar ou remover cabeçalhos de resposta com uma política](#).

O CloudFront fornece políticas de cabeçalhos de resposta gerenciadas que você pode anexar a comportamentos de cache nas distribuições do CloudFront. Com uma política de cabeçalhos de resposta gerenciada, você não precisa gravar ou manter sua própria política. As políticas gerenciadas contêm conjuntos de cabeçalhos de resposta de HTTP para casos de uso comuns.

Para usar uma política de cabeçalhos de resposta gerenciada, anexe-a a um comportamento de cache em sua distribuição. O processo é o mesmo que o de criação de uma política de cabeçalhos de resposta personalizados. No entanto, em vez de criar uma política, você anexa uma das políticas gerenciadas. Você anexa a política por nome (com o console) ou por ID (com o AWS CloudFormation, a AWS CLI ou os SDKs da AWS). Os nomes e IDs são listados na seção a seguir.

Para ter mais informações, consulte [the section called “Criar políticas de cabeçalhos de resposta”](#).

Os tópicos a seguir descrevem as políticas de cabeçalhos de resposta gerenciadas que você pode usar.

Tópicos

- [CORS-and-SecurityHeadersPolicy](#)
- [CORS-With-Preflight](#)
- [CORS-with-preflight-and-SecurityHeadersPolicy](#)
- [SecurityHeadersPolicy](#)
- [SimpleCORS](#)

CORS-and-SecurityHeadersPolicy

[Visualizar essa política no console do CloudFront](#)

Use essa política gerenciada para permitir solicitações simples de CORS de qualquer origem. Essa política também adiciona um conjunto de cabeçalhos de segurança a todas as respostas que o CloudFront envia aos visualizadores. Esta política combina as políticas [the section called “SimpleCORS”](#) e [the section called “SecurityHeadersPolicy”](#) em uma.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

e61eb60c-9c35-4d20-a928-2b84e02af89c

Configurações de política

	Nome do cabeçalho	Valor de cabeçalho	Substituir origem?
Cabeçalhos de CORS:	Access-Control-Allow-Origin	*	Não
Cabeçalhos de segurança:	Referrer-Policy	strict-origin-when-cross-origin	Não
	Strict-Transport-Security	max-age=31536000	Não
	X-Content-Type-Options	nosniff	Sim
	X-Frame-Options	SAMEORIGIN	Não
	X-XSS-Protection	1; mode=block	Não

CORS-With-Preflight

[Visualizar essa política no console do CloudFront](#)

Use essa política gerenciada para permitir solicitações de CORS de qualquer origem, incluindo solicitações de comprovação. Para solicitações de comprovação (usando o método OPTIONS de HTTP), o CloudFront adiciona todos os três cabeçalhos a seguir à resposta. Para solicitações simples de CORS, o CloudFront adiciona apenas o cabeçalho Access-Control-Allow-Origin.

Se a resposta que o CloudFront receber da origem incluir algum desses cabeçalhos, o CloudFront usará o cabeçalho recebido (e seu valor) em sua resposta ao visualizador. O CloudFront não usa o cabeçalho nessa política.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

5cc3b908-e619-4b99-88e5-2cf7f45965bd

Configurações de política

	Nome do cabeçalho	Valor de cabeçalho	Substituir origem?
Cabeçalhos de CORS:	Access-Control-Allow-Methods	DELETE, GET, HEAD, OPTIONS, PATCH, POST, PUT	Não
	Access-Control-Allow-Origin	*	
	Access-Control-Expose-Headers	*	

CORS-with-preflight-and-SecurityHeadersPolicy

[Visualizar essa política no console do CloudFront](#)

Use essa política gerenciada para permitir solicitações de CORS de qualquer origem. Isso inclui solicitações de comprovação. Essa política também adiciona um conjunto de cabeçalhos de segurança a todas as respostas que o CloudFront envia aos visualizadores. Esta política combina as políticas [the section called “CORS-With-Preflight”](#) e [the section called “SecurityHeadersPolicy”](#) em uma.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

eaab4381-ed33-4a86-88ca-d9558dc6cd63

Configurações de política

	Nome do cabeçalho	Valor de cabeçalho	Substituir origem?
Cabeçalhos de CORS:	Access-Control-Allow-Methods	DELETE, GET, HEAD, OPTIONS, PATCH, POST, PUT	Não
	Access-Control-Allow-Origin	*	

	Nome do cabeçalho	Valor de cabeçalho	Substituir origem?
	Access-Control-Expose-Headers	*	
Cabeçalhos de segurança:	Referrer-Policy	strict-origin-when-cross-origin	Não
	Strict-Transport-Security	max-age=31536000	Não
	X-Content-Type-Options	nosniff	Sim
	X-Frame-Options	SAMEORIGIN	Não
	X-XSS-Protection	1; mode=block	Não

SecurityHeadersPolicy

[Visualizar essa política no console do CloudFront](#)

Use essa política gerenciada para adicionar um conjunto de cabeçalhos de segurança a todas as respostas enviadas pelo CloudFront aos visualizadores. Para obter mais informações sobre esses cabeçalhos de segurança, consulte as [diretrizes de segurança na Web do Mozilla](#).

Com essa política de cabeçalhos de resposta, o CloudFront adiciona X-Content-Type-Options: nosniff a todas as respostas. Esse é o caso quando a resposta que o CloudFront recebeu da origem incluiu esse cabeçalho e quando não. Para todos os outros cabeçalhos nessa política, se a resposta que o CloudFront recebeu da origem incluir o cabeçalho, o CloudFront usará o cabeçalho recebido (e seu valor) em sua resposta ao visualizador. Ele não usa o cabeçalho nessa política.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

67f7725c-6f97-4210-82d7-5512b31e9d03

Configurações de política

	Nome do cabeçalho	Valor de cabeçalho	Substituir origem?
Cabeçalhos de segurança:	Referrer-Policy	strict-origin-when-cross-origin	Não
	Strict-Transport-Security	max-age=31536000	Não
	X-Content-Type-Options	nosniff	Sim
	X-Frame-Options	SAMEORIGIN	Não
	X-XSS-Protection	1; mode=block	Não

SimpleCORS

[Visualizar essa política no console do CloudFront](#)

Use essa política gerenciada para permitir [solicitações simples de CORS](#) de qualquer origem. Com essa política, o CloudFront adiciona o cabeçalho `Access-Control-Allow-Origin: *` a todas as respostas para solicitações simples de CORS.

Se a resposta que o CloudFront receber da origem incluir o cabeçalho `Access-Control-Allow-Origin`, o CloudFront usará esse cabeçalho (e seu valor) em sua resposta ao visualizador. O CloudFront não usa o cabeçalho nessa política.

Ao usar o AWS CloudFormation, a AWS CLI ou a API do CloudFront, o ID dessa política é:

60669652-455b-4ae9-85a4-c4c02393f86c

Configurações de política

	Nome do cabeçalho	Valor de cabeçalho	Substituir origem?
Cabeçalhos de CORS:	Access-Control-Allow-Origin	*	Não

Comportamento de solicitações e respostas

Os tópicos a seguir descrevem como o CloudFront lida com solicitações e respostas.

Você pode saber mais sobre como o CloudFront interage com o Amazon S3 ou com origens personalizadas, lida com vários métodos e cabeçalhos HTTP, processa códigos de status e gerencia o armazenamento em cache e as respostas de erro.

Tópicos

- [Como o CloudFront processa solicitações HTTP e HTTPS](#)
- [Comportamento de solicitações e respostas para origens do Amazon S3](#)
- [Comportamento de solicitações e respostas para origens personalizadas](#)
- [Comportamento de solicitações e respostas para grupos de origens](#)
- [Adicionar cabeçalhos personalizados às solicitações de origem](#)
- [Como o CloudFront processa solicitações parciais de um objeto \(Range GETs\)](#)
- [Como o CloudFront processa códigos de status HTTP 3xx da origem](#)
- [Como o CloudFront processa códigos de status HTTP 4xx e 5xx da origem](#)
- [Gerar respostas a erros personalizadas](#)

Como o CloudFront processa solicitações HTTP e HTTPS

Para origens do Amazon S3, o CloudFront aceita solicitações nos protocolos HTTP e HTTPS para objetos em uma distribuição do CloudFront por padrão. O CloudFront encaminha as solicitações ao bucket do Amazon S3 usando o mesmo protocolo usado para fazer as solicitações.

Para origens personalizadas, ao criar a distribuição, é possível especificar como o CloudFront acessa a origem: apenas HTTP ou correspondência com o protocolo usado pelo visualizador. Para mais informações sobre como o CloudFront lida com solicitações HTTP e HTTPS para origens personalizadas, consulte [Protocolos](#).

Para informações sobre como restringir sua distribuição, de modo que os usuários finais só possam acessar objetos usando HTTPS, consulte [Usar HTTPS com o CloudFront](#).

Note

A cobrança de solicitações HTTPS é superior a de solicitações HTTP. Para ter mais informações sobre as taxas de faturamento, acesse [Preço do CloudFront](#).

Comportamento de solicitações e respostas para origens do Amazon S3

Para entender como o CloudFront processa solicitações e respostas ao usar o Amazon S3 como origem, consulte as seguintes seções:

Tópicos

- [Como o CloudFront processa e encaminha solicitações à sua origem do Amazon S3](#)
- [Como o CloudFront processa as respostas da origem do Amazon S3](#)

Como o CloudFront processa e encaminha solicitações à sua origem do Amazon S3

Saiba como o CloudFront processa solicitações do visualizador e as encaminha para a origem do Amazon S3.

Sumário

- [Duração do armazenamento em cache e TTL mínimo](#)
- [Endereços IP do cliente](#)
- [Solicitações GET condicionais](#)
- [Cookies](#)
- [Compartilhamento de recursos de origem cruzada \(CORS\)](#)
- [Solicitações GET que incluem um corpo](#)
- [Métodos HTTP](#)
- [Cabeçalhos de solicitação HTTP removidos ou atualizados pelo CloudFront](#)
- [Tamanho máximo de uma solicitação e de um URL](#)
- [OCSP Stapling](#)

- [Protocolos](#)
- [Strings de consulta](#)
- [Tentativas e tempo limite de conexão da origem](#)
- [Tempo limite de resposta da origem](#)
- [Solicitações simultâneas para o mesmo objeto \(recolhimento de solicitações\)](#)

Duração do armazenamento em cache e TTL mínimo

Para controlar por quanto tempo os objetos permanecem em um cache do CloudFront antes que o CloudFront encaminhe outra solicitação para a sua origem, você pode:

- Configurar sua origem para adicionar um campo de cabeçalho `Cache-Control` ou `Expires` a cada objeto.
- Especificar um valor de TTL mínimo nos comportamentos de cache do CloudFront.
- Usar o valor padrão de 24 horas.

Para obter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Endereços IP do cliente

Se um visualizador enviar uma solicitação ao CloudFront e não incluir um cabeçalho de solicitação `X-Forwarded-For`, o CloudFront receberá o endereço IP do visualizador na conexão TCP, adicionará um cabeçalho `X-Forwarded-For` que inclui o endereço IP e encaminhará a solicitação à origem. Por exemplo, se o CloudFront obter o endereço IP `192.0.2.2` da conexão TCP, ele encaminhará o seguinte cabeçalho à origem:

```
X-Forwarded-For: 192.0.2.2
```

Se um visualizador enviar uma solicitação ao CloudFront e incluir um cabeçalho de solicitação `X-Forwarded-For`, o CloudFront obterá o endereço IP do visualizador na conexão TCP, incluirá ele no fim do cabeçalho `X-Forwarded-For` e encaminhará a solicitação à origem. Por exemplo, se a solicitação do visualizador incluir `X-Forwarded-For: 192.0.2.4, 192.0.2.3` e o CloudFront obtiver o endereço IP `192.0.2.2` da conexão TCP, ele encaminhará o seguinte cabeçalho à origem:

```
X-Forwarded-For: 192.0.2.4, 192.0.2.3, 192.0.2.2
```

Note

O cabeçalho `X-Forwarded-For` contém endereços IPv4 (como 192.0.2.44) e IPv6 (como 2001:0db8:85a3::8a2e:0370:7334).

Solicitações GET condicionais

Ao receber uma solicitação de um objeto que expirou de um cache de borda, o CloudFront encaminha a solicitação à origem do Amazon S3 a fim de receber a versão mais recente do objeto ou a confirmação do Amazon S3 de que o cache de borda do CloudFront já tem a versão mais recente. Ao enviar originalmente o objeto ao CloudFront, o Amazon S3 incluiu os valores `ETag` e `LastModified` na resposta. Na nova solicitação encaminhada pelo CloudFront ao Amazon S3, o CloudFront adiciona um destes cabeçalhos:

- Um cabeçalho `If-Match` ou `If-None-Match` com o valor `ETag` da versão expirada do objeto.
- Um cabeçalho `If-Modified-Since` com o valor `LastModified` da versão expirada do objeto.

O Amazon S3 usa essas informações para determinar se o objeto foi atualizado e, portanto, se deve retornar todo o objeto ao CloudFront ou apenas a um código de status HTTP 304 (não modificado).

Cookies

O Amazon S3 não processa cookies. Se você configurar um comportamento de cache para encaminhar cookies a uma origem do Amazon S3, o CloudFront encaminhará os cookies, mas o Amazon S3 os ignorará. Todas as solicitações futuras do mesmo objeto, independentemente se você variar o cookie ou não, serão fornecidas do objeto existente no cache.

Compartilhamento de recursos de origem cruzada (CORS)

Se quiser que o CloudFront respeite as configurações de compartilhamento de recursos entre origens do Amazon S3, configure o CloudFront para encaminhar os cabeçalhos selecionados ao Amazon S3. Para obter mais informações, consulte [Conteúdo em cache com base nos cabeçalhos de solicitação](#).

Solicitações GET que incluem um corpo

Se a solicitação GET do visualizador incluir um corpo, o CloudFront retornará o código de status HTTP 403 (Proibido).

Métodos HTTP

Se você configurar o CloudFront para processar todos os métodos HTTP compatíveis, ele aceitará as seguintes solicitações de visualizadores e as encaminhará à origem do Amazon S3:

- DELETE
- GET
- HEAD
- OPTIONS
- PATCH
- POST
- PUT

O CloudFront sempre armazena respostas às solicitações GET e HEAD em cache. Também é possível configurar o CloudFront para armazenar respostas a solicitações OPTIONS em cache. O CloudFront não armazena em cache respostas a solicitações que usem outros métodos.

Se você quiser usar carregamentos fragmentados para adicionar objetos a um bucket do Amazon S3, será necessário adicionar um controle de acesso à origem (OAC) do CloudFront à distribuição e conceder ao OAC as permissões necessárias. Para ter mais informações, consulte [the section called “Restringir o acesso a uma origem do Amazon Simple Storage Service”](#).

Important

Se você configurar o CloudFront para aceitar e encaminhar ao Amazon S3 todos os métodos HTTP compatíveis com o CloudFront, será necessário criar um AOC do CloudFront para restringir o acesso ao conteúdo do Amazon S3 e conceder ao OAC as permissões necessárias. Por exemplo, se você configurar o CloudFront para aceitar e encaminhar esses métodos porque deseja usar o método PUT, será necessário configurar as políticas do bucket do Amazon S3 para lidar com solicitações DELETE de forma apropriada para que os visualizadores não possam excluir recursos que você não deseja que eles excluam. Para ter mais informações, consulte [the section called “Restringir o acesso a uma origem do Amazon Simple Storage Service”](#).

Para obter informações sobre as operações compatíveis com o Amazon S3, consulte a [Documentação do Amazon S3](#).

Cabeçalhos de solicitação HTTP removidos ou atualizados pelo CloudFront

O CloudFront remove ou atualiza alguns cabeçalhos antes de encaminhar solicitações à origem do Amazon S3. Para a maioria dos cabeçalhos, esse comportamento é o mesmo que para origens personalizadas. Para obter uma lista completa dos cabeçalhos de solicitação HTTP e a maneira como o CloudFront os processa, consulte [Cabeçalhos de solicitação HTTP e comportamento do CloudFront \(origens do Amazon S3 e personalizadas\)](#).

Tamanho máximo de uma solicitação e de um URL

O tamanho máximo de uma solicitação, com o caminho, a query string (se houver) e os cabeçalhos, é de 20.480 bytes.

O CloudFront cria um URL com base na solicitação. O tamanho máximo do URL é de 8.192 bytes.

Se uma solicitação ou um URL ultrapassar o tamanho máximo, o CloudFront exibirá o código de status HTTP 413 (Entidade de solicitação muito grande) ao visualizador e, depois, encerrará a conexão TCP com ele.

OCSP Stapling

Quando um visualizador envia uma solicitação HTTPS para um objeto, tanto ele quanto o CloudFront devem confirmar com a autoridade de certificação (CA) se o certificado SSL do domínio não foi revogado. O OCSP Stapling acelera a validação do certificado permitindo que o CloudFront valide o certificado e armazene as respostas da CA em cache, a fim de que o cliente não precise validar o certificado diretamente com a CA.

A melhoria da performance do OCSP Stapling é mais acentuada quando o CloudFront recebe um grande número de solicitações HTTPS para objetos no mesmo domínio. Cada servidor em um ponto de presença do CloudFront deve enviar uma solicitação de validação separada. Quando o CloudFront recebe um grande número de solicitações HTTPS para o mesmo domínio, cada servidor no local da borda logo recebe uma resposta da CA que pode "grampear" em um pacote no handshake SSL. Quando o visualizador está convencido de que o certificado é válido, o CloudFront pode servir o objeto solicitado. Caso sua distribuição não tenha muito tráfego em um ponto de presença do CloudFront, é provável que novas solicitações sejam direcionadas para um servidor que ainda não validou o certificado com a CA. Nesse caso, o visualizador executa separadamente a etapa de validação, e o servidor do CloudFront fornece o objeto. Esse servidor do CloudFront também envia uma solicitação de validação para a CA para que, na próxima vez que receber uma solicitação com o mesmo nome de domínio, tenha uma resposta de validação da CA.

Protocolos

O CloudFront encaminha solicitações HTTP ou HTTPS ao servidor de origem com base no protocolo da solicitação do visualizador: HTTP ou HTTPS.

Important

Se o bucket do Amazon S3 estiver configurado como um endpoint de site, não será possível configurar o CloudFront para usar HTTPS para comunicação com a origem, pois o Amazon S3 não é compatível com conexões HTTPS nessa configuração.

Strings de consulta

É possível configurar se o CloudFront encaminha parâmetros de string de consulta para a sua origem no Amazon S3. Para obter mais informações, consulte [Conteúdo em cache com base em parâmetros de string de consulta](#).

Tentativas e tempo limite de conexão da origem

Tempo limite de conexão da origem é o número de segundos que o CloudFront aguarda ao tentar estabelecer uma conexão com a origem.

Tentativas de conexão da origem é o número de vezes que o CloudFront tenta se conectar à origem.

Juntas, essas configurações determinam por quanto tempo o CloudFront tenta se conectar à origem antes de fazer o failover para a origem secundária, no caso de um grupo de origens, ou retornar uma resposta de erro ao visualizador. Por padrão, o CloudFront aguarda até 30 segundos (3 tentativas de 10 segundos cada) antes de tentar se conectar à origem secundária ou retornar uma resposta de erro. Esse tempo pode ser reduzido especificando um tempo limite de conexão mais curto, menos tentativas ou ambos.

Para obter mais informações, consulte [Controlar tempos limite e tentativas da origem](#).

Tempo limite de resposta da origem

O tempo limite de resposta da origem, também conhecido como tempo limite de leitura da origem ou tempo limite de solicitação da origem, aplica-se a estes dois valores:

- A quantidade de tempo, em segundos, que o CloudFront aguarda uma resposta após o encaminhamento de uma solicitação à origem.

- A quantidade de tempo, em segundos, que o CloudFront aguarda após o recebimento de um pacote de resposta da origem e antes do recebimento do próximo pacote.

O comportamento do CloudFront depende do método HTTP da solicitação do visualizador:

- Solicitações GET e HEAD: se a origem não responder dentro de 30 segundos ou parar de responder por 30 segundos, o CloudFront descartará a conexão. Se o número especificado de [tentativas de conexão da origem](#) for maior que 1, o CloudFront tentará obter uma resposta completa novamente. O CloudFront tenta até 3 vezes, conforme determinado pelo valor da configuração de tentativas de conexão da origem. Se a origem não responder durante a terceira tentativa, o CloudFront não tentará novamente enquanto não receber outra solicitação de conteúdo na mesma origem.
- Solicitações DELETE, OPTIONS, PATCH, PUT e POST: se a origem não responder em 30 segundos, o CloudFront interromperá a conexão e não tentará entrar em contato com a origem novamente. O cliente pode reenviar a solicitação, se necessário.

Não é possível alterar o tempo limite de resposta para uma origem do Amazon S3 (um bucket do S3 que não esteja configurado com hospedagem de site estático).

Solicitações simultâneas para o mesmo objeto (recolhimento de solicitações)

Quando um local da borda do CloudFront recebe uma solicitação de um objeto, e o objeto não está no cache ou o objeto em cache expirou, o CloudFront envia imediatamente a solicitação para a origem. No entanto, se houver solicitações simultâneas para o mesmo objeto, ou seja, se solicitações adicionais para o mesmo objeto (com a mesma chave de cache) chegarem ao local da borda antes de o CloudFront receber a resposta à primeira solicitação, o CloudFront fará uma pausa antes de encaminhar solicitações adicionais à origem. Essa breve pausa ajuda a reduzir a carga na origem. O CloudFront envia a resposta da solicitação original a todas as solicitações recebidas enquanto estava em pausa. Isso é chamado de recolhimento de solicitações. Nos logs do CloudFront, a primeira solicitação é identificada como Miss no campo `x-edge-result-type` e as solicitações recolhidas são identificadas como Hit. Para mais informações sobre os logs do CloudFront, consulte [the section called “Registro em log do CloudFront e de funções de borda”](#).

O CloudFront apenas recolhe solicitações que compartilham uma [chave de cache](#). Se as solicitações adicionais não compartilharem a mesma chave de cache porque, por exemplo, você configurou o CloudFront para armazenar em cache com base nas strings de consulta, nos cookies ou nos

cabeçalhos da solicitação, o CloudFront encaminhará todas as solicitações com uma chave de cache exclusiva à origem.

Se quiser evitar o recolhimento de todas as solicitações, será possível usar a política de cache gerenciada `CachingDisabled`, que também impede o armazenamento em cache. Para ter mais informações, consulte [Usar políticas de cache gerenciadas](#).

Se você quiser evitar o recolhimento das solicitações para objetos específicos, defina a TTL mínima para o comportamento de cache como 0 e configure a origem para enviar `Cache-Control: private`, `Cache-Control: no-store`, `Cache-Control: no-cache`, `Cache-Control: max-age=0` ou `Cache-Control: s-maxage=0`. Essas configurações vão aumentar a carga na origem e introduzir latência adicional para as solicitações simultâneas que são pausadas enquanto o CloudFront aguarda a resposta à primeira solicitação.

Important

Atualmente, o CloudFront não comporta recolher solicitações caso você habilite o encaminhamento de cookies na [política de cache](#), na [política de solicitação de origem](#) ou nas configurações de cache herdadas.

Como o CloudFront processa as respostas da origem do Amazon S3

Saiba como o CloudFront processa as respostas da origem do Amazon S3.

Sumário

- [Solicitações canceladas](#)
- [Cabeçalhos de resposta HTTP removidos ou atualizados pelo CloudFront](#)
- [Tamanho máximo do arquivo armazenável em cache](#)
- [Redirecionamentos](#)

Solicitações canceladas

Se o objeto não estiver no cache de ponto de presença e o visualizador encerrar a sessão (por exemplo, fechar o navegador) depois de o CloudFront obter o objeto da origem, mas antes de conseguir fornecer o objeto solicitado, ele não armazenará o objeto em cache no ponto de presença.

Cabeçalhos de resposta HTTP removidos ou atualizados pelo CloudFront

O CloudFront remove ou atualiza os seguintes campos de cabeçalho antes de encaminhar a resposta da origem do Amazon S3 ao visualizador:

- X-Amz-Id-2
- X-Amz-Request-Id
- Set-Cookie: se você configurar o CloudFront para encaminhar cookies, ele encaminhará o campo de cabeçalho Set-Cookie aos clientes. Para obter mais informações, consulte [Conteúdo em cache com base em cookies](#).
- Trailer
- Transfer-Encoding: se a origem do Amazon S3 retornar esse campo de cabeçalho, o CloudFront definirá o valor de chunked antes de retornar a resposta ao visualizador.
- Upgrade
- Via: o CloudFront define o seguinte valor na resposta para o visualizador:

Via: *versão HTTP string alfanumérica*.cloudfront.net (CloudFront)

Por exemplo, o valor é semelhante ao seguinte:

Via: 1.1 1026589cc7887e7a0dc7827b4example.cloudfront.net (CloudFront)

Tamanho máximo do arquivo armazenável em cache

O tamanho máximo do corpo de uma resposta salva pelo CloudFront no cache é de 50 GB. Isso inclui respostas de transferência em partes que não especificam o valor de cabeçalho Content-Length.

É possível usar o CloudFront para armazenar em cache um objeto maior que isso usando solicitações de intervalo para solicitar os objetos em partes que sejam cada uma de 50 GB ou menores. O CloudFront armazena essas partes em cache porque cada uma delas é de 50 GB ou menor. Depois que o visualizador recuperar todas as partes do objeto, ele poderá reconstruir o objeto original, maior. Para ter mais informações, consulte [Usar solicitações de intervalo para armazenar objetos grandes em cache](#).

Redirecionamentos

É possível configurar um bucket do Amazon S3 para redirecionar todas as solicitações para outro nome de host, que pode ser outro bucket do Amazon S3 ou um servidor HTTP. Se você configurar um bucket para redirecionar todas as solicitações e se ele for a origem de uma distribuição do CloudFront, recomendamos configurá-lo para redirecionar todas as solicitações para uma distribuição do CloudFront usando o nome de domínio da distribuição (por exemplo, d111111abcdef8.cloudfront.net) ou um nome de domínio alternativo (CNAME) associado a uma distribuição (por exemplo, example.com). Caso contrário, as solicitações do visualizador ignorarão o CloudFront, e os objetos serão fornecidos diretamente da nova origem.

Note

Se você redirecionar as solicitações para um nome de domínio alternativo, deverá também atualizar o serviço de DNS do seu domínio adicionando um registro CNAME. Para obter mais informações, consulte [Usar URLs personalizados adicionando nomes de domínio alternativos \(CNAMEs\)](#).

Veja o que acontece quando você configura um bucket para redirecionar todas as solicitações:

1. Um visualizador (por exemplo, um navegador) solicita um objeto do CloudFront.
2. O CloudFront encaminha a solicitação para o bucket do Amazon S3 que é a origem da distribuição.
3. O Amazon S3 retorna um código de status HTTP 301 (Movido permanentemente) e o novo local.
4. O CloudFront armazena o código de status de redirecionamento e o novo local e retorna os valores ao visualizador. O CloudFront não segue o redirecionamento para obter o objeto do novo local.
5. O visualizador envia outra solicitação do objeto, mas desta vez especifica o novo local obtido do CloudFront:
 - Se o bucket do Amazon S3 estiver redirecionando todas as solicitações para uma distribuição do CloudFront, usando o nome de domínio da distribuição ou um nome de domínio alternativo, o CloudFront solicitará o objeto do bucket do Amazon S3 ou do servidor HTTP no novo local. Quando o novo local retornar o objeto, o CloudFront o retorna ao visualizador e o armazenará em cache em um ponto de presença.

- Se o bucket do Amazon S3 estiver redirecionando solicitações para outro local, a segunda solicitação ignorará o CloudFront. O bucket do Amazon S3 ou servidor HTTP no novo local retorna o objeto diretamente para o visualizador, para que ele nunca seja armazenado em cache em um cache de ponto de presença do CloudFront.

Comportamento de solicitações e respostas para origens personalizadas

Para entender como o CloudFront processa solicitações e respostas ao usar origens personalizadas, consulte as seguintes seções:

Tópicos

- [Como o CloudFront processa e encaminha solicitações para sua origem personalizada](#)
- [Como o CloudFront processa respostas da sua origem personalizada](#)

Como o CloudFront processa e encaminha solicitações para sua origem personalizada

Saiba como o CloudFront processa solicitações do visualizador e as encaminha para a origem personalizada.

Sumário

- [Autenticação](#)
- [Duração do armazenamento em cache e TTL mínimo](#)
- [Endereços IP do cliente](#)
- [Autenticação SSL no lado do cliente](#)
- [Compactação](#)
- [Solicitações condicionais](#)
- [Cookies](#)
- [Compartilhamento de recursos de origem cruzada \(CORS\)](#)
- [Criptografia](#)
- [Solicitações GET que incluem um corpo](#)
- [Métodos HTTP](#)

- [Cabeçalhos de solicitação HTTP e comportamento do CloudFront \(origens do Amazon S3 e personalizadas\)](#)
- [Versão HTTP](#)
- [Tamanho máximo de uma solicitação e de um URL](#)
- [Associação OCSP](#)
- [Conexões persistentes](#)
- [Protocolos](#)
- [Strings de consulta](#)
- [Tentativas e tempo limite de conexão da origem](#)
- [Tempo limite de resposta da origem](#)
- [Solicitações simultâneas para o mesmo objeto \(recolhimento de solicitações\)](#)
- [User-AgentCabeçalho](#)

Autenticação

Se você encaminhar o cabeçalho `Authorization` para sua origem, será possível configurar o servidor de origem para solicitar a autenticação do cliente para os seguintes tipos de solicitações:

- DELETE
- GET
- HEAD
- PATCH
- PUT
- POST

Para solicitações `OPTIONS`, é possível configurar a autenticação do cliente somente se usar as seguintes configurações do CloudFront:

- O CloudFront foi configurado para encaminhar o cabeçalho `Authorization` à origem
- O CloudFront foi configurado para não armazenar a resposta a solicitações `OPTIONS` em cache

Para ter mais informações, consulte [Configurar o CloudFront para encaminhar o cabeçalho `Authorization`](#).

É possível usar HTTP ou HTTPS para encaminhar solicitações para o servidor de origem. Para ter mais informações, consulte [Usar HTTPS com o CloudFront](#).

Duração do armazenamento em cache e TTL mínimo

Para controlar por quanto tempo os objetos permanecem em um cache do CloudFront antes que o CloudFront encaminhe outra solicitação para a sua origem, você pode:

- Configurar sua origem para adicionar um campo de cabeçalho `Cache-Control` ou `Expires` a cada objeto.
- Especificar um valor de TTL mínimo nos comportamentos de cache do CloudFront.
- Usar o valor padrão de 24 horas.

Para obter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Endereços IP do cliente

Se um visualizador enviar uma solicitação ao CloudFront e não incluir um cabeçalho de solicitação `X-Forwarded-For`, o CloudFront obterá o endereço IP do visualizador na conexão TCP, adicionará um cabeçalho `X-Forwarded-For` que inclui o endereço IP e encaminhará a solicitação à origem. Por exemplo, se o CloudFront obter o endereço IP `192.0.2.2` da conexão TCP, ele encaminhará o seguinte cabeçalho à origem:

```
X-Forwarded-For: 192.0.2.2
```

Se um visualizador enviar uma solicitação ao CloudFront e incluir um cabeçalho de solicitação `X-Forwarded-For`, o CloudFront obterá o endereço IP do visualizador na conexão TCP, incluirá ele no fim do cabeçalho `X-Forwarded-For` e encaminhará a solicitação à origem. Por exemplo, se a solicitação do visualizador incluir `X-Forwarded-For: 192.0.2.4,192.0.2.3` e o CloudFront obtiver o endereço IP `192.0.2.2` da conexão TCP, ele encaminhará o seguinte cabeçalho à origem:

```
X-Forwarded-For: 192.0.2.4,192.0.2.3,192.0.2.2
```

Algumas aplicações, como balanceadores de carga (inclusive o Elastic Load Balancing), firewalls de aplicações Web, proxies reversos, sistemas de prevenção de invasão e API Gateway, adicionam o endereço IP do servidor de borda do CloudFront que encaminhou a solicitação no fim do cabeçalho `X-Forwarded-For`. Por exemplo, se o CloudFront incluir `X-Forwarded-For: 192.0.2.2`

em uma solicitação encaminhada ao ELB e o endereço IP do servidor de borda do CloudFront for 192.0.2.199, a solicitação recebida pela instância do EC2 conterá o seguinte cabeçalho:

```
X-Forwarded-For: 192.0.2.2,192.0.2.199
```

Note

O cabeçalho `X-Forwarded-For` contém endereços IPv4 (como 192.0.2.44) e IPv6 (como 2001:0db8:85a3::8a2e:0370:7334).

Observe também que o cabeçalho `X-Forwarded-For` pode ser modificado por cada nó no caminho para o servidor atual (CloudFront). Para obter mais informações, consulte a seção 8.1 do [RFC 7239](#). Também é possível modificar o cabeçalho usando as funções de computação de borda do CloudFront.

Autenticação SSL no lado do cliente

O CloudFront não é compatível com a autenticação do cliente com certificados SSL no lado do cliente. Se uma origem solicitar um certificado no lado do cliente, o CloudFront interromperá a solicitação.

Compactação

Para ter mais informações, consulte [Fornecer arquivos compactados](#).

Solicitações condicionais

Ao receber uma solicitação de um objeto que expirou de um cache de ponto de presença, o CloudFront encaminha a solicitação à origem a fim de obter a versão mais recente do objeto ou a confirmação da origem de que o cache do ponto de presença do CloudFront já tem a versão mais recente. Normalmente, ao enviar o objeto pela última vez ao CloudFront, a origem inclui um valor `ETag` ou `LastModified`, ou os dois, na resposta. Na nova solicitação encaminhada pelo CloudFront à origem, o CloudFront adiciona um destes (ou os dois):

- Um cabeçalho `If-Match` ou `If-None-Match` com o valor `ETag` da versão expirada do objeto.
- Um cabeçalho `If-Modified-Since` com o valor `LastModified` da versão expirada do objeto.

A origem usa essas informações para determinar se o objeto foi atualizado e, portanto, se deve retornar todo o objeto ao CloudFront ou apenas a um código de status HTTP 304 (não modificado).

Note

As solicitações condicionais `If-Modified-Since` e `If-None-Match` não são compatíveis com o CloudFront quando ele estiver configurado para encaminhar cookies (todos ou um subconjunto).

Para ter mais informações, consulte [Conteúdo em cache com base em cookies](#).

Cookies

É possível configurar o CloudFront para encaminhar cookies à origem. Para obter mais informações, consulte [Conteúdo em cache com base em cookies](#).

Compartilhamento de recursos de origem cruzada (CORS)

Se quiser que o CloudFront respeite as configurações de compartilhamento de recursos entre origens, configure o CloudFront para encaminhar o cabeçalho `Origin` à origem. Para ter mais informações, consulte [Conteúdo em cache com base nos cabeçalhos de solicitação](#).

Criptografia

É possível solicitar que os usuários usem HTTPS para enviar solicitações ao CloudFront e que o CloudFront as encaminhe à origem personalizada usando o protocolo usado pelo visualizador. Para mais informações, consulte as configurações da distribuição:

- [Política de protocolo do visualizador](#)
- [Protocolo \(somente origens personalizadas\)](#)

O CloudFront encaminha solicitações HTTPS para o servidor de origem usando os protocolos SSLv3, TLSv1.0, TLSv1.1 e TLSv1.2. Para origens personalizadas, é possível escolher os protocolos SSL a serem usados pelo CloudFront na comunicação com a origem:

- Se você estiver usando o console do CloudFront, escolha os protocolos usando as caixas de seleção `Origin SSL Protocols` (Protocolos SSL da origem). Para obter mais informações, consulte [Criar uma distribuição](#).
- Se você estiver usando a API do CloudFront, especifique os protocolos usando o elemento `OriginSslProtocols`. Para mais informações, consulte [OriginSslProtocols](#) e [DistributionConfig](#) na Referência da API do Amazon CloudFront.

Se a origem for um bucket do Amazon S3, o CloudFront sempre usará o TLSv1.2.

 Important

Outras versões de SSL e TLS não são compatíveis.

Para mais informações sobre como usar HTTPS com o CloudFront, consulte [Usar HTTPS com o CloudFront](#). Para ver as listas de criptografias compatíveis com o CloudFront para comunicação HTTPS entre os visualizadores e o CloudFront e entre o CloudFront e a origem, consulte [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

Solicitações GET que incluem um corpo

Se a solicitação GET do visualizador incluir um corpo, o CloudFront retornará o código de status HTTP 403 (Proibido).

Métodos HTTP

Se você configurar o CloudFront para processar todos os métodos HTTP compatíveis, ele aceitará as seguintes solicitações de visualizadores e as encaminhará à origem personalizada:

- DELETE
- GET
- HEAD
- OPTIONS
- PATCH
- POST
- PUT

O CloudFront sempre armazena respostas às solicitações GET e HEAD em cache. Também é possível configurar o CloudFront para armazenar respostas a solicitações OPTIONS em cache. O CloudFront não armazena em cache respostas a solicitações que usam outros métodos.

Para obter informações sobre como configurar se sua origem personalizada processa esses métodos ou não, consulte a documentação referente a ela.

⚠ Important

Se você configurar o CloudFront para aceitar e encaminhar todos os métodos HTTP compatíveis com o CloudFront à origem, configure o servidor de origem para lidar com todos eles. Por exemplo, se você configurar o CloudFront para aceitar e encaminhar esses métodos porque deseja usar POST, será necessário configurar o servidor de origem para lidar com solicitações DELETE de forma apropriada para que os visualizadores não possam excluir recursos selecionados. Para obter mais informações, consulte a documentação do seu servidor HTTP.

Cabeçalhos de solicitação HTTP e comportamento do CloudFront (origens do Amazon S3 e personalizadas)

A tabela a seguir lista os cabeçalhos de solicitação HTTP que você pode encaminhar às origens personalizadas e do Amazon S3 (com as exceções observadas). Para cada cabeçalho, a tabela inclui informações sobre o seguinte:

- O comportamento do CloudFront se você não configurar o CloudFront para encaminhar o cabeçalho à origem, o que faz com que ele armazene os objetos em cache com base nos valores de cabeçalho.
- Se é possível configurar o CloudFront para armazenar os objetos em cache com base nos valores do cabeçalho em questão.

É possível configurar o CloudFront para armazenar os objetos em cache com base nos valores dos cabeçalhos Date e User-Agent, mas não recomendamos fazer isso. Há vários valores possíveis para esses cabeçalhos, e o armazenamento em cache com base nesses valores faz com que o CloudFront encaminhe significativamente mais solicitações à origem.

Para obter mais informações sobre o armazenamento em cache com base nos valores de cabeçalho, consulte [Conteúdo em cache com base nos cabeçalhos de solicitação](#).

Cabeçalho	Comportamento se você não configurar o CloudFront para armazenar em cache com base nos valores de cabeçalho	O armazenamento em cache com base nos valores de cabeçalho é compatível
Outros cabeçalhos definidos	Configurações de cache herdadas: o CloudFront encaminha os cabeçalhos para a origem.	Sim
Accept	O CloudFront remove o cabeçalho.	Sim
Accept-Charset	O CloudFront remove o cabeçalho.	Sim
Accept-Encoding	<p>Se o valor contiver gzip ou br, o CloudFront encaminhará um cabeçalho normalizado Accept-Encoding à origem.</p> <p>Para obter mais informações, consulte Suporte à compactação e Fornecer arquivos compactados.</p>	Sim
Accept-Language	O CloudFront remove o cabeçalho.	Sim
Authorization	<ul style="list-style-type: none"> Solicitações GET e HEAD: o CloudFront remove o campo de cabeçalho Authorization antes de encaminhar a solicitação à origem. Solicitações OPTIONS: o CloudFront remove o campo de cabeçalho Authorization antes de encaminhar a solicitação à origem se você 	Sim

Cabeçalho	Comportamento se você não configurar o CloudFront para armazenar em cache com base nos valores de cabeçalho	O armazenamento em cache com base nos valores de cabeçalho é compatível
	<p>configurá-lo para armazenar respostas a solicitações OPTIONS em cache.</p> <p>O CloudFront encaminha o campo de cabeçalho <code>Authorization</code> à origem se você não configurá-lo para armazenar respostas a solicitações OPTIONS em cache.</p> <ul style="list-style-type: none"> Solicitações DELETE, PATCH, POST e PUT: o CloudFront não remove o campo de cabeçalho antes de encaminhar a solicitação à origem. 	
Cache-Control	O CloudFront encaminha o cabeçalho à origem.	Não
CloudFront-Forwarded-Proto	<p>O CloudFront não adiciona o cabeçalho antes de encaminhar a solicitação à origem.</p> <p>Para obter mais informações, consulte Configurar o armazenamento em cache com base no protocolo da solicitação.</p>	Sim

Cabeçalho	Comportamento se você não configurar o CloudFront para armazenar em cache com base nos valores de cabeçalho	O armazenamento em cache com base nos valores de cabeçalho é compatível
CloudFront-Is-Desktop-Viewer	<p>O CloudFront não adiciona o cabeçalho antes de encaminhar a solicitação à origem.</p> <p>Para obter mais informações, consulte Configurar o armazenamento em cache com base no tipo de dispositivo.</p>	Sim
CloudFront-Is-Mobile-Viewer	<p>O CloudFront não adiciona o cabeçalho antes de encaminhar a solicitação à origem.</p> <p>Para obter mais informações, consulte Configurar o armazenamento em cache com base no tipo de dispositivo.</p>	Sim
CloudFront-Is-Tablet-Viewer	<p>O CloudFront não adiciona o cabeçalho antes de encaminhar a solicitação à origem.</p> <p>Para obter mais informações, consulte Configurar o armazenamento em cache com base no tipo de dispositivo.</p>	Sim
CloudFront-Viewer-Country	O CloudFront não adiciona o cabeçalho antes de encaminhar a solicitação à origem.	Sim

Cabeçalho	Comportamento se você não configurar o CloudFront para armazenar em cache com base nos valores de cabeçalho	O armazenamento em cache com base nos valores de cabeçalho é compatível
Connection	O CloudFront substitui esse cabeçalho por <code>Connection: Keep-Alive</code> antes de encaminhar a solicitação à origem.	Não
Content-Length	O CloudFront encaminha o cabeçalho à origem.	Não
Content-MD5	O CloudFront encaminha o cabeçalho à origem.	Sim
Content-Type	O CloudFront encaminha o cabeçalho à origem.	Sim
Cookie	Se você configurar o CloudFront para encaminhar cookies, ele encaminhará o campo de cabeçalho <code>Cookie</code> à origem. Em caso negativo, o CloudFront removerá o campo de cabeçalho <code>Cookie</code> . Para obter mais informações, consulte Conteúdo em cache com base em cookies .	Não
Date	O CloudFront encaminha o cabeçalho à origem.	Sim, mas não recomendado
Expect	O CloudFront remove o cabeçalho.	Sim

Cabeçalho	Comportamento se você não configurar o CloudFront para armazenar em cache com base nos valores de cabeçalho	O armazenamento em cache com base nos valores de cabeçalho é compatível
From	O CloudFront encaminha o cabeçalho à origem.	Sim
Host	O CloudFront define o valor do nome de domínio da origem associada ao objeto solicitado. Não é possível fazer o armazenamento em cache com base no cabeçalho Host para origens do Amazon S3 ou MediaStore.	Sim (personalizada) Não (S3 e MediaStore)
If-Match	O CloudFront encaminha o cabeçalho à origem.	Sim
If-Modified-Since	O CloudFront encaminha o cabeçalho à origem.	Sim
If-None-Match	O CloudFront encaminha o cabeçalho à origem.	Sim
If-Range	O CloudFront encaminha o cabeçalho à origem.	Sim
If-Unmodified-Since	O CloudFront encaminha o cabeçalho à origem.	Sim
Max-Forwards	O CloudFront encaminha o cabeçalho à origem.	Não
Origin	O CloudFront encaminha o cabeçalho à origem.	Sim

Cabeçalho	Comportamento se você não configurar o CloudFront para armazenar em cache com base nos valores de cabeçalho	O armazenamento em cache com base nos valores de cabeçalho é compatível
Pragma	O CloudFront encaminha o cabeçalho à origem.	Não
Proxy-Authenticate	O CloudFront remove o cabeçalho.	Não
Proxy-Authorization	O CloudFront remove o cabeçalho.	Não
Proxy-Connection	O CloudFront remove o cabeçalho.	Não
Range	O CloudFront encaminha o cabeçalho à origem. Para obter mais informações, consulte Como o CloudFront processa solicitações parciais de um objeto (Range GETs) .	Sim, por padrão
Referer	O CloudFront remove o cabeçalho.	Sim
Request-Range	O CloudFront encaminha o cabeçalho à origem.	Não
TE	O CloudFront remove o cabeçalho.	Não
Trailer	O CloudFront remove o cabeçalho.	Não
Transfer-Encoding	O CloudFront encaminha o cabeçalho à origem.	Não

Cabeçalho	Comportamento se você não configurar o CloudFront para armazenar em cache com base nos valores de cabeçalho	O armazenamento em cache com base nos valores de cabeçalho é compatível
Upgrade	O CloudFront remove o cabeçalho, a menos que você tenha estabelecido uma conexão WebSocket.	Não (exceto para conexões WebSocket)
User-Agent	O CloudFront substitui o valor desse campo de cabeçalho por <code>Amazon CloudFront</code> . Se você quiser que o CloudFront armazene o conteúdo em cache com base no dispositivo do usuário, consulte Configurar o armazenamento em cache com base no tipo de dispositivo .	Sim, mas não recomenda do
Via	O CloudFront encaminha o cabeçalho à origem.	Sim
Warning	O CloudFront encaminha o cabeçalho à origem.	Sim
X-Amz-Cf-Id	O CloudFront adiciona o cabeçalho à solicitação do visualizador antes de encaminhá-la à origem. O valor do cabeçalho contém uma string criptografada que identifica exclusivamente a solicitação.	Não
X-Edge-*	O CloudFront remove todos os cabeçalhos X-Edge-*	Não

Cabeçalho	Comportamento se você não configurar o CloudFront para armazenar em cache com base nos valores de cabeçalho	O armazenamento em cache com base nos valores de cabeçalho é compatível
X-Forwarded-For	O CloudFront encaminha o cabeçalho à origem. Para obter mais informações, consulte Endereços IP do cliente .	Sim
X-Forwarded-Proto	O CloudFront remove o cabeçalho.	Não
X-HTTP-Method-Override	O CloudFront remove o cabeçalho.	Sim
X-Real-IP	O CloudFront remove o cabeçalho.	Não

Versão HTTP

O CloudFront encaminha as solicitações à origem personalizada usando HTTP/1.1.

Tamanho máximo de uma solicitação e de um URL

O tamanho máximo de uma solicitação, com o caminho, a query string (se houver) e os cabeçalhos, é de 20.480 bytes.

O CloudFront cria um URL com base na solicitação. O tamanho máximo do URL é de 8.192 bytes.

Se uma solicitação ou um URL ultrapassar esses limites máximos, o CloudFront retornará o código de status HTTP 413, Request Entity Too Large (Entidade de solicitação muito grande), ao visualizador e encerrará a conexão TCP com ele.

Associação OCSP

Quando um visualizador envia uma solicitação HTTPS para um objeto, tanto ele quanto o CloudFront devem confirmar com a autoridade de certificação (CA) se o certificado SSL do domínio não foi revogado. O OCSP Stapling acelera a validação do certificado permitindo que o CloudFront valide o certificado e armazene as respostas da CA em cache, a fim de que o cliente não precise validar o certificado diretamente com a CA.

A melhoria da performance do OCSP Stapling é mais acentuada quando o CloudFront recebe inúmeras solicitações HTTPS para objetos no mesmo domínio. Cada servidor em um ponto de presença do CloudFront deve enviar uma solicitação de validação separada. Quando o CloudFront recebe um grande número de solicitações HTTPS para o mesmo domínio, cada servidor no ponto de presença logo recebe uma resposta da CA que pode "grampear" em um pacote no handshake SSL. Quando o visualizador acreditar que o certificado é válido, o CloudFront poderá fornecer o objeto solicitado. Caso sua distribuição não tenha muito tráfego em um ponto de presença do CloudFront, é provável que novas solicitações sejam direcionadas para um servidor que ainda não validou o certificado com a CA. Nesse caso, o visualizador executa separadamente a etapa de validação, e o servidor do CloudFront fornece o objeto. Esse servidor do CloudFront também envia uma solicitação de validação para a CA para que, na próxima vez que receber uma solicitação com o mesmo nome de domínio, tenha uma resposta de validação da CA.

Conexões persistentes

Ao obter uma resposta da origem, o CloudFront tenta manter a conexão por alguns segundos caso chegue outra solicitação nesse período. A manutenção de uma conexão persistente economiza o tempo necessário para restabelecer a conexão TCP e executar outro handshake TLS para solicitações subsequentes.

Para obter mais informações, inclusive como configurar a duração de conexões persistentes, consulte [Tempo limite keep alive \(somente origens personalizadas\)](#) na seção [Referência de configurações da distribuição](#).

Protocolos

O CloudFront encaminha solicitações HTTP ou HTTPS ao servidor de origem levando em consideração:

- O protocolo da solicitação enviada pelo visualizador ao CloudFront: HTTP ou HTTPS.

- O valor do campo Origin Protocol Policy (Política de protocolo da origem) no console do CloudFront ou, se você estiver usando a API do CloudFront, o elemento `OriginProtocolPolicy` no tipo complexo `DistributionConfig`. No console do CloudFront, as opções são HTTP Only (Somente HTTP), HTTPS Only (Somente HTTPS) e Match Viewer (Corresponder visualizador).

Se você especificar HTTP Only (Somente HTTP) ou HTTPS Only (Somente HTTPS), o CloudFront encaminhará as solicitações ao servidor de origem usando o protocolo especificado, independentemente do protocolo da solicitação do visualizador.

Se você especificar Match Viewer (Corresponder visualizador), o CloudFront encaminhará as solicitações ao servidor de origem usando o protocolo da solicitação do visualizador. O CloudFront armazenará o objeto em cache somente uma vez se os visualizadores fizerem solicitações usando protocolos HTTP e HTTPS.

Important

Se o CloudFront encaminhar uma solicitação à origem usando o protocolo HTTPS, e o servidor de origem retornar um certificado inválido ou autoassinado, o CloudFront interromperá a conexão TCP.

Para obter informações sobre como atualizar uma distribuição usando o console do CloudFront, consulte [Atualizar uma distribuição](#). Para obter informações sobre como atualizar uma distribuição usando a API do CloudFront, acesse [UpdateDistribution](#) na Referência da API do Amazon CloudFront.

Strings de consulta

É possível configurar se o CloudFront encaminha parâmetros de query strings à origem. Para obter mais informações, consulte [Conteúdo em cache com base em parâmetros de string de consulta](#).

Tentativas e tempo limite de conexão da origem

Tempo limite de conexão da origem é o número de segundos que o CloudFront aguarda ao tentar estabelecer uma conexão com a origem.

Tentativas de conexão da origem é o número de vezes que o CloudFront tenta se conectar à origem.

Juntas, essas configurações determinam por quanto tempo o CloudFront tenta se conectar à origem antes de fazer o failover para a origem secundária, no caso de um grupo de origens, ou retornar uma resposta de erro ao visualizador. Por padrão, o CloudFront aguarda até 30 segundos (3 tentativas de 10 segundos cada) antes de tentar se conectar à origem secundária ou retornar uma resposta de erro. Esse tempo pode ser reduzido especificando um tempo limite de conexão mais curto, menos tentativas ou ambos.

Para obter mais informações, consulte [Controlar tempos limite e tentativas da origem](#).

Tempo limite de resposta da origem

O tempo limite de resposta da origem, também conhecido como tempo limite de leitura da origem ou tempo limite de solicitação da origem, aplica-se a estes dois valores:

- A quantidade de tempo, em segundos, que o CloudFront aguarda uma resposta após o encaminhamento de uma solicitação à origem.
- A quantidade de tempo, em segundos, que o CloudFront aguarda após o recebimento de um pacote de resposta da origem e antes do recebimento do próximo pacote.

O comportamento do CloudFront depende do método HTTP da solicitação do visualizador:

- Solicitações GET e HEAD: se a origem não responder ou parar de responder dentro da duração do tempo limite da resposta, o CloudFront interromperá a conexão. Se o número especificado de [tentativas de conexão da origem](#) for maior que 1, o CloudFront tentará obter uma resposta completa novamente. O CloudFront tenta até 3 vezes, conforme determinado pelo valor da configuração de tentativas de conexão da origem. Se a origem não responder durante a terceira tentativa, o CloudFront não tentará novamente enquanto não receber outra solicitação de conteúdo na mesma origem.
- Solicitações DELETE, OPTIONS, PATCH, PUT e POST: se a origem não responder em 30 segundos, o CloudFront interromperá a conexão e não tentará entrar em contato com a origem novamente. O cliente pode reenviar a solicitação, se necessário.

Para mais informações, inclusive como configurar o tempo limite de resposta da origem, consulte [Tempo limite de resposta \(somente origens personalizadas\)](#).

Solicitações simultâneas para o mesmo objeto (recolhimento de solicitações)

Quando um local da borda do CloudFront recebe uma solicitação de um objeto, e o objeto não está no cache ou o objeto em cache expirou, o CloudFront envia imediatamente a solicitação para a origem. No entanto, se houver solicitações simultâneas para o mesmo objeto, ou seja, se solicitações adicionais para o mesmo objeto (com a mesma chave de cache) chegarem ao local da borda antes de o CloudFront receber a resposta à primeira solicitação, o CloudFront fará uma pausa antes de encaminhar solicitações adicionais à origem. Essa breve pausa ajuda a reduzir a carga na origem. O CloudFront envia a resposta da solicitação original a todas as solicitações recebidas enquanto estava em pausa. Isso é chamado de recolhimento de solicitações. Nos logs do CloudFront, a primeira solicitação é identificada como `Miss` no campo `x-edge-result-type` e as solicitações recolhidas são identificadas como `Hit`. Para mais informações sobre os logs do CloudFront, consulte [the section called “Registro em log do CloudFront e de funções de borda”](#).

O CloudFront apenas recolhe solicitações que compartilham uma [chave de cache](#). Se as solicitações adicionais não compartilharem a mesma chave de cache porque, por exemplo, você configurou o CloudFront para armazenar em cache com base nas strings de consulta, nos cookies ou nos cabeçalhos da solicitação, o CloudFront encaminhará todas as solicitações com uma chave de cache exclusiva à origem.

Se quiser evitar o recolhimento de todas as solicitações, será possível usar a política de cache gerenciada `CachingDisabled`, que também impede o armazenamento em cache. Para ter mais informações, consulte [Usar políticas de cache gerenciadas](#).

Se você quiser evitar o recolhimento das solicitações para objetos específicos, defina a TTL mínima para o comportamento de cache como 0 e configure a origem para enviar `Cache-Control: private`, `Cache-Control: no-store`, `Cache-Control: no-cache`, `Cache-Control: max-age=0` ou `Cache-Control: s-maxage=0`. Essas configurações vão aumentar a carga na origem e introduzir latência adicional para as solicitações simultâneas que são pausadas enquanto o CloudFront aguarda a resposta à primeira solicitação.

Important

Atualmente, o CloudFront não comporta recolher solicitações caso você habilite o encaminhamento de cookies na [política de cache](#), na [política de solicitação de origem](#) ou nas configurações de cache herdadas.

User-Agent Cabeçalho

Se você quiser que o CloudFront armazene diferentes versões dos objetos em cache com base no dispositivo usado pelo usuário para visualizar o conteúdo, recomendamos configurar o CloudFront para encaminhar um ou mais dos cabeçalhos à origem personalizada:

- `CloudFront-Is-Desktop-Viewer`
- `CloudFront-Is-Mobile-Viewer`
- `CloudFront-Is-SmartTV-Viewer`
- `CloudFront-Is-Tablet-Viewer`

Com base no valor do cabeçalho `User-Agent`, o CloudFront define o valor desses cabeçalhos como `true` ou `false` antes de encaminhar a solicitação para a origem. Se o dispositivo se encaixar em mais de uma categoria, mais de um valor poderá ser `true`. Por exemplo, para alguns tablets, o CloudFront pode definir tanto `CloudFront-Is-Mobile-Viewer` quanto `CloudFront-Is-Tablet-Viewer` como `true`. Para mais informações sobre como configurar o CloudFront para armazenar em cache com base nos cabeçalhos de solicitação, consulte [Conteúdo em cache com base nos cabeçalhos de solicitação](#).

É possível configurar o CloudFront para armazenar os objetos em cache com base nos valores do cabeçalho `User-Agent`, mas não recomendamos fazer isso. Há vários valores possíveis para o cabeçalho `User-Agent`, e o armazenamento em cache com base nesses valores faz com que o CloudFront encaminhe significativamente mais solicitações à origem.

Se você não configurar o CloudFront para armazenar os objetos em cache com base nos valores do cabeçalho `User-Agent`, o CloudFront adicionará um cabeçalho `User-Agent` com o seguinte valor antes de encaminhar uma solicitação à origem:

```
User-Agent = Amazon CloudFront
```

O CloudFront adiciona esse cabeçalho, independentemente se a solicitação do visualizador inclui um cabeçalho `User-Agent` ou não. Se a solicitação do visualizador incluir um cabeçalho `User-Agent`, o CloudFront o removerá.

Como o CloudFront processa respostas da sua origem personalizada

Saiba como o CloudFront processa respostas da origem personalizada.

Sumário

- [Respostas 100 Continue](#)
- [Armazenamento em cache](#)
- [Solicitações canceladas](#)
- [Negociação de conteúdo](#)
- [Cookies](#)
- [Conexões TCP interrompidas](#)
- [Cabeçalhos de resposta HTTP que o CloudFront remove ou substitui](#)
- [Tamanho máximo do arquivo armazenável em cache](#)
- [Origem indisponível](#)
- [Redirecionamentos](#)
- [Transfer-EncodingCabeçalho](#)

Respostas **100 Continue**

Não é possível que a origem envie mais de uma resposta 100-Continue ao CloudFront. Após a primeira resposta 100-Continue, o CloudFront espera uma resposta HTTP 200 OK. Se a origem enviar outra resposta 100-Continue após a primeira, o CloudFront retornará um erro.

Armazenamento em cache

- Certifique-se de que o servidor de origem defina valores válidos e precisos para os campos de cabeçalho Date e Last-Modified.
- Normalmente, o CloudFront respeita um cabeçalho Cache-Control: no-cache na resposta da origem. Para ver uma exceção, consulte [Solicitações simultâneas para o mesmo objeto \(recolhimento de solicitações\)](#).

Solicitações canceladas

Se o objeto não estiver no cache de ponto de presença e o visualizador encerrar a sessão (por exemplo, fechar o navegador) depois de o CloudFront obter o objeto da origem, mas antes de conseguir fornecer o objeto solicitado, ele não armazenará o objeto em cache no ponto de presença.

Negociação de conteúdo

Se a origem retornar Vary: * na resposta e o valor de Minimum TTL do comportamento de cache correspondente for 0, o CloudFront armazenará o objeto em cache, mas, mesmo assim,

encaminhará todas as solicitações subsequentes do objeto à origem a fim de confirmar se o cache contém a versão mais recente do objeto. O CloudFront não inclui cabeçalhos condicionais, como `If-None-Match` ou `If-Modified-Since`. Conseqüentemente, a origem retorna o objeto ao CloudFront em resposta a cada solicitação.

Se a origem retornar `Vary: *` na resposta e o valor de `Minimum TTL` do comportamento de cache correspondente for qualquer outro valor, o CloudFront processará o cabeçalho `Vary` conforme descrito em [Cabeçalhos de resposta HTTP que o CloudFront remove ou substitui](#).

Cookies

Se você permitir cookies para um comportamento de cache e a origem retornar cookies com um objeto, o CloudFront armazenará tanto o objeto quanto os cookies em cache. Observe que isso reduz a capacidade de armazenamento em cache de um objeto. Para obter mais informações, consulte [Conteúdo em cache com base em cookies](#).

Conexões TCP interrompidas

Se a conexão TCP entre o CloudFront e a origem cair enquanto a origem estiver retornando um objeto ao CloudFront, o comportamento do CloudFront dependerá da inclusão ou não de um cabeçalho `Content-Length` na resposta pela origem:

- **Cabeçalho `Content-Length`:** o CloudFront retorna o objeto ao visualizador assim que o obtém da origem. No entanto, se o valor do cabeçalho `Content-Length` não corresponder ao tamanho do objeto, o CloudFront não o armazenará em cache.
- **`Transfer-Encoding: Chunked`:** o CloudFront retorna o objeto ao visualizador assim que o obtém da origem. No entanto, se a resposta em partes não for concluída, o CloudFront não armazenará o objeto em cache.
- **Sem cabeçalho `Content-Length`:** o CloudFront retorna o objeto ao visualizador e o armazena em cache, mas o objeto não pode ser concluído. Sem um cabeçalho `Content-Length`, o CloudFront não consegue determinar se a conexão TCP foi interrompida de forma acidental ou proposicional.

Recomendamos que você configure o servidor HTTP para adicionar um cabeçalho `Content-Length` a fim de impedir que o CloudFront armazene objetos parciais em cache.

Cabeçalhos de resposta HTTP que o CloudFront remove ou substitui

O CloudFront remove ou atualiza os seguintes campos de cabeçalho antes de encaminhar a resposta da origem ao visualizador:

- **Set-Cookie**: se você configurar o CloudFront para encaminhar cookies, ele encaminhará o campo de cabeçalho Set-Cookie aos clientes. Para obter mais informações, consulte [Conteúdo em cache com base em cookies](#).
- **Trailer**
- **Transfer-Encoding**: se a origem retornar esse campo de cabeçalho, o CloudFront definirá o valor de chunked antes de retornar a resposta ao visualizador.
- **Upgrade**
- **Vary** – Observe o seguinte:
 - Se você configurar o CloudFront para encaminhar os cabeçalhos específicos do dispositivo à origem (CloudFront-Is-Desktop-Viewer, CloudFront-Is-Mobile-Viewer, CloudFront-Is-SmartTV-Viewer, CloudFront-Is-Tablet-Viewer) e a origem para retornar Vary:User-Agent ao CloudFront, o CloudFront retornará Vary:User-Agent ao visualizador. Para obter mais informações, consulte [Configurar o armazenamento em cache com base no tipo de dispositivo](#).
 - Se você configurar a origem para incluir Accept-Encoding ou Cookie no cabeçalho Vary, o CloudFront incluirá os valores na resposta ao visualizador.
 - Se você configurar o CloudFront para encaminhar cabeçalhos à origem e configurar a origem para retornar os nomes de cabeçalho ao CloudFront no cabeçalho Vary (por exemplo, Vary:Accept-Charset, Accept-Language), o CloudFront retornará o cabeçalho Vary com esses valores ao visualizador.
 - Para obter informações sobre como o CloudFront processa um valor de * no cabeçalho Vary, consulte [Negociação de conteúdo](#).
 - Se você configurar a origem para incluir qualquer outro valor no cabeçalho Vary, o CloudFront removerá os valores antes de retornar a resposta ao visualizador.
- **Via**: o CloudFront define o seguinte valor na resposta para o visualizador:

Via: *versão HTTP string alfanumérica*.cloudfront.net (CloudFront)

Por exemplo, o valor é semelhante ao seguinte:

Via: 1.1 1026589cc7887e7a0dc7827b4example.cloudfront.net (CloudFront)

Tamanho máximo do arquivo armazenável em cache

O tamanho máximo do corpo de uma resposta salva pelo CloudFront no cache é de 50 GB. Isso inclui respostas de transferência em partes que não especificam o valor de cabeçalho Content-Length.

É possível usar o CloudFront para armazenar em cache um objeto maior que isso usando solicitações de intervalo para solicitar os objetos em partes que sejam cada uma de 50 GB ou menores. O CloudFront armazena essas partes em cache porque cada uma delas é de 50 GB ou menor. Depois que o visualizador recuperar todas as partes do objeto, ele poderá reconstruir o objeto original, maior. Para ter mais informações, consulte [Usar solicitações de intervalo para armazenar objetos grandes em cache](#).

Origem indisponível

Se o servidor de origem estiver indisponível e o CloudFront receber uma solicitação de um objeto que está no cache do ponto de presença, mas expirou (por exemplo, porque o período especificado na diretiva `Cache-Control max-age` passou), o CloudFront fornecerá a versão expirada do objeto ou uma página de erro personalizada. Para mais informações sobre o comportamento do CloudFront ao configurar páginas de erro personalizadas, consulte [Como o CloudFront processará erros quando páginas de erro personalizadas estiverem configuradas](#).

Em alguns casos, um objeto que é raramente solicitado é removido e se torna indisponível no ponto de presença de caches. O CloudFront não pode fornecer um objeto que foi removido.

Redirecionamentos

Se você alterar a localização de um objeto no servidor de origem, poderá configurar o servidor da Web para redirecionar as solicitações para o novo local. Depois de configurar o redirecionamento, a primeira vez que um visualizador enviar uma solicitação para o objeto, o CloudFront a enviará à origem, e a origem responderá com um redirecionamento (por exemplo, `302 Moved Temporarily`). O CloudFront armazena o redirecionamento em cache e o retorna ao visualizador. O CloudFront não acompanha o redirecionamento.

Você pode configurar o servidor da web para redirecionar as solicitações para um destes locais:

- O novo URL do objeto no servidor de origem. Ao seguir o redirecionamento para o novo URL, o visualizador ignora o CloudFront e vai diretamente à origem. Por isso, recomendamos que você não redirecione as solicitações para o novo URL do objeto na origem.

- O novo URL do CloudFront do objeto. Quando o visualizador envia a solicitação que contém o novo URL do CloudFront, o CloudFront obtém o objeto do novo local na origem, armazena-o em cache no ponto de presença e retorna-o ao visualizador. As solicitações subsequentes do objeto são fornecidas pelo ponto de presença. Isso evita a latência e a carga associadas à solicitação do objeto pelo visualizador da origem. No entanto, cada nova solicitação do objeto será cobrada por duas solicitações ao CloudFront.

Transfer-EncodingCabeçalho

O CloudFront é compatível apenas com o valor `chunked` do cabeçalho `Transfer-Encoding`. Se a origem retornar `Transfer-Encoding: chunked`, o CloudFront retornará o objeto ao cliente assim que for recebido no ponto de presença e o armazenará em partes para solicitações subsequentes.

Se o visualizador fizer uma solicitação `Range GET` e a origem retornar `Transfer-Encoding: chunked`, o CloudFront retornará o objeto inteiro ao visualizador, em vez do intervalo solicitado.

Recomendamos que você use codificação em partes se o tamanho do conteúdo da sua resposta não puder ser predeterminado. Para obter mais informações, consulte [Conexões TCP interrompidas](#).

Comportamento de solicitações e respostas para grupos de origens

As solicitações a um grupo de origens funcionarão da mesma forma que as solicitações a uma origem que não esteja configurada como um grupo de origens, exceto quando houver um failover da origem. Assim como ocorre com qualquer outra origem, quando o CloudFront recebe uma solicitação, e o conteúdo já está armazenado em cache em um ponto de presença, o conteúdo é fornecido aos visualizadores do cache. Quando há uma falha de cache e a origem é um grupo de origens, as solicitações do visualizador são encaminhadas para a origem primária no grupo de origens.

O comportamento da solicitação e da resposta para a origem primária é o mesmo de uma origem que não esteja incluída em um grupo de origens. Para obter mais informações, consulte [Comportamento de solicitações e respostas para origens do Amazon S3](#) e [Comportamento de solicitações e respostas para origens personalizadas](#).

A tabela a seguir descreve o comportamento para origem de failover quando a origem primária retorna códigos de status HTTP específicos:

- Código de status HTTP 2xx (êxito): o CloudFront armazena o arquivo e o retorna ao visualizador.
- Código de status HTTP 3xx (redirecionamento): o CloudFront retorna o código de status ao visualizador.
- Código de status HTTP 4xx ou 5xx (erro de cliente/servidor): se o código de status retornado foi configurado para failover, o CloudFront envia a mesma solicitação à origem secundária no grupo de origens.
- Código de status HTTP 4xx ou 5xx (erro de cliente/servidor): se o código de status retornado não foi configurado para failover, o CloudFront retornará o erro ao visualizador.

O CloudFront faz failover para a origem secundária somente quando o método HTTP da solicitação do visualizador for GET, HEAD ou OPTIONS. O CloudFront não faz failover quando o visualizador envia um método HTTP diferente (por exemplo POST, PUT etc.).

Quando o CloudFront envia uma solicitação a uma origem secundária, o comportamento de resposta é o mesmo que para uma origem do CloudFront que não esteja em um grupo de origens.

Para mais informações sobre grupos de origens, consulte [Otimizar a alta disponibilidade com o failover de origem do CloudFront](#).

Adicionar cabeçalhos personalizados às solicitações de origem

Você pode configurar o CloudFront para adicionar cabeçalhos personalizados às solicitações que ele envia à sua origem. É possível usar cabeçalhos personalizados para enviar e reunir informações da origem que você não recebe com solicitações regulares do visualizador. É possível até mesmo personalizar os cabeçalhos para cada origem. O CloudFront comporta cabeçalhos personalizados para origens personalizadas e origens do Amazon S3.

Sumário

- [Casos de uso](#)
- [Configurar o CloudFront para adicionar cabeçalhos personalizados às solicitações de origem](#)
- [Cabeçalhos personalizados que o CloudFront não pode adicionar às solicitações da origem](#)
- [Configurar o CloudFront para encaminhar o cabeçalho Authorization](#)

Casos de uso

É possível usar cabeçalhos personalizados, como nos seguintes exemplos:

Identificar solicitações do CloudFront

Você pode identificar as solicitações que a origem recebe do CloudFront. Isso poderá ser útil se você quiser saber se os usuários estão ignorando o CloudFront ou se você estiver usando mais de uma CDN e quiser obter informações sobre quais solicitações são provenientes de cada CDN.

Note

Se estiver usando uma origem do Amazon S3 e habilitar o [Registro em log de acesso do servidor do Amazon S3](#), os logs não incluirão as informações de cabeçalho.

Determinar quais solicitações vêm de uma distribuição específica

Se você configurar mais de uma distribuição do CloudFront para usar a mesma origem, poderá adicionar cabeçalhos personalizados diferentes em cada distribuição. Depois, você poderá usar os logs da origem para determinar quais solicitações vieram de qual distribuição do CloudFront.

Habilitar o compartilhamento de recursos de origem cruzada (CORS)

Se alguns dos visualizadores não forem compatíveis com o compartilhamento de recursos de origem cruzada (CORS), você poderá configurar o CloudFront para sempre adicionar o cabeçalho da `Origin` às solicitações que ele enviar à sua origem. Depois, você poderá configurar sua origem para retornar o cabeçalho `Access-Control-Allow-Origin` para cada solicitação. Você também deve [configurar o CloudFront para respeitar as configurações do CORS](#).

Controlar o acesso ao conteúdo

Você pode usar cabeçalhos personalizados para controlar o acesso ao conteúdo. Ao configurar a origem para responder às solicitações somente quando elas incluírem um cabeçalho personalizado que é adicionado pelo CloudFront, você evita que os usuários ignorem o CloudFront e acessem o conteúdo diretamente na origem. Para ter mais informações, consulte [Restringir o acesso a arquivos em origens personalizadas](#).

Configurar o CloudFront para adicionar cabeçalhos personalizados às solicitações de origem

A fim de configurar uma distribuição para adicionar cabeçalhos personalizados às solicitações que ela enviar à sua origem, atualize a configuração da origem usando um dos seguintes métodos:

- Console do CloudFront: ao criar ou atualizar uma distribuição, especifique nomes e valores de cabeçalho nas configurações Adicionar cabeçalhos personalizados. Para ter mais informações, consulte [Adicionar cabeçalho personalizado](#).
- API do CloudFront : para cada origem à qual você deseja adicionar cabeçalhos personalizados, especifique os nomes e os valores dos cabeçalhos no campo CustomHeaders em Origin. Para ter mais informações, consulte [CreateDistribution](#) ou [UpdateDistribution](#) na Referência da API do Amazon CloudFront.

Se os nomes e os valores dos cabeçalhos especificados ainda não estiverem na solicitação do visualizador, o CloudFront os adicionará à solicitação da origem. Se houver um cabeçalho presente, o CloudFront substituirá o valor antes de encaminhar a solicitação para a origem.

Para ver as cotas que se aplicam aos cabeçalhos personalizados de origem, consulte [Cotas para cabeçalhos](#).

Cabeçalhos personalizados que o CloudFront não pode adicionar às solicitações da origem

Não é possível configurar o CloudFront para adicionar um dos seguintes cabeçalhos às solicitações que ele envia à origem:

- Cache-Control
- Connection
- Content-Length
- Cookie
- Host
- If-Match
- If-Modified-Since
- If-None-Match
- If-Range
- If-Unmodified-Since
- Max-Forwards
- Pragma
- Proxy-Authorization

- Proxy-Connection
- Range
- Request-Range
- TE
- Trailer
- Transfer-Encoding
- Upgrade
- Via
- Cabeçalhos que começam com X-Amz -
- Cabeçalhos que começam com X-Edge -
- X-Real-IP

Configurar o CloudFront para encaminhar o cabeçalho **Authorization**

Quando o CloudFront encaminha uma solicitação de visualizador para sua origem, o CloudFront remove alguns cabeçalhos de visualizador por padrão, incluindo o `Authorization` cabeçalho. Para garantir que sua origem sempre receba o `Authorization` cabeçalho em solicitações de origem, você tem as seguintes opções:

- Adicione o `Authorization` cabeçalho à chave de cache usando uma política de cache. Todos os cabeçalhos na chave de cache são automaticamente incluídos nas solicitações de origem. Para ter mais informações, consulte [Controlar a chave de cache com uma política](#).
- Use uma política de solicitação de origem que encaminha todos os cabeçalhos do visualizador para a origem. Você não pode encaminhar o `Authorization` cabeçalho individualmente em uma política de solicitação de origem, mas ao encaminhar todos os cabeçalhos do visualizador, o CloudFront inclui o `Authorization` cabeçalho nas solicitações do visualizador. O CloudFront fornece uma política de solicitação de origem gerenciada para esse caso de uso, chamada `Managed-AllViewer`. Para ter mais informações, consulte [Usar políticas de solicitação de origem gerenciadas](#).

Como o CloudFront processa solicitações parciais de um objeto (Range GETs)

Para um objeto grande, o visualizador (navegador ou outro cliente) pode fazer várias solicitações GET e usar o cabeçalho de solicitação Range para baixar o objeto em partes menores. Essas solicitações de intervalos de bytes, também conhecidas como solicitações Range GET, melhoram a eficiência de downloads parciais e a recuperação de transferências parciais com falha.

Ao receber uma solicitação Range GET, o CloudFront verifica o cache no local de borda que recebeu a solicitação. Se o cache desse ponto de presença já contiver todo o objeto ou a parte solicitada dele, o CloudFront fornecerá imediatamente o intervalo solicitado do cache.

Se o cache não contiver o intervalo solicitado, o CloudFront encaminhará a solicitação à origem. (Para otimizar a performance, o CloudFront pode solicitar um intervalo maior que o solicitado pelo cliente em Range GET.) O que acontece em seguida depende se a origem é compatível ou não com solicitações Range GET:

- Se a origem for compatível com solicitações **Range GET**: ele exibirá o intervalo solicitado. O CloudFront fornece o intervalo solicitado e o armazena em cache para futuras solicitações. (O Amazon S3 oferece suporte a solicitações Range GET, assim como muitos servidores HTTP.)
- Se a origem não for compatível com solicitações **Range GET**: ele exibirá todo o objeto. O CloudFront atende à solicitação atual enviando o objeto inteiro enquanto também o armazena em cache para solicitações futuras. Depois de armazenar o objeto inteiro em cache em um cache de ponto de presença, o CloudFront responde a novas solicitações Range GET fornecendo o intervalo solicitado.

Nos dois casos, o CloudFront começa a fornecer o intervalo ou o objeto solicitado ao usuário final assim que o primeiro byte chega da origem.

Note

Se o visualizador fizer uma solicitação Range GET e a origem retornar `Transfer-Encoding: chunked`, o CloudFront retornará o objeto inteiro ao visualizador, em vez do intervalo solicitado.

Normalmente, o CloudFront segue a especificação RFC do cabeçalho Range. No entanto, se os cabeçalhos Range não seguirem os seguintes requisitos, o CloudFront retornará o código de status 200 com todo o objeto, em vez do código de status 206 com os intervalos especificados:

- Os intervalos devem estar indicados em ordem crescente. Por exemplo, 100-200, 300-400 é válido; 300-400, 100-200 não é válido.
- Os intervalos não devem se sobrepor. Por exemplo, 100-200, 150-250 não é válido.
- Todas as especificações dos intervalos devem ser válidas. Por exemplo, você não pode especificar um valor negativo como parte de um intervalo.

Para obter mais informações sobre o cabeçalho de solicitação Range, consulte [Solicitações de intervalo](#) na RFC 7233, ou [Intervalo](#) no MDN Web Docs.

Usar solicitações de intervalo para armazenar objetos grandes em cache

Quando o armazenamento em cache está ativado, o CloudFront não recupera nem armazena em cache um objeto maior que 50 GB. Quando uma origem indica que o objeto é maior que isso (no cabeçalho de resposta Content-Length), o CloudFront fecha a conexão com a origem e exibe um erro ao visualizador. (Com o armazenamento em cache desativado, o CloudFront pode recuperar um objeto maior que isso da origem e passá-lo para o visualizador. No entanto, o CloudFront não armazena o objeto em cache.)

No entanto, com solicitações de intervalo, é possível usar o CloudFront para armazenar em cache um objeto maior que o [tamanho máximo de arquivo armazenável em cache](#).

Example Exemplo

1. Pense em uma origem com um objeto de 100 GB. Com o armazenamento em cache habilitado, o CloudFront não recupera nem armazena em cache um objeto desse tamanho. No entanto, o visualizador pode enviar várias solicitações de intervalo para recuperar esse objeto em partes, sendo cada uma menor que 50 GB.
2. O visualizador pode solicitar o objeto em partes de 20 GB enviando uma solicitação com o cabeçalho Range: bytes=0-21474836480 para recuperar a primeira parte, outra solicitação com o cabeçalho Range: bytes=21474836481-42949672960 para recuperar a próxima parte, e assim por diante.
3. Quando o visualizador tiver recebido todas as partes, ele pode combiná-las para construir o objeto original de 100 GB.

4. Nesse caso, o CloudFront armazena em cache cada uma das partes de 20 GB do objeto e pode responder a solicitações subsequentes para a mesma parte do cache.

Como o CloudFront processa códigos de status HTTP 3xx da origem

Quando o CloudFront solicita um objeto do bucket do Amazon S3 ou de um servidor de origem personalizado, a origem às vezes retorna um código de status HTTP 3xx. Isso normalmente indica uma das seguintes situações:

- O URL do objeto foi alterado (por exemplo, códigos de status 301, 302, 307 ou 308)
- O objeto não foi alterado desde a última vez que o CloudFront o solicitou (código de status 304)

O CloudFront armazena em cache as respostas 3xx de acordo com as configurações na distribuição do CloudFront e os cabeçalhos na resposta. O CloudFront armazena 307 e 308 respostas em cache somente quando você inclui o cabeçalho `Cache-Control` nas respostas da origem. Para ter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Se a origem retornar um código de status de redirecionamento (por exemplo, 301 ou 307), o CloudFront não acompanhará o redirecionamento. O CloudFront transmite a resposta 301 ou 307 ao visualizador, que pode acompanhar o redirecionamento enviando uma nova solicitação.

Como o CloudFront processa códigos de status HTTP 4xx e 5xx da origem

Quando o CloudFront solicita um objeto do bucket do Amazon S3 ou servidor de origem personalizada, a origem pode retornar um código de status HTTP 4xx ou 5xx, que indica um erro. O comportamento do CloudFront depende do seguinte:

- Se você configurou páginas de erro personalizadas
- Se você configurou o tempo que o CloudFront armazenará em cache as respostas a erros da origem (TTL mínima de armazenamento de erros em cache).
- O código do status
- Em relação a códigos de status 5xx, se o objeto solicitado estiver no cache de borda do CloudFront.

- Para alguns códigos de status 4xx, se a origem exibir um cabeçalho `Cache-Control max-age` ou `Cache-Control s-maxage`

O CloudFront sempre armazena respostas às solicitações GET e HEAD em cache. Também é possível configurar o CloudFront para armazenar respostas a solicitações OPTIONS em cache. O CloudFront não armazena em cache respostas a solicitações que usam outros métodos.

Se a origem não responder, a solicitação do CloudFront à origem expirará, o que é considerado um erro HTTP 5xx da origem, mesmo que a origem não tenha respondido com esse erro. Nessa situação, o CloudFront continua fornecendo conteúdo armazenado em cache. Para obter mais informações, consulte [Origem indisponível](#).

Se você ativou o registro em log, o CloudFront gravará os resultados nos logs, independentemente do código de status HTTP.

Para mais informações sobre os recursos e as opções relacionadas à mensagem de erro retornada pelo CloudFront, consulte o seguinte:

- Para obter informações sobre as configurações de páginas de erro personalizadas no console do CloudFront, consulte [Páginas de erro personalizadas e erro de armazenamento em cache](#).
- Para mais informações sobre o TTL mínimo de armazenamento de erros em cache no console do CloudFront, consulte [Erro ao armazenar TTL mínimo em cache \(segundos\)](#).
- Para consultar uma lista de códigos de status HTTP armazenados em cache pelo CloudFront, consulte [Códigos de status HTTP 4xx e 5xx armazenados em cache pelo CloudFront](#).

Tópicos

- [Como o CloudFront processará erros quando páginas de erro personalizadas estiverem configuradas](#)
- [Como o CloudFront processará erros quando páginas de erro personalizadas não estiverem configuradas](#)
- [Códigos de status HTTP 4xx e 5xx armazenados em cache pelo CloudFront](#)

Como o CloudFront processará erros quando páginas de erro personalizadas estiverem configuradas

Se você configurou páginas de erro personalizadas, o comportamento do CloudFront será determinado de acordo com o objeto solicitado estar ou não no cache de ponto de presença.

O objeto solicitado não está no cache de borda

O CloudFront continuará tentando obter o objeto solicitado da origem quando todas estas opções forem verdadeiras:

- Um visualizador solicita um objeto.
- O objeto não está no ponto de presença de caches
- Sua origem retorna um código de status HTTP 4xx ou 5xx e uma das seguintes situações é verdadeira:
 - A origem retorna um código de status HTTP 5xx, em vez de retornar um código de status 304 (Não modificado) ou uma versão atualizada do objeto
 - A origem retorna um código de status HTTP 4xx que não é restrito por um cabeçalho de controle de cache e é incluído na seguinte lista de códigos de status: [Códigos de status HTTP 4xx e 5xx armazenados em cache pelo CloudFront](#).
 - A origem retorna um código de status HTTP 4xx sem um cabeçalho `Cache-Control max-age` ou `Cache-Control s-maxage`, e o código de status é incluído na seguinte lista de códigos de status: [Control Códigos de status HTTP 4xx que o CloudFront armazena em cache com base em cabeçalhos de Cache-Control](#).

O CloudFront faz o seguinte:

1. No cache de ponto de presença do CloudFront que recebeu a solicitação do visualizador, o CloudFront verifica a configuração da distribuição e obtém o caminho da página de erro personalizada correspondente ao código de status retornado pela origem.
2. O CloudFront encontra o primeiro comportamento de cache na distribuição que tem um padrão de caminho correspondente ao caminho da página de erro personalizada.
3. O ponto de presença do CloudFront envia uma solicitação da página de erro personalizada à origem especificada no comportamento de cache.
4. A origem retorna a página de erro personalizada para o ponto de presença.

5. O CloudFront retorna a página de erro personalizada ao visualizador que fez a solicitação e a armazena em cache por no máximo:
 - A quantidade de tempo especificada pelo TTL mínimo de armazenamento de erros em cache (10 segundos, por padrão)
 - A quantidade de tempo especificada por um cabeçalho `Cache-Control max-age` ou `Cache-Control s-maxage` retornado pela origem quando a primeira solicitação gerou o erro.
6. Após o término do tempo de armazenamento em cache (determinado na Etapa 5), o CloudFront tentará obter o objeto solicitado novamente encaminhando outra solicitação à origem. O CloudFront continua tentando em intervalos especificados pelo TTL mínimo de armazenamento de erros em cache.

O objeto solicitado está no cache de borda

O CloudFront continuará fornecendo o objeto que está no cache de ponto de presença quando todas estas opções forem verdadeiras:

- Um visualizador solicita um objeto.
- O objeto está no cache do ponto de presença, mas expirou
- A origem retorna um código de status HTTP 5xx, em vez de retornar um código de status 304 (Não modificado) ou uma versão atualizada do objeto

O CloudFront faz o seguinte:

1. Se a origem retornar um código de status 5xx, o CloudFront fornecerá o objeto mesmo se ele tiver expirado. Pela duração do TTL mínimo de armazenamento de erros em cache, o CloudFront continuará respondendo a solicitações do visualizador fornecendo o objeto do cache do ponto de presença.

Se a origem retornar um código de status 4xx, o CloudFront retornará o código de status ao visualizador, não o objeto solicitado.

2. Após o término desse TTL mínimo, o CloudFront tentará obter o objeto solicitado novamente encaminhando outra solicitação para a origem. Observe que, se o objeto não for solicitado com frequência, o CloudFront poderá removê-lo do cache do ponto de presença enquanto o servidor de origem ainda estiver retornando respostas 5xx. Para obter informações sobre o tempo de permanência de objetos de caches de ponto de presença do CloudFront, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Como o CloudFront processará erros quando páginas de erro personalizadas não estiverem configuradas

Se você não configurou páginas de erro personalizadas, o comportamento do CloudFront será determinado de acordo com o objeto solicitado estar ou não no cache de ponto de presença.

O objeto solicitado não está no cache de borda

O CloudFront continuará tentando obter o objeto solicitado da origem quando todas estas opções forem verdadeiras:

- Um visualizador solicita um objeto.
- O objeto não está no ponto de presença de caches
- Sua origem retorna um código de status HTTP 4xx ou 5xx e uma das seguintes situações é verdadeira:
 - A origem retorna um código de status HTTP 5xx, em vez de retornar um código de status 304 (Não modificado) ou uma versão atualizada do objeto
 - A origem retorna um código de status HTTP 4xx que não é restrito por um cabeçalho de controle de cache e é incluído na seguinte lista de códigos de status: [Códigos de status HTTP 4xx e 5xx armazenados em cache pelo CloudFront](#)
 - A origem retorna um código de status HTTP 4xx sem um cabeçalho `Cache-Control max-age` ou `Cache-Control s-maxage`, e o código de status é incluído na seguinte lista de códigos de status: [Control Códigos de status HTTP 4xx que o CloudFront armazena em cache com base em cabeçalhos de Cache-Control](#).

O CloudFront faz o seguinte:

1. O CloudFront retorna o código de status 4xx ou 5xx ao visualizador e também armazena o código de status no cache do ponto de presença que recebeu a solicitação por no máximo:
 - A quantidade de tempo especificada pelo TTL mínimo de armazenamento de erros em cache (10 segundos, por padrão)
 - A quantidade de tempo especificada por um cabeçalho `Cache-Control max-age` ou `Cache-Control s-maxage` retornado pela origem quando a primeira solicitação gerou o erro.
2. Durante o tempo de armazenamento em cache (determinado na etapa 1), o CloudFront responderá a solicitações subsequentes do visualizador do mesmo objeto com o código de status 4xx ou 5xx armazenado em cache.

3. Após o término do tempo de armazenamento em cache (determinado na Etapa 1), o CloudFront tentará obter o objeto solicitado novamente encaminhando outra solicitação à origem. O CloudFront continua tentando em intervalos especificados pelo TTL mínimo de armazenamento de erros em cache.

O objeto solicitado está no cache de borda

O CloudFront continuará fornecendo o objeto que está no cache de ponto de presença quando todas estas opções forem verdadeiras:

- Um visualizador solicita um objeto.
- O objeto está no cache do ponto de presença, mas expirou
- A origem retorna um código de status HTTP 5xx, em vez de retornar um código de status 304 (Não modificado) ou uma versão atualizada do objeto

O CloudFront faz o seguinte:

1. Se a origem retornar um código de erro 5xx, o CloudFront fornecerá o objeto mesmo se ele tiver expirado. Pela duração do TTL mínimo de armazenamento de erros em cache (10 segundos, por padrão), o CloudFront continua respondendo a solicitações do visualizador fornecendo o objeto do cache de ponto de presença.


Se a origem retornar um código de status 4xx, o CloudFront retornará o código de status ao visualizador, não o objeto solicitado.

2. Após o término desse TTL mínimo, o CloudFront tentará obter o objeto solicitado novamente encaminhando outra solicitação para a origem. Observe que, se o objeto não for solicitado com frequência, o CloudFront poderá removê-lo do cache do ponto de presença enquanto o servidor de origem ainda estiver retornando respostas 5xx. Para obter informações sobre o tempo de permanência de objetos de caches de ponto de presença do CloudFront, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Códigos de status HTTP 4xx e 5xx armazenados em cache pelo CloudFront

O CloudFront armazena em cache os códigos de status HTTP 4xx e 5xx retornados pela origem, dependendo do código de status específico que é retornado e se a origem retorna cabeçalhos específicos na resposta.

O CloudFront armazena em cache os códigos de status HTTP 4xx e 5xx a seguir que são exibidos pela origem. Se você configurou uma página de erro personalizada para um código de status HTTP, o CloudFront a armazenará em cache.

 Note

Se você estiver usando a política de cache gerenciada [CachingDisabled](#), o CloudFront não armazenará esses códigos de status nem páginas de erro personalizadas.

404	Não encontrado
414	URI da solicitação muito grande
500	Internal Server Error
501	Não implementado
502	Gateway inválido
503	Serviço indisponível
504	Tempo limite do gateway

Códigos de status HTTP 4xx que o CloudFront armazena em cache com base em cabeçalhos de **Cache-Control**

O CloudFront apenas armazena os seguintes códigos de status HTTP 4xx retornados pela origem se a origem retornar um cabeçalho `Cache-Control max-age` ou `Cache-Control s-maxage`. Se você configurou uma página de erro personalizada para um dos códigos de status HTTP a seguir, e a origem exibir um dos cabeçalhos de controle de cache, o CloudFront a armazenará em cache.

400	Solicitação inválida
403	Proibido
405	Método não permitido
412 ¹	Falha na pré-condição
415 ¹	Tipo de mídia incompatível

¹ O CloudFront não é compatível com a criação de páginas de erro personalizadas para esses códigos de status HTTP.

Gerar respostas a erros personalizadas

Se um objeto fornecido pelo CloudFront estiver indisponível por algum motivo, seu servidor web normalmente retornará um código de status HTTP relevante para o CloudFront para indicar isso. Por exemplo, se um visualizador solicitar um URL inválido, o servidor Web exibirá um código de status HTTP 404 (Não encontrado) para o CloudFront e o CloudFront exibirá esse código de status para o visualizador. Em vez de usar essa resposta a erros padrão, é possível criar uma resposta personalizada que o CloudFront exiba ao visualizador.

Se você configurar o CloudFront para retornar uma página de erro personalizada para um código de status HTTP, mas a página de erro personalizada não estiver disponível, o CloudFront retornará ao visualizador o código de status recebido do CloudFront da origem que contém as páginas de erro personalizadas. Por exemplo, suponha que sua origem personalizada retorne um código de status 500 e você configurou o CloudFront para obter uma página de erro personalizada para esse código em um bucket do Amazon S3. Porém, alguém excluiu a página de erro personalizada do bucket do Amazon S3 por acidente. O CloudFront retorna um código de status HTTP 404 (Não encontrado) para o visualizador que solicitou o objeto.

Quando o CloudFront retorna uma página de erro personalizada ao visualizador, você paga as cobranças padrão equivalentes do CloudFront para a página de erro personalizada, e não as cobranças do objeto solicitado. Para obter mais informações sobre as cobranças do CloudFront, consulte [Definição de preço do Amazon CloudFront](#).

Tópicos

- [Configurar o comportamento de resposta a erros](#)
- [Criar uma página de erro personalizada para códigos de status HTTP específicos](#)
- [Armazenar objetos e páginas de erro personalizadas em diferentes locais](#)
- [Alterar códigos de resposta exibidos pelo CloudFront](#)
- [Controlar por quanto tempo o CloudFront armazena erros em cache](#)

Configurar o comportamento de resposta a erros

Você tem várias opções para gerenciar como o CloudFront reage quando há um erro. Para configurar respostas de erro personalizadas, você pode usar o console e a API do CloudFront ou o AWS CloudFormation. Independentemente de como você optar por atualizar a configuração, considere as seguintes dicas e recomendações:

- Salve suas páginas de erro personalizadas em um local acessível ao CloudFront. Recomendamos que você os armazene em um bucket do Amazon S3 e que você [não os armazene no mesmo local que o restante do conteúdo do seu site ou aplicativo](#). Se você armazenar as páginas de erro personalizadas na mesma origem do seu site ou aplicativo e a origem começar a retornar erros 5xx, o CloudFront não poderá obter as páginas de erro personalizadas porque o servidor de origem não está disponível. Para obter mais informações, consulte [Armazenar objetos e páginas de erro personalizadas em diferentes locais](#).
- Certifique-se de que o CloudFront tenha permissão para obter suas páginas de erro personalizadas. Se as páginas de erro personalizadas forem armazenadas no Amazon S3, elas deverão estar acessíveis ao público ou você deve configurar um [controle de acesso à origem \(OAC\)](#) do CloudFront. Se as páginas de erro personalizadas forem armazenadas em uma origem personalizada, as páginas deverão estar acessíveis publicamente.
- (Opcional) Configure sua origem para adicionar um Cache-Control ou Expires cabeçalho junto com as páginas de erro personalizadas, se desejar. Você também pode usar a configuração de TTL mínimo de cache de erro para controlar por quanto tempo o CloudFront armazena em cache as páginas de erro personalizadas. Para ter mais informações, consulte [Controlar por quanto tempo o CloudFront armazena erros em cache](#).

Configurar respostas a erros personalizadas

Para configurar respostas de erro personalizadas no console do CloudFront, você deve ter uma distribuição do CloudFront. No console, as configurações para respostas de erro personalizadas só estão disponíveis para distribuições existentes. Para saber como criar uma distribuição, consulte [Conceitos básicos de uma distribuição básica do CloudFront](#).

Console

Para configurar respostas de erro personalizadas (console)

1. Faça login no AWS Management Console e abra a página Distributions (Distribuições) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#distributions>.
2. Na lista de distribuições, escolha a distribuição a ser atualizada.
3. Escolha a guia Error Pages (Páginas de erro) e escolha Create Custom Error Response (Criar resposta de erro personalizada).
4. Insira os valores aplicáveis. Para ter mais informações, consulte [Páginas de erro personalizadas e erro de armazenamento em cache](#).
5. Depois de inserir os valores desejados, escolha Create (Criar).

CloudFront API or AWS CloudFormation

Para configurar respostas de erro personalizadas com a API do CloudFront ou do AWS CloudFormation, use o tipo `CustomErrorResponse` em uma distribuição. Para obter mais informações, consulte:

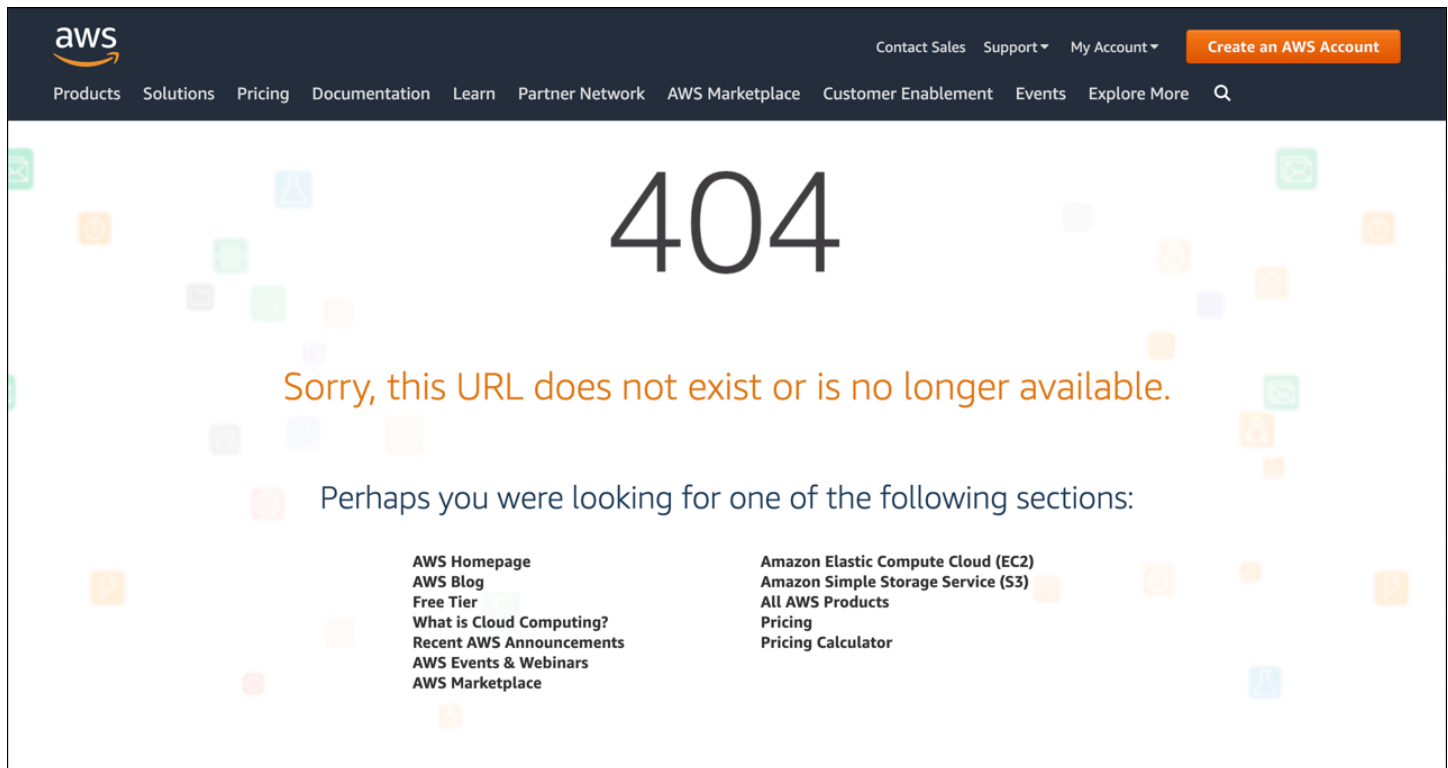
- [AWS::CloudFront::Distribution CustomErrorResponse](#) no Manual do usuário do AWS CloudFormation
- [CustomErrorResponse](#) na Referência da API do Amazon CloudFront

Criar uma página de erro personalizada para códigos de status HTTP específicos

Se você preferir exibir uma mensagem de erro personalizada em vez da mensagem padrão, como por exemplo, uma página que usa a mesma formatação do resto do seu site, você pode fazer com que o CloudFront retorne ao visualizador um objeto (como um arquivo HTML) que contenha sua mensagem de erro personalizada.

Para especificar o arquivo que deseja retornar e os erros para os quais o arquivo deve ser retornado, atualize sua distribuição do CloudFront para especificar esses valores. Para obter mais informações, consulte [Configurar o comportamento de resposta a erros](#).

Por exemplo, o seguinte item é uma página de erro personalizada:



Você pode especificar um objeto diferente para cada código de status HTTP compatível ou usar o mesmo objeto para todos os códigos de status compatíveis. Você pode optar por especificar páginas de erro personalizadas para alguns códigos de status e não para outros.

Os objetos fornecidos pelo CloudFront podem estar indisponíveis por vários motivos. Eles se encaixam em duas categorias gerais:

- Erros de cliente indicam um problema com a solicitação. Por exemplo, um objeto com o nome especificado não está disponível ou o usuário não tem as permissões necessárias para obter um objeto no bucket do Amazon S3. Quando ocorre um erro de cliente, a origem retorna um código de status HTTP no intervalo 4xx para o CloudFront.
- Erros do servidor indicam um problema com o servidor de origem. Por exemplo, o servidor HTTP está ocupado ou indisponível. Quando ocorre um erro de servidor, seu servidor de origem retorna um código de status HTTP no intervalo 5xx para o CloudFront ou o CloudFront não recebe uma resposta do seu servidor de origem por um determinado período de tempo e assume um código de status 504 (Tempo limite do Gateway).

Os códigos de status HTTP para os quais o CloudFront pode retornar uma página de erro personalizada incluem:

- 400, 403, 404, 405, 414, 416
- 500, 501, 502, 503, 504

Observações

- Se o CloudFront detectar que a solicitação pode não ser segura, retornará um erro 400 (solicitação inválida) em vez de uma página de erro personalizada.
- É possível criar uma página de erro personalizada para o código de status HTTP 416 (intervalo solicitado insatisfatório) e alterar o código de status HTTP retornado pelo CloudFront aos visualizadores quando a origem retornar um código de status 416 ao CloudFront. Para ter mais informações, consulte [Alterar códigos de resposta exibidos pelo CloudFront](#). No entanto, o CloudFront não armazena em cache as respostas do código de status 416, portanto, mesmo se você especificar um valor para o TTL mínimo de cache de erro para o código de status 416, o CloudFront não o usará.
- Em alguns casos, o CloudFront não retorna uma página de erro personalizada para o código de status HTTP 503, mesmo se você configurar o CloudFront para isso. Se o código de erro do CloudFront for `Capacity Exceeded` ou `Limit Exceeded`, o CloudFront retornará um código de status 503 ao visualizador sem usar sua página de erro personalizada.
- Se você criou uma página de erro personalizada, o CloudFront exibirá `Connection: close` ou `Connection: keep-alive` para os seguintes códigos de resposta:
 - O CloudFront exibe `Connection: close` para os códigos de status 400, 405, 414, 416, 500 e 501
 - O CloudFront exibe `Connection: keep-alive` para os códigos de status 403, 404, 502, 503 e 504.

Para obter uma explicação detalhada de como o CloudFront lida com respostas de erro de sua origem, consulte [Como o CloudFront processa códigos de status HTTP 4xx e 5xx da origem](#).

Armazenar objetos e páginas de erro personalizadas em diferentes locais

Se você quiser armazenar seus objetos e páginas de erro personalizadas em locais diferentes, sua distribuição deverá incluir um comportamento de cache para o qual o seguinte é verdadeiro:

- O valor de Path Pattern é correspondente às suas mensagens de erro personalizadas. Por exemplo, imagine que você salvou páginas de erro personalizadas para erros 4xx em um bucket do Amazon S3 em um diretório denominado `/4xx-errors`. Sua distribuição deverá incluir um comportamento de cache para o qual o padrão de caminho roteia solicitações de suas páginas de erro personalizadas para esse local, por exemplo, `/4xx-errors/*`.
- O valor de Origin especifica o valor de Origin ID da origem que contém suas páginas de erro personalizadas.

Para ter mais informações, consulte [Configurações de comportamento de cache](#).

Alterar códigos de resposta exibidos pelo CloudFront

Você pode configurar o CloudFront para retornar um código de status HTTP diferente para o visualizador do que o CloudFront recebeu da origem. Por exemplo, se a origem retornar um código de status 500 para o CloudFront, você poderá solicitar que o CloudFront retorne uma página de erro personalizada e um código de status 200 (OK) ao visualizador. Há vários motivos pelos quais você pode querer que o CloudFront retorne um código de status ao visualizador diferente daquele retornado por sua origem ao CloudFront:

- Alguns dispositivos de internet (alguns firewalls e proxies corporativos, por exemplo) interceptam códigos de status HTTP 4xx e 5xx e impedem que a resposta seja retornada ao visualizador. Neste caso, se você substituir `200`, a resposta não será interceptada.
- Se não for necessário distinguir diferentes erros de cliente ou erros de servidor, você pode especificar `400` ou `500` como o valor que o CloudFront retorna para todos os códigos de status 4xx ou 5xx.
- Você pode retornar um código de status `200` (OK) e um site estático para que seus clientes não saibam que seu site está inativo.

Se você habilitar [os logs padrão do CloudFront](#) e configurar o CloudFront para alterar o código de status HTTP na resposta, o valor da coluna `sc-status` nos logs conterá o código de status especificado. No entanto, o valor da coluna `x-edge-result-type` não é afetado. Ele contém o tipo de resultado da resposta da origem. Por exemplo, digamos que você configure o CloudFront

para retornar um código de status 200 ao visualizador quando a origem retornar 404 (Not Found) ao CloudFront. Quando a origem responde a uma solicitação com um código de status 404, o valor na coluna `sc-status` no log será 200, mas o valor na coluna `x-edge-result-type` será `Error`.

Você pode configurar o CloudFront para retornar qualquer um dos seguintes códigos de status HTTP junto com uma página de erro personalizada:

- 200
- 400, 403, 404, 405, 414, 416
- 500, 501, 502, 503, 504

Controlar por quanto tempo o CloudFront armazena erros em cache

Por padrão, o CloudFront armazena em cache as respostas durante 10 segundos. Por sua vez, o CloudFront envia a próxima solicitação do objeto à origem para ver se o problema que causou o erro foi resolvido e se o objeto solicitado está disponível.

É possível especificar a duração, o Error Caching Minimum TTL (TTL mínimo de armazenamento em cache do erro), de cada código de status 4xx e 5xx armazenado em cache pelo CloudFront. (Para ter mais informações, consulte [Códigos de status HTTP 4xx e 5xx armazenados em cache pelo CloudFront](#).) Quando você especificar uma duração, observe o seguinte:

- Se você especificar uma duração curta de armazenamento de erros em cache, o CloudFront encaminhará mais solicitações para a origem do que se você especificar um período maior. Para erros 5xx, isso pode agravar o problema que originalmente fez com que a origem retornasse um erro.
- Quando a origem retorna um erro para um objeto, o CloudFront responde às solicitações do objeto com a resposta de erro ou com a página de erro personalizada até que a duração do armazenamento em cache expire. Se você especificar uma duração longa para o armazenamento de erros em cache, o CloudFront poderá continuar a responder às solicitações com uma resposta de erro ou com a página de erro personalizada por um longo período após o objeto se tornar disponível novamente.


Note

É possível criar uma página de erro personalizada para o código de status HTTP 416 (intervalo solicitado insatisfatório) e alterar o código de status HTTP retornado pelo

CloudFront aos visualizadores quando a origem retornar um código de status 416 ao CloudFront. (Para obter mais informações, consulte [Alterar códigos de resposta exibidos pelo CloudFront](#).) No entanto, o CloudFront não armazena em cache as respostas do código de status 416, portanto, mesmo se você especificar um valor para o TTL mínimo de cache de erro para o código de status 416, o CloudFront não o usará.

Para controlar o tempo que o CloudFront armazena erros em cache para objetos individuais, configure seu servidor de origem para adicionar o cabeçalho aplicável à resposta de erro desse objeto.

Se a origem adicionar uma diretiva `Cache-Control: max-age` ou `Cache-Control: s-maxage`, ou um cabeçalho `Expires`, o CloudFront armazenará as respostas de erro em cache pelo valor no cabeçalho ou pelo valor do Error Caching Minimum TTL (TTL mínimo de armazenamento em cache de erros), o que for maior.

 Note

Os valores `Cache-Control: max-age` e `Cache-Control: s-maxage` não podem ser maiores que o valor de Maximum TTL (TTL máximo) definido para o comportamento de cache para o qual a página de erro está sendo buscada.

Se a origem adicionar outras diretivas `Cache-Control` ou nenhum cabeçalho, o CloudFront armazenará as respostas de erro em cache pelo valor de Error Caching Minimum TTL (TTL mínimo de armazenamento em cache de erros).

Se o tempo de expiração de um código de status 4xx ou 5xx para um objeto for maior do que você deseja e o objeto estiver disponível novamente, você poderá invalidar o código de erro armazenado em cache usando a URL do objeto solicitado. Se a origem estiver retornando uma resposta de erro para vários objetos, você precisará invalidar cada objeto separadamente. Para obter mais informações sobre como invalidar objetos, consulte [Invalidar arquivos para remover conteúdo](#).

Adicionar, remover ou substituir conteúdo distribuído pelo CloudFront

Esta seção explica como verificar se o CloudFront pode acessar o conteúdo que você deseja disponibilizar para seus visualizadores, como especificar os objetos no seu site ou aplicação e como remover ou substituir conteúdo.

Tópicos

- [Adicionar e acessar conteúdo distribuído pelo CloudFront](#)
- [Usar o versionamento de arquivos para atualizar ou remover conteúdo com uma distribuição do CloudFront](#)
- [Personalizar o formato do URL para arquivos no CloudFront](#)
- [Especificar um objeto raiz padrão](#)
- [Invalidar arquivos para remover conteúdo](#)
- [Fornecer arquivos compactados](#)

Adicionar e acessar conteúdo distribuído pelo CloudFront

Para que o CloudFront distribua conteúdo (objetos), adicione arquivos a uma das origens que você especificou para a distribuição e exponha um link do CloudFront para os arquivos. Um ponto de presença do CloudFront não busca os novos arquivos de uma origem enquanto não recebe solicitações do visualizador para eles. Para obter mais informações, consulte [Como o CloudFront entrega conteúdo](#).

Ao adicionar um arquivo a ser distribuído pelo CloudFront, adicione-o a um dos buckets do Amazon S3 especificados na distribuição ou, para uma origem personalizada, a um diretório no domínio especificado. Além disso, confirme se o padrão de caminho no comportamento de cache aplicável envia solicitações para a origem correta.

Por exemplo, imagine que o padrão de caminho de um comportamento de cache é *.html. Se você não tiver nenhum outro comportamento de cache configurado para encaminhar solicitações para essa origem, o CloudFront só encaminhará arquivos *.html. Neste cenário, por exemplo, o CloudFront nunca distribuirá arquivos .jpg que você enviou para a origem, porque não criou um comportamento de cache que inclui arquivos .jpg.

Os servidores do CloudFront não determinam o tipo MIME para os objetos fornecidos por eles. Quando você envia um arquivo para sua origem, recomendamos que você defina o campo de cabeçalho Content-Type para ela.

Usar o versionamento de arquivos para atualizar ou remover conteúdo com uma distribuição do CloudFront

Para atualizar o conteúdo existente que o CloudFront está configurado para distribuir, recomendamos que você use um identificador de versão em nomes de arquivos ou de pastas. Isso ajuda a dar a você controle sobre o gerenciamento do conteúdo que o CloudFront fornece.

Atualizar arquivos existentes usando nomes de arquivos com versão

Ao atualizar arquivos existentes em uma distribuição do CloudFront, recomendamos incluir algum tipo de identificador de versão nos nomes dos arquivos ou do diretório para ter um melhor controle do conteúdo. Esse identificador pode ser data e hora, um número sequencial ou outro método de distinção de duas versões do mesmo objeto.

Por exemplo, em vez de nomear um arquivo gráfico image.jpg, você pode denominá-lo image_1.jpg. Se você quiser começar a fornecer uma nova versão do arquivo, nomeie o novo arquivo image_2.jpg e atualize os links do aplicativo web ou site para apontar para image_2.jpg. Como alternativa, você pode colocar todos os gráficos em um diretório images_v1 e, quando quiser começar a fornecer novas versões de um ou mais gráficos, crie um novo diretório images_v2 e atualize os links para apontar para esse diretório. Com o versionamento, não é necessário esperar que um objeto expire para que o CloudFront comece a fornecer uma nova versão, nem pagar por invalidação de objeto.

Mesmo se você versionar seus arquivos, recomendamos que defina uma data de expiração.

Para obter mais informações, consulte [Gerenciar o tempo de permanência do conteúdo no cache \(expiração\)](#).

Note

A especificação de nomes de arquivos ou do diretório com versão não está relacionada ao versionamento de objetos do Amazon S3.

Remover conteúdo para que o CloudFront não o distribua

É possível remover arquivos da origem que você não deseja mais incluir na distribuição do CloudFront. No entanto, o CloudFront continuará exibindo aos visualizadores o conteúdo do cache de borda até que os arquivos expirem.

Se você quiser remover um arquivo imediatamente, faça o seguinte:

- Use o versionamento de arquivos. Ao usar o versionamento, as versões diferentes de um arquivo têm nomes diferentes que você pode usar na distribuição do CloudFront, para alterar o arquivo que será retornado aos visualizadores. Para ter mais informações, consulte [Atualizar arquivos existentes usando nomes de arquivos com versão](#).
- Invalide o arquivo. Para ter mais informações, consulte [Invalidar arquivos para remover conteúdo](#).

Personalizar o formato do URL para arquivos no CloudFront

Depois de configurar a origem com os objetos (conteúdo) que você deseja que o CloudFront forneça a seus visualizadores, use os URLs corretos para fazer referência a esses objetos no site ou no código da aplicação para que o CloudFront possa fornecê-los.

O nome de domínio usado nos URLs para objetos nas páginas da web ou no aplicativo web pode ser um dos seguintes:

- O nome de domínio, como `d111111abcdef8.cloudfront.net`, que o CloudFront atribui automaticamente ao criar uma distribuição
- O seu próprio nome de domínio, como `example.com`

Por exemplo, use um dos seguintes URLs para retornar o arquivo `image.jpg`:

```
https://d111111abcdef8.cloudfront.net/images/image.jpg
```

```
https://example.com/images/image.jpg
```

Você usa o mesmo formato de URL para armazenar o conteúdo em buckets do Amazon S3 ou em uma origem personalizada, como um de seus próprios servidores Web.

Note

O formato do URL depende, em parte, do valor especificado para Caminho de origem na distribuição. Esse valor fornece ao CloudFront um caminho de diretório superior para os seus objetos. Para mais informações sobre a configuração do caminho de origem ao criar uma distribuição, consulte [Caminho de origem](#).

Para obter mais informações sobre formatos de URL, consulte as seções a seguir.

Usar seu próprio nome de domínio (example.com)

Em vez de usar o nome de domínio padrão que o CloudFront atribui para você ao criar uma distribuição, você pode [adicionar um nome de domínio alternativo](#) que facilite o trabalho, como `example.com`. Ao configurar o seu próprio nome de domínio com o CloudFront, use um URL como este para objetos na distribuição:

```
https://example.com/images/image.jpg
```

Se você planejar usar HTTPS entre os visualizadores e o CloudFront, consulte [Usar nomes de domínio alternativos e HTTPS](#).

Usar uma barra (/) no final de URLs

Ao especificar URLs para diretórios na distribuição do CloudFront, escolha sempre usar ou nunca usar uma barra no final. Por exemplo, escolha apenas um dos seguintes formatos para todos os URLs:

```
https://d111111abcdef8.cloudfront.net/images/
```

```
https://d111111abcdef8.cloudfront.net/images
```

Por que é importante?

Os dois formatos funcionam para vinculação aos objetos do CloudFront, mas ser consistente pode ajudar a evitar problemas quando você desejar invalidar um diretório posteriormente. O CloudFront armazena os URLs exatamente como eles são definidos, incluindo as barras no final. Portanto, se o formato for inconsistente, será necessário invalidar os URLs de diretório com e sem a barra, a fim de garantir que o CloudFront remova o diretório.

É inconveniente ter que invalidar ambos os formatos de URLs, podendo gerar custos adicionais. Isso ocorre porque se você precisar duplicar invalidações para cobrir os dois tipos de URLs, poderá exceder o número máximo de invalidações livres permitidas para o mês. E se isso acontecer, será necessário pagar por todas as invalidações, mesmo que exista apenas um formato para cada URL de diretório no CloudFront.

Criar URLs assinados para conteúdo restrito

Se possuir conteúdo para o qual deseja restringir o acesso, crie URLs assinados. Por exemplo, se deseja distribuir o conteúdo apenas para os usuários autenticados, crie URLs válidos somente durante um período específico ou disponíveis somente a partir de um endereço IP específico. Para ter mais informações, consulte [Veicular conteúdo privado com URLs e cookies assinados](#).

Especificar um objeto raiz padrão

É possível configurar o CloudFront para retornar um objeto específico (o objeto raiz padrão) quando um usuário solicita o URL raiz da sua distribuição ao invés de solicitar um objeto em sua distribuição. A especificação de um objeto raiz padrão evita a exposição do conteúdo da sua distribuição.

Tópicos

- [Como especificar um objeto raiz padrão](#)
- [Como funciona o objeto raiz padrão](#)
- [Como funciona o CloudFront se você não define um objeto raiz](#)

Como especificar um objeto raiz padrão

Para evitar a exposição do conteúdo de sua distribuição ou a devolução de um erro, execute as etapas a seguir para especificar um objeto raiz padrão para sua distribuição.

Para especificar um objeto raiz padrão para sua distribuições

1. Faça upload do objeto raiz padrão na origem à qual sua distribuição aponta.

O arquivo pode ser qualquer tipo compatível com o CloudFront. Para obter uma lista de restrições do nome do arquivo, consulte a descrição do elemento `DefaultRootObject` em [DistributionConfig](#).

Note

Se o nome do arquivo do objeto raiz padrão for muito longo ou contiver um caractere inválido, o CloudFront retornará o erro HTTP 400 Bad Request - InvalidDefaultRootObject. Além disso, o CloudFront armazena o código em cache por 10 segundos (por padrão) e grava os resultados nos logs de acesso.

2. Confirme se as permissões do objeto concedem pelo menos acesso `read` ao CloudFront.

Para obter mais informações sobre permissões do Amazon S3, consulte [Identity and Access Management no Amazon S3](#) no Guia do usuário do Amazon Simple Storage Service.

3. Atualize a distribuição para fazer referência ao objeto raiz padrão usando o console ou a API do CloudFront.

Para especificar um objeto raiz padrão usando o console do CloudFront:

- a. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
- b. Na lista de distribuições no painel superior, selecione a distribuição a ser atualizada.
- c. No painel Settings (Configurações), na guia General (Geral), escolha Edit (Editar).
- d. Na caixa de diálogo Edit settings (Editar configurações), no campo Default root object (Objeto raiz padrão), insira o nome do arquivo do objeto raiz padrão.

Insira somente o nome do objeto, por exemplo, `index.html`. Não adicione `/` antes do nome do objeto.

- e. Escolha Salvar alterações.

Para atualizar a configuração usando a API do CloudFront, especifique um valor para o elemento `DefaultRootObject` na distribuição. Para obter informações sobre o uso da API do CloudFront para especificar um objeto raiz padrão, consulte [UpdateDistribution](#) na Referência da API do Amazon CloudFront.

4. Confirme se você ativou o objeto raiz padrão ao solicitar o URL raiz. Se o navegador não exibir o objeto raiz padrão, execute as seguintes etapas:
 - a. Confirme se a distribuição está totalmente implantada consultando o status dela no console do CloudFront.

- b. Repita as etapas 2 e 3 para verificar se foram concedidas as permissões corretas e se a configuração da distribuição foi atualizada corretamente para especificar o objeto raiz padrão.

Como funciona o objeto raiz padrão

Suponhamos, os seguintes pontos de solicitação do objeto `image.jpg`:

```
https://d111111abcdef8.cloudfront.net/image.jpg
```

Em contrapartida, os seguintes pontos de solicitação do URL raiz da mesma distribuição, em vez de um objeto específico, como no primeiro exemplo:

```
https://d111111abcdef8.cloudfront.net/
```

Ao definir um objeto raiz padrão, uma solicitação do usuário chamando a raiz da distribuição retornará o objeto raiz padrão. Por exemplo, se você designar o arquivo `index.html` como o objeto raiz padrão, uma solicitação de:

```
https://d111111abcdef8.cloudfront.net/
```

Retorna:

```
https://d111111abcdef8.cloudfront.net/index.html
```

Note

O CloudFront não determina se um URL com várias barras finais (`https://d111111abcdef8.cloudfront.net///`) é equivalente a `https://d111111abcdef8.cloudfront.net/`. O servidor de origem faz essa comparação.

Se você definir um objeto raiz padrão, uma solicitação do usuário de um subdiretório da distribuição não retornará o objeto raiz padrão. Por exemplo, suponha que `index.html` seja seu objeto raiz padrão, e o CloudFront receba uma solicitação de usuário final do diretório `install` da distribuição do CloudFront:

```
https://d111111abcdef8.cloudfront.net/install/
```

O CloudFront não retornará o objeto raiz padrão, mesmo que uma cópia de `index.html` apareça no diretório `install`.

Se você configurar a distribuição para permitir todos os métodos HTTP compatíveis com o CloudFront, o objeto raiz padrão se aplicará a todos os métodos. Por exemplo, se o objeto raiz padrão for `index.php` e você escrever a aplicação para enviar uma solicitação POST à raiz do domínio (`https://example.com`), o CloudFront enviará a solicitação para `https://example.com/index.php`.

O comportamento dos objetos raiz padrão do CloudFront é diferente do comportamento dos documentos de índice do Amazon S3. Ao configurar um bucket do Amazon S3 como um site e especificar o documento de índice, o Amazon S3 retornará o documento de índice mesmo que o usuário solicite um subdiretório no bucket. (Uma cópia do documento de índice deve ser exibida em cada subdiretório.) Para obter mais informações sobre como configurar buckets do Amazon S3 como sites e sobre documentos de índice, consulte o capítulo [Hospedar sites no Amazon S3](#) no Manual do usuário do Amazon Simple Storage Service.

Important

Lembre-se de que um objeto raiz padrão se aplica apenas à sua distribuição do CloudFront. É necessário gerenciar a segurança da sua origem. Por exemplo, se você estiver usando uma origem do Amazon S3, ainda precisará definir as ACLs do bucket do Amazon S3 adequadamente para garantir o nível de acesso desejado no bucket.

Como funciona o CloudFront se você não define um objeto raiz

Se você não definir um objeto raiz padrão, as solicitações da raiz da distribuição passarão para o servidor de origem. Se estiver usando uma origem do Amazon S3, qualquer um dos seguintes poderá ser retornado:

- Uma lista do conteúdo do seu bucket do Amazon S3: em qualquer uma destas condições, o conteúdo da origem ficará visível para qualquer pessoa que usar o CloudFront para acessar sua distribuição:
 - Seu bucket não está configurado corretamente.
 - As permissões do Amazon S3 no bucket associado à distribuição e nos objetos do bucket concedem acesso a todos.
 - Um usuário final acessa sua origem usando o URL raiz dela.
- Uma lista de conteúdo privado da sua origem: se você configurar a origem como uma distribuição privada (somente você e o CloudFront têm acesso), o conteúdo do bucket do Amazon S3 associado à distribuição ficará visível para qualquer pessoa que tiver as credenciais para acessar

sua distribuição por meio do CloudFront. Nesse caso, os usuários não podem acessar seu conteúdo pelo URL raiz da origem. Para obter mais informações sobre distribuição de conteúdo privado, consulte [the section called “Restringir conteúdo com URLs e cookies assinados”](#).

- **Error 403 Forbidden:** o CloudFront retornará esse erro se as permissões do bucket do Amazon S3 associado à sua distribuição ou as permissões dos objetos desse bucket negarem acesso ao CloudFront e a todos.

Invalidar arquivos para remover conteúdo

Se você precisar remover um arquivo de caches de borda do CloudFront antes de ele expirar, poderá executar uma das seguintes ações:

- Invalidar o arquivo dos pontos de presença de caches. Na próxima vez que um visualizador solicitar o arquivo, o CloudFront recorrerá à origem para obter a versão mais recente do arquivo.
- Usar o versionamento de arquivos para fornecer uma versão diferente do arquivo com um nome diferente. Para ter mais informações, consulte [Atualizar arquivos existentes usando nomes de arquivos com versão](#).

Tópicos

- [Escolher entre invalidar arquivos e usar nomes de arquivos com versionamento](#)
- [Determinar quais arquivos invalidar](#)
- [O que você precisa saber ao invalidar arquivos](#)
- [Invalidar arquivos](#)
- [Máximo de solicitações de invalidação simultâneas](#)
- [Pagar pela invalidação de arquivos](#)

Escolher entre invalidar arquivos e usar nomes de arquivos com versionamento

Para controlar as versões de arquivos fornecidos da distribuição, você pode invalidá-los ou dar a eles nomes de arquivo com versão. Se quiser atualizar os arquivos com frequência, recomendamos dar preferência ao versionamento de arquivos, pelos seguintes motivos:

- O versionamento possibilita controlar o arquivo retornado por uma solicitação mesmo quando o usuário tem uma versão armazenada em cache localmente ou atrás de um proxy de armazenamento em cache corporativo. Se você invalidar o arquivo, o usuário continuará vendo a versão antiga até ela expirar desses caches.
- Os logs de acesso do CloudFront incluem os nomes dos arquivos, portanto, o versionamento facilita a análise dos resultados de alterações nos arquivos.
- O versionamento é uma maneira de fornecer diferentes versões de arquivos para diferentes usuários.
- O versionamento simplifica o uso de versões mais antigas e mais recentes de revisões de arquivo.
- O versionamento é menos caro. Você precisa pagar para o CloudFront transferir novas versões de seus arquivos para pontos de presença, mas não pela invalidação de arquivos.

Para obter mais informações sobre versionamento de arquivo, consulte [Atualizar arquivos existentes usando nomes de arquivos com versão](#).

Determinar quais arquivos invalidar

Se você quiser invalidar vários arquivos, como todos os arquivos de um diretório ou cujos nomes começam com os mesmos caracteres, inclua o curinga * no final do caminho de invalidação. Para obter mais informações sobre como usar o curinga *, consulte [Invalidation paths](#).

Para invalidar arquivo, você pode especificar o caminho de cada arquivo ou um caminho que termine com o curinga *, que pode se aplicar a um ou vários arquivos, como mostrado nestes exemplos:

- /images/image1.jpg
- /images/image*
- /images/

Para invalidar arquivos selecionados, mas os usuários não necessariamente acessam todos os arquivos na origem, determine quais arquivos os visualizadores solicitaram do CloudFront e invalide-os. Para determinar quais arquivos os visualizadores solicitaram, habilite o registro em log de acesso do CloudFront. Para obter mais informações sobre os logs de acesso, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

O que você precisa saber ao invalidar arquivos

Ao especificar um arquivo para invalidar, consulte as seguintes informações:

Diferenciação de letras maiúsculas e minúsculas

Os caminhos de invalidação diferenciam letras maiúsculas de minúsculas. Por exemplo, `/images/image.jpg` e `/images/Image.jpg` especificam dois arquivos diferentes.

Alterar o URI usando uma função do Lambda

Se a distribuição do CloudFront acionar uma função do Lambda nos eventos de solicitação do visualizador, e se a função alterar o URI do arquivo solicitado, recomendamos invalidar os dois URIs para remover o arquivo dos caches de borda do CloudFront:

- O URI na solicitação do visualizador
- O URI após a alteração pela função

Example Exemplo

Suponha que a função do Lambda altere o URI de um arquivo de:

```
https://d111111abcdef8.cloudfront.net/index.html
```

Para um URI que inclui um diretório de linguagem:

```
https://d111111abcdef8.cloudfront.net/en/index.html
```

Para invalidar o arquivo, é necessário especificar os seguintes caminhos:

- `/index.html`
- `/en/index.html`

Para obter mais informações, consulte [Invalidation paths](#).

Objeto raiz padrão

Para invalidar o objeto raiz padrão (arquivo), especifique o caminho da mesma forma que especificaria o caminho de qualquer outro arquivo. Para ter mais informações, consulte [Como funciona o objeto raiz padrão](#).

Encaminhamento de cookies

Se você configurou o CloudFront para encaminhar cookies para a origem, os caches de borda do CloudFront poderão conter várias versões do arquivo. Ao invalidar um arquivo, o CloudFront invalida cada versão do arquivo armazenada em cache, independentemente de seus cookies associados. Você não pode invalidar seletivamente algumas versões, mas não outras, com base nos cookies associados. Para obter mais informações, consulte [Conteúdo em cache com base em cookies](#).

Encaminhamento de cabeçalhos

Se você configurar o CloudFront para encaminhar uma lista de cabeçalhos para a origem e armazenar em cache com base nos valores dos cabeçalhos, os caches de borda do CloudFront poderão conter várias versões do arquivo. Ao invalidar um arquivo, o CloudFront invalida cada versão do arquivo armazenada em cache, independentemente dos valores de cabeçalho. Você não pode invalidar seletivamente algumas versões, mas não outras, com base nos valores de cabeçalho. (Se você configurar o CloudFront para encaminhar todos os cabeçalhos para a origem, o CloudFront não armazenará seus arquivos em cache.) Para obter mais informações, consulte [Conteúdo em cache com base nos cabeçalhos de solicitação](#).

Encaminhamento de query strings

Se você configurar o CloudFront para encaminhar strings de consulta para a origem, deverá incluí-las ao invalidar arquivos, como mostrado nestes exemplos:

- `/images/image.jpg?parameter1=a`
- `/images/image.jpg?parameter1=b`

Se as solicitações do cliente incluírem cinco query strings diferentes para o mesmo arquivo, você poderá invalidar o arquivo cinco vezes, uma vez para cada query string, ou usar o curinga "*" no caminho de invalidação, como mostrado neste exemplo:

```
/images/image.jpg*
```

Para obter mais informações sobre como usar curingas no caminho de invalidação, consulte [Invalidation paths](#).

Para obter mais informações sobre query strings, consulte [Conteúdo em cache com base em parâmetros de string de consulta](#).

Para determinar quais strings de consulta estão em uso, habilite o registro em log no CloudFront. Para obter mais informações, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Máximo permitido

Consulte mais informações sobre o número máximo de invalidações permitidas em [Máximo de solicitações de invalidação simultâneas](#).

Arquivos do Microsoft Smooth Streaming

Não será possível invalidar arquivos de mídia no formato Microsoft Smooth Streaming se o Smooth Streaming estiver habilitado para o comportamento de cache correspondente.

Caracteres não ASCII ou não seguros no caminho

Se o caminho incluir caracteres não ASCII ou não seguros, conforme definido na [RFC 1738](#), codifique-os por URL. Não codifique em URL outros caracteres do caminho; caso contrário, o CloudFront não invalidará a versão antiga do arquivo atualizado.

Caminhos de invalidação

O caminho é relativo à distribuição. Por exemplo, para invalidar o arquivo em `https://d111111abcdef8.cloudfront.net/images/image2.jpg`, seria necessário especificar `/images/image2.jpg`.

Note

No [Console do CloudFront](#), é possível omitir a barra à esquerda no caminho, da seguinte forma: `images/image2.jpg`. Ao usar a API do CloudFront diretamente, os caminhos de invalidação devem começar com uma barra à esquerda.

Você também pode invalidar vários arquivos simultaneamente usando o curinga `*`. O `*`, que substitui 0 ou mais caracteres, deve ser o último caractere no caminho de invalidação.

Important

Para usar curingas (`*`) na invalidação, é necessário colocá-los no final do caminho. Os asteriscos (`*`) inseridos em qualquer outro lugar são tratados como uma correspondência literal de caracteres em vez de uma invalidação de curingas.

Se você usar a AWS Command Line Interface (AWS CLI) para invalidar arquivos e especificar um caminho que inclua o curinga `*`, será necessário colocar o caminho entre aspas (`"`), como `"/**"`.

O tamanho máximo de um caminho é 4.000 caracteres.

Example Exemplo: caminhos de invalidação

- Para invalidar todos os arquivos em um diretório:

`/directory-path/*`

- Para invalidar um diretório, todos os respectivos subdiretórios e todos os arquivos no diretório e nos subdiretórios:

*/directory-path**

- Para invalidar todos os arquivos com o mesmo nome, mas com diferentes extensões de nome de arquivo, como logo.jpg, logo.png e logo.gif:

*/directory-path/file-name.**

- Para invalidar todos os arquivos de um diretório com nome de arquivo começando com os mesmos caracteres (como todos os arquivos de um vídeo no formato HLS), independentemente da extensão do nome do arquivo:

*/directory-path/initial-characters-in-file-name**

- Ao configurar o CloudFront para armazenamento em cache com base nos parâmetros da string de consulta e você quiser invalidar todas as versões de um arquivo:

*/directory-path/file-name.file-name-extension**

- Para invalidar todos os arquivos em uma distribuição:

*/**

Para obter informações sobre a invalidação de arquivos ao usar uma função do Lambda para alterar o URI, consulte [Changing the URI Using a Lambda Function](#).

Se o caminho de invalidação for um diretório e você não padronizou um método para especificar os diretórios (com ou sem uma barra (/) no final), recomendamos invalidar o diretório com e sem a barra no final, por exemplo, /images e /images/.

Signed URLs

Se você estiver usando URLs assinados, invalide um arquivo incluindo apenas a parte do URL antes do ponto de interrogação (?).

Invaldar arquivos

É possível usar o console do CloudFront para criar e executar uma invalidação, exibir uma lista das invalidações enviadas anteriormente e exibir informações detalhadas sobre uma invalidação específica. Você também pode copiar uma invalidação existente, editar a lista de caminhos de arquivos e executar a invalidação editada. Você não pode remover invalidações da lista.

Sumário

- [Invalidar arquivos](#)
- [Copiar, editar e executar novamente uma invalidação existente](#)
- [Cancelar invalidações](#)
- [Listar invalidações](#)
- [Exibir informações sobre uma invalidação](#)


Invalidar arquivos

Para invalidar arquivos usando o console do CloudFront, siga o procedimento a seguir.

Console

Como invalidar arquivos (console)

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha a distribuição para a qual você deseja invalidar arquivos.
3. Escolha a guia Invalidations.
4. Escolha Criar invalidação.
5. Para os arquivos que você deseja invalidar, insira um caminho de invalidação por linha. Para obter informações sobre como especificar caminhos de invalidação, consulte [O que você precisa saber ao invalidar arquivos](#).

 Important

Especifique cuidadosamente os caminhos de arquivos. Você não pode cancelar uma solicitação de invalidação depois de iniciá-la.


6. Escolha Criar invalidação.

CloudFront API

Para saber mais sobre como invalidar objetos e exibir informações sobre invalidações, consulte os seguintes tópicos da Referência da API do Amazon CloudFront:

- [CreateInvalidation](#)

- [ListInvalidations](#)
- [GetInvalidation](#)


 Note

Se você usar a AWS Command Line Interface (AWS CLI) para invalidar arquivos e especificar um caminho que inclua o curinga *, será necessário colocar o caminho entre aspas ("), como o seguinte exemplo:

```
aws cloudfront create-invalidation --distribution-id distribution_ID --paths  
"/*"
```

Copiar, editar e executar novamente uma invalidação existente

Você pode copiar uma invalidação criada anteriormente, atualize a lista de caminhos de invalidação e executar a invalidação atualizada. Não é possível copiar uma invalidação existente, atualizar os caminhos de invalidação e salvar a invalidação atualizada sem a executar.

 Important

Se você copiar uma invalidação que ainda está em andamento, atualizar a lista de caminhos de invalidação e executar a invalidação atualizada, o CloudFront não interromperá nem excluirá a invalidação copiada. Se qualquer caminho de invalidação for exibido no original e na cópia, o CloudFront tentará invalidar os arquivos duas vezes, e as duas invalidações serão contabilizadas no número máximo de invalidações gratuitas para o mês. Se você já tiver atingido o número máximo de invalidações gratuitas, as invalidações de cada arquivo serão cobradas. Para ter mais informações, consulte [Máximo de solicitações de invalidação simultâneas](#).

Para copiar, editar e reexecutar uma invalidação existente

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Selecione a distribuição que contém a invalidação que deseja copiar.
3. Escolha a guia Invalidations.

4. Escolha a invalidação que você deseja copiar.

Se não tiver certeza de qual invalidação deseja copiar, escolha uma invalidação e selecione Visualizar detalhes para exibir informações detalhadas sobre ela.

5. Escolha Copiar para novo.
6. Atualize a lista de caminhos de invalidação, se aplicável.
7. Escolha Criar invalidação.

Cancelar invalidações

Ao enviar uma solicitação de invalidação ao CloudFront, o CloudFront encaminhará a solicitação a todos os pontos de presença em alguns segundos, e cada ponto de presença iniciará o processamento da invalidação imediatamente. Consequentemente, não é possível cancelar uma invalidação após o envio.

Listar invalidações

É possível exibir uma lista das últimas 100 invalidações criadas e executadas para uma distribuição usando o console do CloudFront. Se quiser obter uma lista com mais de 100 invalidações, use a operação de API `ListInvalidations`. Para obter mais informações, consulte [ListInvalidations](#) na Referência da API do Amazon CloudFront.

Para listar invalidações

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Selecione a distribuição para a qual deseja exibir uma lista de invalidações.
3. Escolha a guia Invalidations.

Note

Você não pode remover invalidações da lista.

Exibir informações sobre uma invalidação

Você pode exibir informações detalhadas sobre uma invalidação, inclusive o ID da distribuição, ID da invalidação, status da invalidação, data e hora de criação da invalidação e uma lista completa dos caminhos de invalidação.

Para exibir informações sobre uma invalidação

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Selecione a distribuição que contém a invalidação cujas informações detalhadas você deseja exibir.
3. Escolha a guia Invalidations.
4. Escolha o ID de invalidação aplicável ou o ID de invalidação e selecione Visualizar detalhes.

Máximo de solicitações de invalidação simultâneas

Se você estiver invalidando arquivos individualmente, poderá ter solicitações de invalidação de até 3.000 arquivos por distribuição em andamento ao mesmo tempo. Isso pode ser uma solicitação de invalidação para até 3.000 arquivos, até 3.000 solicitações para um arquivo ou qualquer outra combinação que não ultrapasse 3.000 arquivos. Por exemplo, você pode enviar 30 solicitações de invalidação de 100 arquivos cada. Enquanto as 30 solicitações de invalidação estiverem em andamento, não será possível enviar mais solicitações de invalidação. Se você exceder o número máximo, o CloudFront retornará uma mensagem de erro.

Se estiver usando o curinga “*”, poderá ter solicitações de até 15 caminhos de invalidação em andamento ao mesmo tempo. Você também pode ter solicitações de invalidação de até 3.000 arquivos individuais por distribuição em andamento ao mesmo tempo. O máximo de solicitações de invalidação de curinga não depende do máximo de arquivos de invalidação individuais.

Pagar pela invalidação de arquivos

Os 1.000 primeiros caminhos de invalidação enviados por mês são gratuitos. Você paga por caminho de invalidação que ultrapassar esse número em um mês. Um caminho de invalidação que pode ser para um único arquivo (como `/images/logo.jpg`) ou para vários arquivos (como `/images/*`). Um caminho que inclui o curinga * conta como um caminho, mesmo que ele faça com que o CloudFront invalide milhares de arquivos.

O máximo de 1.000 caminhos de invalidação gratuitos por mês se aplica ao número total de caminhos de invalidação em todas as distribuições criadas com uma conta da AWS. Por exemplo, se você usar a Conta da AWS `john@example.com` para criar três distribuições e enviar 600 caminhos de invalidação para cada distribuição em um mês (para um total de 1.800 caminhos de invalidação), a AWS cobrará 800 caminhos de invalidação nesse mês.

A cobrança para enviar um caminho de invalidação é a mesma, independentemente do número de arquivos sendo invalidados: um único arquivo (`/images/logo.jpg`) ou todos os arquivos associados a uma distribuição (`/*`). Como a cobrança é feita por caminho na solicitação de invalidação, mesmo que você agrupe vários caminhos em uma única solicitação, cada um deles ainda assim será contabilizado individualmente para fins de cobrança.

Consulte mais informações sobre o preço de invalidação em [Preço do Amazon CloudFront](#). Para obter mais informações sobre os caminhos de invalidação, consulte [Invalidation paths](#).

Fornecer arquivos compactados

Será possível usar o CloudFront para compactar automaticamente determinados tipos de objetos (arquivos) e fornecer os objetos compactados quando os visualizadores (navegadores da Web ou outros clientes) forem compatíveis com eles. Os visualizadores indicam sua compatibilidade com esses objetos compactados com o cabeçalho HTTP `Accept-Encoding`.

O CloudFront pode compactar objetos usando os formatos de compactação Gzip e Brotli. Quando o visualizador é compatível com os dois formatos e ambos estão presentes no servidor de cache acessado, o CloudFront prefere o Brotli. Se somente um formato de compactação estiver presente no servidor de cache, o CloudFront o retornará.

Se houver solicitações subsequentes do visualizador para o mesmo objeto, o CloudFront exibirá a primeira versão armazenada em cache. Por exemplo, se um visualizador solicitar um objeto específico que use a compactação Gzip, o objeto estiver armazenado em cache e o visualizador aceitar o formato Gzip, as solicitações subsequentes para o mesmo objeto sempre exibirão a versão Gzip, mesmo que o visualizador aceite Brotli e Gzip.

Note

Os navegadores da Web Chrome e Firefox são compatíveis com a compactação Brotli somente quando a solicitação é enviada usando HTTPS. Esses navegadores não são compatíveis com o Brotli com solicitações HTTP.

Quando os objetos solicitados são compactados, os downloads podem ficar mais rápidos, porque os objetos são menores, chegando a menos de um quarto do tamanho do original em alguns casos. Especialmente em arquivos JavaScript e CSS, downloads mais rápidos podem resultar em uma renderização de páginas da Web mais rápida para seus usuários. Além disso, como o custo da transferência de dados do CloudFront é baseado na quantidade total de dados fornecidos, o fornecimento de objetos compactados pode ser mais barato que fornecê-los não compactados.

Algumas origens personalizadas também podem compactar objetos. A origem pode ser capaz de compactar objetos que o CloudFront não compacta (consulte [Tipos de arquivos compactados pelo CloudFront](#)). Se a origem retornar um arquivo compactado ao CloudFront, o CloudFront detectará que o objeto está compactado com base na presença do cabeçalho Content-Encoding e não compactará o objeto novamente.

Configurar o CloudFront para compactar objetos

Para configurar o CloudFront para compactar objetos, atualize o comportamento de cache ao qual você quer fornecer os objetos compactados realizando todas estas etapas:

1. Verifique se a configuração Compress objects automatically (Compactar objetos automaticamente) está marcada com Yes (Sim). (No AWS CloudFormation ou na API do CloudFront, defina Compress como true.)
2. Use uma [política de cache](#) para especificar configurações de armazenamento em cache e verifique se as configurações Gzip e Brotli estão habilitadas. (No AWS CloudFormation ou na API do CloudFront, defina EnableAcceptEncodingGzip e EnableAcceptEncodingBrotli como true.)
3. Certifique-se de que os valores TTL na política de cache estão definidos para um valor maior que zero. Ao definir os valores de TTL como zero, o armazenamento em cache do conteúdo compactado é desabilitado.

Para atualizar um comportamento de cache, é possível usar qualquer uma das seguintes ferramentas:

- O [Console do CloudFront](#)
- [AWS CloudFormation](#)
- Os [SDKs da AWS e as ferramentas de linha de comando](#)

Como a compactação do CloudFront funciona

Ao configurar o CloudFront para compactar objetos (consulte a seção anterior), é assim que ele funciona:

1. Um visualizador solicita um objeto. O visualizador inclui o cabeçalho HTTP `Accept-Encoding` na solicitação, e o valor de cabeçalho inclui `gzip`, `br` ou ambos. Isso indica que o visualizador é compatível com os objetos compactados. Quando o visualizador for compatível tanto com o Gzip quanto com o Brotli, o CloudFront preferirá o Brotli.

Note

Os navegadores da Web Chrome e Firefox são compatíveis com a compactação Brotli somente quando a solicitação é enviada usando HTTPS. Esses navegadores não são compatíveis com o Brotli com solicitações HTTP.

2. No local da borda, o CloudFront verifica se o cache tem uma cópia compactada do objeto solicitado.
3. Se o objeto compactado já estiver no cache, o CloudFront o enviará para o visualizador e ignorará as etapas restantes.

Se o objeto compactado não estiver no cache, o CloudFront encaminhará a solicitação para a origem.

Note

Se uma cópia descompactada do objeto já estiver no cache, o CloudFront poderá enviá-la para o visualizador sem encaminhar a solicitação para a origem. Por exemplo, isso pode acontecer quando o CloudFront [ignorou a compactação anteriormente](#). Quando isso acontece, o CloudFront armazena em cache o objeto não compactado e continua a fornecê-lo até o objeto expirar, ser despejado ou ser invalidado.

4. Se a origem retornar um objeto compactado, conforme indicado pela presença de um cabeçalho `Content-Encoding` na resposta HTTP, o CloudFront enviará o objeto compactado ao visualizador, o adicionará ao cache e ignorará a etapa restante. O CloudFront não compacta o objeto novamente.

Se a origem retornar um objeto descompactado para o CloudFront (não há cabeçalho Content-Encoding na resposta HTTP), o CloudFront determinará se é possível compactar o objeto. Para obter mais informações sobre como o CloudFront determina se é possível compactar um objeto, consulte a seção a seguir.

5. Se for possível compactar o objeto, o CloudFront o compactará, o retornará para o visualizador e o adicionará ao cache. (Em casos raros, o CloudFront pode [ignorar a compactação](#) e enviar o objeto não compactado ao visualizador.)

Quando o CloudFront compacta objetos

A lista a seguir fornece mais informações sobre quando o CloudFront compacta objetos.

A solicitação usa HTTP 1.0

Se uma solicitação ao CloudFront usar HTTP 1.0, o CloudFront removerá o cabeçalho Accept-Encoding e não compactará o objeto na resposta.

Cabeçalho da solicitação **Accept-Encoding**

Se o cabeçalho Accept-Encoding estiver ausente da solicitação do visualizador ou se ele não contiver gzip ou br como valor, o CloudFront não compactará o objeto na resposta. Se o cabeçalho Accept-Encoding incluir valores adicionais, como deflate, o CloudFront os removerá antes de encaminhar a solicitação à origem.

Quando o CloudFront estiver [configurado para compactar objetos](#), ele incluirá o cabeçalho Accept-Encoding na chave de cache e nas solicitações de origem automaticamente.

Conteúdo dinâmico

O CloudFront nem sempre compacta conteúdo dinâmico. Às vezes, as respostas para conteúdo dinâmico são compactadas, outras vezes, não.

Quando você configura o CloudFront para compactar objetos, o conteúdo já é armazenado em cache

O CloudFront compacta objetos quando os obtém da origem. Ao configurar o CloudFront para compactar objetos, o CloudFront não compactará objetos que já estejam armazenados em cache em locais da borda. Além disso, quando um objeto em cache expira em um local da borda e o CloudFront encaminha outra solicitação do objeto à origem, o CloudFront não compactará o arquivo se a origem retornar um código de status HTTP 304, ou seja, o local da borda já tem a versão mais recente do arquivo. Para que o CloudFront compacte os objetos que já estão em

locais da borda, é necessário invalidar os objetos. Para obter mais informações, consulte [Invalidar arquivos para remover conteúdo](#).

A origem já está configurada para compactar objetos

Se você configurar o CloudFront para compactar objetos, e a origem também compactar os objetos, a origem deverá incluir um cabeçalho Content-Encoding, que indica que o objeto já está compactado. Quando a resposta de uma origem incluir o cabeçalho Content-Encoding, o CloudFront não compactará o objeto, qualquer que seja o valor do cabeçalho. O CloudFront envia a resposta ao visualizador e armazena o objeto em cache no local da borda.

Tipos de arquivos compactados pelo CloudFront

Para obter uma lista completa dos tipos de arquivos compactados pelo CloudFront, consulte [Tipos de arquivos compactados pelo CloudFront](#).

Tamanho dos objetos compactados pelo CloudFront

O CloudFront compacta objetos de 1.000 bytes a 10.000.000 bytes.

Content-LengthCabeçalho

A origem deve incluir um cabeçalho Content-Length na resposta, que o CloudFront usa para determinar se o tamanho do objeto está no intervalo de compactação do CloudFront. Caso o cabeçalho Content-Length esteja ausente, contenha um valor inválido ou contenha um valor fora do intervalo de tamanhos compactados pelo CloudFront, o CloudFront não compactará o objeto.

Código de status HTTP da resposta

O CloudFront compacta objetos somente quando o código de status HTTP da resposta é 200, 403 ou 404.

A resposta não tem corpo

Quando a resposta HTTP da origem não tiver corpo, não haverá nada para o CloudFront compactar.

ETagCabeçalho

O CloudFront às vezes modifica o cabeçalho ETag na resposta HTTP ao compactar objetos. Para ter mais informações, consulte [the section called “Conversão do cabeçalho ETag”](#).

O CloudFront ignora a compactação

O CloudFront compacta objetos na base do melhor esforço. Em casos raros, o CloudFront ignora a compactação. O CloudFront toma essa decisão com base em vários fatores, incluindo

a capacidade do host. Se o CloudFront ignorar a compactação de um objeto, ele armazenará o objeto não compactado em cache e continuará a fornecê-lo aos visualizadores até o objeto expirar, ser despejado ou ser invalidado.

Tipos de arquivos compactados pelo CloudFront

Se você configurar o CloudFront para compactar objetos, o CloudFront compactará os objetos que tiverem os seguintes valores no cabeçalho de resposta Content-Type:

- application/dash+xml
- application/eot
- application/font
- application/font-sfnt
- application/javascript
- application/json
- application/opentype
- application/otf
- application/pdf
- application/pkcs7-mime
- application/protobuf
- application/rss+xml
- application/truetype
- application/ttf
- application/vnd.apple.mpegurl
- application/vnd.mapbox-vector-tile
- application/vnd.ms-fontobject
- application/wasm
- application/xhtml+xml
- application/xml
- application/x-font-opentype
- application/x-font-truetype
- application/x-font-ttf

- application/x-httpd-cgi
- application/x-javascript
- application/x-mpegurl
- application/x-opentype
- application/x-otf
- application/x-perl
- application/x-ttf
- font/eot
- font/opentype
- font/otf
- font/ttf
- image/svg+xml
- text/css
- text/csv
- text/html
- text/javascript
- text/js
- text/plain
- text/richtext
- text/tab-separated-values
- text/xml
- text/x-component
- text/x-java-source
- text/x-script
- vnd.apple.mpegurl

Conversão do cabeçalho **ETag**

Quando o objeto não compactado da origem inclui um cabeçalho HTTP ETag válido e forte, e o CloudFront compacta o objeto, o CloudFront também converte o valor forte do cabeçalho ETag em um ETag fraco e retorna o valor fraco de ETag ao visualizador. Os visualizadores podem armazenar

o valor fraco de ETag e usá-lo para enviar solicitações condicionais com o cabeçalho HTTP If-None-Match. Isso permite que os visualizadores, o CloudFront e a origem tratem as versões compactadas e não compactadas de um objeto como semanticamente equivalentes, o que reduz a transferência de dados desnecessária.

Um valor de cabeçalho ETag válido e forte começa com um caractere de aspas duplas ("). Para converter o valor forte de ETag em um valor fraco, o CloudFront adiciona os caracteres W/ ao início do valor forte de ETag.

Quando o objeto da origem inclui um valor de cabeçalho ETag fraco (um valor que começa com os caracteres W/), o CloudFront não modifica esse valor e o retorna ao visualizador como o recebeu da origem.

Quando o objeto da origem inclui um valor de cabeçalho ETag inválido (o valor não começa com " ou com W/), o CloudFront remove o cabeçalho ETag e retorna o objeto ao visualizador sem o cabeçalho de resposta ETag.

Para mais informações, consulte as páginas nos documentos do MDN na Web:

- [Políticas](#) (cabeçalho HTTP ETag)
- [Validação fraca](#) (solicitações condicionais HTTP)
- [Cabeçalho HTTP If-None-Match](#)

Usar proteções do AWS WAF

Você pode usar o [AWS WAF](#) para proteger as distribuições do CloudFront e os servidores de origem. O AWS WAF é um firewall de aplicações da web que ajuda a proteger APIs e aplicações da web bloqueando solicitações antes que elas cheguem aos servidores. Para ter mais informações, consulte [Accelerate and protect your websites using CloudFront and AWS WAF](#) e [Guidelines for Implementing AWS WAF](#).

Para habilitar as proteções do AWS WAF:

- Use a proteção com um clique no console do CloudFront. A proteção com um clique cria uma lista de controle de acesso da web (ACL da web) do AWS WAF, configura regras para proteger os servidores de ameaças comuns da web e anexa a ACL da web à distribuição do CloudFront para você. Os tópicos desta seção pressupõem o uso de proteções com um clique.
- Use uma ACL (lista de controle de acesso) da web pré-configurada que você cria no console do AWS WAF ou usando as APIs do AWS WAF. Para obter mais informações, consulte [Web access control lists \(ACLs\)](#) no Guia do desenvolvedor do AWS WAF e [AssociateWebACL](#) na Referência de API do AWS WAF

É possível habilitar o AWS WAF ao:

- Criar uma distribuição
- Use o painel de Segurança para editar as configurações de segurança de uma distribuição existente

Quando você usa a proteção com um clique, o CloudFront aplica um conjunto de proteções recomendado pela AWS para:

- Bloquear endereços IP de possíveis ameaças com base na inteligência interna de ameaças da Amazon.
- Proteger contra as vulnerabilidades mais comuns encontradas em aplicações da web, conforme descrito no [OWASP Top 10](#).
- Defender contra agentes mal-intencionados que descobrem vulnerabilidades de aplicações.

⚠ Important

Você deverá habilitar o AWS WAF se quiser visualizar as métricas de segurança no painel de Segurança do CloudFront. Sem o AWS WAF habilitado, você só poderá usar o painel de Segurança para habilitar o AWS WAF ou configurar as restrições geográficas do CloudFront. Para obter mais informações sobre o painel, consulte [Gerenciar as proteções de segurança do AWS WAF no painel de segurança do CloudFront](#), posteriormente nessa seção.

Tópicos

- [Habilitar o AWS WAF para distribuições](#)
- [Gerenciar as proteções de segurança do AWS WAF no painel de segurança do CloudFront](#)
- [Configurar a limitação de taxa](#)
- [Desabilitar as proteções de segurança do AWS WAF](#)

Habilitar o AWS WAF para distribuições

É possível habilitar o AWS WAF ao criar uma distribuição ou habilitar as proteções de segurança para uma lista de controle de acesso (ACL) existente.

Se você habilitar AWS WAF para a distribuição do CloudFront, também será possível habilitar o controle de bots e configurar a proteção de segurança por categoria de bot.

Tópicos

- [Habilitar o AWS WAF para uma nova distribuição](#)
- [Usar uma ACL da web existente](#)
- [Habilitar o controle de bots](#)
- [Configurar a proteção por categoria de bot](#)

Habilitar o AWS WAF para uma nova distribuição

O procedimento a seguir mostra como habilitar o AWS WAF ao criar uma distribuição do CloudFront.

Como habilitar o AWS WAF para uma nova distribuição

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.

2. No painel de navegação, escolha Distribuições e Criar distribuição.
3. Conforme necessário, siga as etapas em [Criar uma distribuição](#).
4. Na seção Web Application Firewall, selecione Editar e Habilitar proteções de segurança.
5. Preencha os seguintes campos:
 - Usar o modo de monitor: habilite o modo de monitor quando quiser primeiro coletar dados para testar como a proteção funcionará. Ao habilitar o modo de monitor, as solicitações não serão bloqueadas se as proteções estiverem ativas. Em vez disso, o modo de monitor coleta dados sobre solicitações que seriam bloqueadas se as proteções estivessem ativas. Quando estiver tudo pronto para começar o bloqueio, você poderá habilitá-lo na página Segurança.
 - Proteções adicionais: escolha as opções que você deseja habilitar. Se você habilitar a limitação de taxa, consulte [the section called “Configurar a limitação de taxa”](#) para obter mais informações.
 - Estimativa de preço: é possível abrir a seção para exibir um campo no qual você insere um número diferente de solicitações/mês e vê uma nova estimativa.
6. Revise as configurações de distribuição restantes e escolha Criar distribuição.

Após a criação de uma distribuição, o CloudFront cria o painel Segurança. É possível usar esse painel para desabilitar ou habilitar o AWS WAF. Se você ainda não habilitou o AWS WAF, as tabelas e os elementos gráficos no painel permanecerão em branco.

Usar uma ACL da web existente

Se você tem uma ACL da Web, é possível usá-la em vez da proteção oferecida pelo AWS WAF.

Para usar uma configuração existente do AWS WAF

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Execute um destes procedimentos:
 - a. Escolha Criar distribuição, siga as etapas em [Criar uma distribuição](#) e retorne a este tópico.
 - b. Escolha uma configuração existente e selecione a guia Segurança.
3. Na seção Web Application Firewall (WAF), selecione Editar e Habilitar proteções de segurança.
4. Escolha Usar configuração existente do WAF. Essa opção aparece somente se você tiver ACLs da web configuradas.
5. Escolha sua ACL da web existente na tabela Escolher uma ACL da web.

6. Revise as configurações de distribuição restantes e escolha Criar distribuição.

Habilitar o controle de bots

Se você habilitar AWS WAF para a distribuição do CloudFront, será possível visualizar as solicitações de bots para determinado intervalo de tempo no painel de segurança no console do CloudFront. Também é possível habilitar ou desabilitar o controle de bots aqui.

São geradas cobranças quando você habilita o controle de bots. O painel de segurança fornece uma estimativa de custo.

Se você desabilitar o controle de bots, o painel de segurança exibirá o tráfego de bot por tipo e categoria. Se você desabilitar o controle de bots, o tráfego de bots será exibido com base na amostragem da solicitação.

Para habilitar o controle de bots

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Distribuições e escolha a distribuição que você deseja alterar.
3. Escolha a guia Segurança.
4. Role para baixo até a seção Solicitações de bots para determinado intervalo de tempo e escolha Habilitar controle de bots.
5. Na caixa de diálogo Bot Control (Controle de bots), em Configuração, marque a caixa de seleção Enable Bot Control for common bots (Habilitar controle de bots para bots comuns).
6. Escolha Salvar alterações.

Configurar a proteção por categoria de bot

Ao habilitar o controle de bots, você poderá configurar como cada bot não verificado é tratado por categoria de bot. Por exemplo, é possível configurar um bot de biblioteca HTTP como o Modo de monitor e atribuir um Desafio a um verificador de link.

Note

Os bots que a AWS reconhece como comuns e verificáveis, como crawlers de mecanismos de pesquisa conhecidos, não estão sujeitos às ações que você define aqui. O controle de

bots confirma que os bots validados vêm da origem informada antes de marcá-los como verificados.

Como configurar a proteção para uma categoria de bot

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Distribuições e escolha a distribuição que você deseja alterar.
3. Escolha a guia Segurança.
4. No gráfico Solicitações por categoria de bot, aponte para qualquer um dos itens na coluna Ação de bot não verificado e escolha o ícone de lápis para editá-lo.
5. Abra a lista resultante e escolha uma destas opções:
 - Bloquear
 - Permitir
 - Modo de monitor
 - CAPTCHA
 - Desafio
6. Escolha a marca de seleção ao lado da lista para salvar a alteração.

Gerenciar as proteções de segurança do AWS WAF no painel de segurança do CloudFront

O CloudFront cria um painel de segurança para cada uma das distribuições. Use os painéis no console do CloudFront. Com os painéis, é possível usar o CloudFront e o AWS WAF juntos em um único local para monitorar e gerenciar proteções de segurança comuns de aplicativos web. Os painéis fornecem as tarefas e os dados a seguir:

- Configuração de segurança: é possível habilitar e desabilitar as proteções do AWS WAF e ver todas as proteções específicas da aplicação, como as proteções do WordPress.
- Tendências de segurança: incluem solicitações permitidas e bloqueadas, solicitações de desafio e CAPTCHA, além dos principais tipos de ataque. É possível ver as taxas de tráfego e como elas mudam com o tempo. Por exemplo, se todas as solicitações aumentarem em 3%, mas as

solicitações permitidas aumentarem em 14%, isso indicará que você permitiu que uma parte maior do tráfego passasse no período atual.

- Solicitações de bot: é possível ver quanto tráfego vem dos bots, quais são os tipos de bots (verificados e não verificados) e como as alocações percentuais dos tipos de bots (verificados e não verificados) mudam com o tempo. Para ter mais informações sobre como habilitar controle de bots, consulte [Habilitar o controle de bots](#).
- Logs de solicitação: os dados de log podem ajudar a responder a perguntas sobre tendências de segurança ou solicitações de bots. É possível pesquisar os logs sem escrever consultas e visualizar gráficos agregados para ajudar a determinar se um conjunto filtrado de logs está sendo conduzido principalmente por um subconjunto de métodos HTTP, endereços IP, caminhos de URI ou países. Você pode passar o mouse sobre os valores nos gráficos e bloquear endereços IP e países. Para ter mais informações, consulte [Habilitar logs do AWS WAF](#).
- Gerenciamento de restrições geográficas: o CloudFront e o AWS WAF fornecem recursos de restrição geográfica. O CloudFront fornece restrições geográficas gratuitamente, mas as métricas para restrições geográficas do CloudFront não são exibidas no painel de segurança. Para ver as métricas de solicitações de países bloqueados, é necessário usar restrições geográficas do AWS WAF. Para fazer isso, passe o mouse sobre uma barra de país no painel de segurança e bloqueie o país. Para ter mais informações, consulte [Usar restrições geográficas do CloudFront](#).
- A opção Bloquear poderá não estar disponível se você tiver criado uma regra personalizada do AWS WAF fora do console do CloudFront para bloquear países.

Tópicos

- [Pré-requisitos](#)
- [Habilitar logs do AWS WAF](#)

Pré-requisitos

Você deverá habilitar o AWS WAF se quiser visualizar as métricas de segurança no painel de Segurança do CloudFront. Se você não habilitar o AWS WAF, só poderá usar o painel de Segurança para habilitar o AWS WAF ou configurar as restrições geográficas do CloudFront.

Para obter mais informações sobre como habilitar o AWS WAF, consulte [Habilitar o AWS WAF para distribuições](#).

Habilitar logs do AWS WAF

Os dados de log do AWS WAF podem ajudar você a isolar padrões de tráfego específicos. Por exemplo, os logs podem mostrar de onde vem determinado tráfego ou o que ele faz.

Caso você habilite o registro em log do AWS WAF no CloudWatch, o painel de segurança do CloudFront consulta, agrega e exibe insights dos logs do CloudWatch. Não cobramos pelo uso do painel de segurança, mas os preços do CloudWatch se aplicam aos logs consultados por meio do painel. Para obter mais informações, consulte [Amazon CloudWatch Pricing](#) (Preços do Amazon CloudWatch).

Para habilitar logs

1. Insira o volume esperado de solicitações na caixa Número de solicitações/mês para estimar os custos de habilitar os logs.
2. Marque a caixa de seleção Enable AWS WAF logs (Habilitar logs do WAF).
3. Escolha Habilitar.

O CloudFront cria um grupo de logs do CloudWatch e atualiza a configuração do AWS WAF a fim de começar a gerar logs para o CloudWatch. No primeiro uso, os dados de log podem levar vários minutos para serem exibidos. A seção Solicitações do gráfico lista cada solicitação. Abaixo das solicitações individuais, os gráficos de barras agregam dados por método HTTP, principais caminhos de URI, principais endereços IP e principais países. Os gráficos podem ajudar você a encontrar padrões. Por exemplo, é possível ver um volume desproporcional de solicitações de um único endereço IP ou dados de um país que você não viu anteriormente nos logs. É possível filtrar solicitações com base no país, no cabeçalho do host e em outros atributos para ajudar a encontrar tráfego indesejado. Depois de identificar esse tráfego, passe o mouse sobre uma solicitação individual ou item do gráfico e bloqueie um endereço IP ou país.

Note

As métricas exibidas são baseadas no web ACL. Portanto, se você associar a mesma ACL da web a várias distribuições, verá todas as métricas da sua ACL da web, não apenas as solicitações AWS WAF processadas para essa distribuição.

Configurar a limitação de taxa

A limitação de taxa é uma das recomendações que você pode receber ao configurar as proteções de segurança.

O CloudFront sempre permite a limitação de taxa no modo de monitor. Quando o modo de monitor está habilitado, o CloudFront captura métricas que informam se a taxa que você configurou no campo Limitação de taxa foi excedida, com que frequência e quanto.

Depois de salvar a distribuição, o CloudFront começa a coletar dados com base no número no campo Limitação de taxa.

Você pode gerenciar as configurações de limitação de taxa na seção Segurança – Web Application Firewall (WAF), na guia Segurança de qualquer distribuição do CloudFront.

Como configurar a limitação de taxa

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Distributions (Distribuições) e escolha a distribuição que deseja alterar.
3. Escolha a guia Segurança.
4. Na seção Web Application Firewall (WAF), ao lado de Limitação de taxa, escolha a mensagem do Modo de monitor para exibir uma caixa de diálogo com detalhes sobre os dados coletados. Se desejar, você poderá alterar o limite da taxa. Depois de ajustar a taxa, você pode escolher Ativar bloqueio (na caixa de diálogo) para desativar o modo de monitor. O CloudFront começará a bloquear solicitações que excedam o limite de taxa especificado.

Desabilitar as proteções de segurança do AWS WAF

Se sua distribuição não precisar de proteções de segurança AWS WAF, será possível desabilitar esse atributo usando o console do CloudFront.

Se você ativou a proteção do AWS WAF anteriormente e não escolheu uma configuração WAF existente (também conhecida como proteção com um clique), o CloudFront criou automaticamente uma ACL da web para você. Para ACLs da web criadas assim, o console do CloudFront desassociará o recurso e excluirá a ACL da web.

Desassociar uma ACL da web é diferente de excluí-la. A desassociação remove a ACL da web da sua distribuição, mas ela não é excluída da sua Conta da AWS. Para ter mais informações, consulte [Associar ou desassociar uma ACL da web a um recurso da AWS](#) na AWS WAF, AWS Firewall Manager e no Guia do desenvolvedor do AWS Shield Advanced.

Consulte o procedimento a seguir para desativar as proteções do AWS WAF e desassociar a ACL da web da sua distribuição.

Para desativar as proteções de segurança do AWS WAF no CloudFront

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Distributions (Distribuições) e escolha a distribuição que deseja alterar.
3. Escolha a guia Security (Segurança) e depois Edit (Editar).
4. Na seção Web Application Firewall (WAF), escolha Disable AWS WAF protection (Desabilitar proteção WAF).
5. Escolha Salvar alterações.

Observações

- Se você desativou a proteção de segurança do AWS WAF e ainda deseja excluir a ACL da web da sua Conta da AWS, é possível excluí-la manualmente. Siga o procedimento para [excluir uma ACL da web](#). No console AWS WAF & Shield, na página Web ACLs (ACLs da web), você deve escolher a lista Global (CloudFront) para encontrar as ACLs da web.
- Ao excluir uma distribuição do console do CloudFront, o CloudFront também tentará excluir a ACL da web se você escolher a proteção com um clique. Esse é o melhor esforço e não é sempre uma garantia. Para ter mais informações, consulte [Excluir uma distribuição do](#) .

Configurar o acesso seguro e restringir o acesso ao conteúdo

O CloudFront oferece várias opções para proteger o conteúdo que fornece. Veja a seguir algumas maneiras de usar o CloudFront para proteger e restringir o acesso ao conteúdo:

- Configure conexões HTTPS
- Impeça que usuários em locais geográficos específicos acessem o conteúdo
- Exigir que os usuários acessem o conteúdo usando signed URLs ou signed cookies do CloudFront
- Configure criptografia em nível de campo para campos de conteúdo específicos
- Use o AWS WAF para controlar o acesso ao seu conteúdo

Também é necessário implementar uma arquitetura resiliente a DDoS para a infraestrutura e as aplicações. Para ter informações, consulte [AWS Best Practices for DDoS Resiliency](#).

Tópicos

- [Usar HTTPS com o CloudFront](#)
- [Usar nomes de domínio alternativos e HTTPS](#)
- [Veicular conteúdo privado com URLs e cookies assinados](#)
- [Restringir o acesso a uma origem da AWS](#)
- [Restringir o acesso aos Application Load Balancers](#)
- [Restringir a distribuição geográfica do conteúdo](#)
- [Use criptografia de nível de campo para ajudar a proteger dados confidenciais](#)

Usar HTTPS com o CloudFront

É possível configurar o CloudFront para exigir que os visualizadores usem HTTPS para as suas conexões sejam criptografadas durante a comunicação do CloudFront com os visualizadores. Também é possível configurar o CloudFront para usar HTTPS com sua origem a fim de que as conexões sejam criptografadas quando o CloudFront se comunicar com os usuários.

Se você configurar o CloudFront para exigir HTTPS na comunicação com os visualizadores e com a origem, veja aqui o que acontece quando o CloudFront recebe uma solicitação:

1. Um visualizador envia uma solicitação HTTPS ao CloudFront. Há uma negociação SSL/TLS entre o visualizador e o CloudFront. No fim, o visualizador envia a solicitação em um formato criptografado.
2. Se o local da borda do CloudFront contiver uma resposta em cache, o CloudFront criptografará a resposta e a retornará ao visualizador. O visualizador, por sua vez, descriptografará a resposta.
3. Se o local da borda do CloudFront não contiver uma resposta em cache, o CloudFront realizará uma negociação SSL/TLS com a origem e, quando ela for concluída, encaminhará a solicitação para a origem em formato criptografado.
4. Sua origem descriptografa a solicitação, processa-a (gera uma resposta), criptografa-a e a devolve para o CloudFront.
5. O CloudFront descriptografa a resposta, criptografa-a novamente e a encaminha para o visualizador. O CloudFront também armazena em cache a resposta no local da borda para que ela esteja disponível na próxima vez que for solicitada.
6. O visualizador descriptografa a resposta.

O processo funciona basicamente da mesma forma, quer a origem seja um bucket do Amazon S3, um MediaStore ou uma origem personalizada, como um servidor HTTP/S.

Note

Para ajudar a impedir ataques do tipo renegociação SSL, o CloudFront não é compatível com a renegociação para solicitações do visualizador e da origem.

Para obter informações sobre como exigir HTTPS entre os visualizadores e o CloudFront, e entre o CloudFront e a origem, consulte os tópicos a seguir.

Tópicos

- [Exigir HTTPS na comunicação entre visualizadores e CloudFront](#)
- [Exigir HTTPS na comunicação entre o CloudFront e a origem personalizada](#)
- [Exigir HTTPS na comunicação entre o CloudFront e a origem do Amazon S3](#)
- [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#)
- [Protocolos e criptografias compatíveis entre o CloudFront e a origem](#)

Exigir HTTPS na comunicação entre visualizadores e CloudFront

É possível configurar um ou mais comportamentos de cache na distribuição do CloudFront para exigir HTTPS para a comunicação entre os visualizadores e o CloudFront. Também é possível configurar um ou mais comportamentos de cache para permitir HTTP e HTTPS, de forma que o CloudFront exija HTTPS de alguns objetos, mas não de outros. As etapas de configuração dependem do nome de domínio sendo usado nos URLs do objeto:

- Se estiver usando o nome de domínio atribuído pelo CloudFront à distribuição, como `d111111abcdef8.cloudfront.net`, altere a configuração Viewer Protocol Policy (Política de protocolo do visualizador) de um ou mais comportamentos de cache para exigir comunicação HTTPS. Nessa configuração, o CloudFront fornece o certificado SSL/TLS.

Para alterar o valor de Viewer Protocol Policy (Política de protocolo do visualizador) usando o console do CloudFront, consulte o procedimento apresentado posteriormente nesta seção.

Para obter informações sobre como usar a API do CloudFront para alterar o valor do elemento `ViewerProtocolPolicy`, consulte [UpdateDistribution](#) na Referência da API do Amazon CloudFront.

- Se estiver usando o nome do seu próprio domínio, como `example.com`, você precisará alterar várias configurações do CloudFront. Você também precisa usar um certificado SSL/TLS fornecido pelo AWS Certificate Manager (ACM) ou importar um certificado de uma autoridade de certificação de terceiros para o ACM ou para o armazenamento de certificados do IAM. Para obter mais informações, consulte [Usar nomes de domínio alternativos e HTTPS](#).

Note

Para garantir que os objetos obtidos pelos visualizadores do CloudFront sejam criptografados na origem, sempre use HTTPS entre o CloudFront e a origem. Se você alterou recentemente de HTTP para HTTPS entre o CloudFront e a origem, recomendamos invalidar os objetos nos pontos de presença do CloudFront. O CloudFront retornará um objeto ao visualizador, independentemente de o protocolo usado por ele (HTTP ou HTTPS) corresponder ou não ao protocolo usado pelo CloudFront para obter o objeto. Para obter mais informações sobre como remover ou substituir objetos em uma distribuição, consulte [Adicionar, remover ou substituir conteúdo distribuído pelo CloudFront](#).

Exigir HTTPS de visualizadores

Para exigir HTTPS entre os visualizadores e o CloudFront para um ou mais comportamentos de cache, execute o procedimento a seguir.

Como configurar o CloudFront para exigir HTTPS entre os visualizadores e o CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel superior do console do CloudFront, escolha o ID da distribuição que você deseja atualizar.
3. Na guia Behaviors (Comportamentos), selecione o comportamento de cache que você deseja atualizar e, em seguida, escolha Edit (Editar).
4. Especifique um dos seguintes valores para Viewer Protocol Policy (Política de protocolo do visualizador):

Redirect HTTP to HTTPS

Os visualizadores podem usar os dois protocolos. Solicitações HTTP GET e HEAD são automaticamente redirecionados para solicitações HTTPS. O CloudFront retorna o código de status HTTP 301 (movido permanentemente) com o novo URL HTTPS. Depois, o visualizador reenvia a solicitação para o CloudFront usando o URL de HTTPS.

Important

Se você enviar POST, PUT, DELETE, OPTIONS ou PATCH por HTTP com um comportamento de cache HTTP para HTTPS e a versão do protocolo de solicitação HTTP 1.1 ou superior, o CloudFront redirecionará a solicitação para um local HTTPS com um código de status HTTP 307 (redirecionamento temporário). Isso garante que a solicitação seja enviada novamente para o novo local usando o mesmo método e carga do corpo.

Se você enviar solicitações POST, PUT, DELETE, OPTIONS ou PATCH por HTTP para comportamento de cache HTTPS com a versão do protocolo de solicitação inferior a HTTP 1.1, o CloudFront retornará um código de status HTTP 403 (proibido).

Quando um visualizador faz uma solicitação HTTP que é redirecionada para uma solicitação HTTPS, o CloudFront cobra pelas duas solicitações. Para a solicitação HTTP, a cobrança é

somente pela solicitação e cabeçalhos retornados pelo CloudFront para o visualizador. Para a solicitação HTTPS, a cobrança é pela solicitação e pelos cabeçalhos e objeto retornados por sua origem.

HTTPS Only (Somente HTTPS)

Os visualizadores só podem acessar seu conteúdo se estiverem usando HTTPS. Se um visualizador enviar uma solicitação HTTP, em vez de HTTPS, o CloudFront retornará o código de status HTTP 403 (proibido) e não retornará o objeto.

5. Escolha Salvar alterações.
6. Repita as etapas 3 a 5 para cada comportamento de cache adicional para o qual você deseja exigir HTTPS entre os visualizadores e o CloudFront.
7. Antes de usar a configuração atualizada em um ambiente de produção, confirme:
 - Se o padrão de caminho de cada comportamento de cache se aplica apenas às solicitações nas quais os visualizadores devem usar HTTPS.
 - Se os comportamentos de cache estão listados na ordem em que você deseja que o CloudFront os avalie. Para obter mais informações, consulte [Padrão de caminho](#).
 - Se os comportamentos de cache estão roteando as solicitações para as origens corretas.

Exigir HTTPS na comunicação entre o CloudFront e a origem personalizada

Você pode exigir o uso de HTTPS na comunicação entre o CloudFront e sua origem.

Note

Se a origem for um bucket do Amazon S3 configurado como um endpoint do site, não será possível configurar o CloudFront para usar HTTPS com a origem porque o Amazon S3 não é compatível com HTTPS para endpoints de site.

Para exigir HTTPS entre o CloudFront e sua origem, siga os procedimentos deste tópico para fazer o seguinte:

1. Em sua distribuição, altere a configuração Origin Protocol Policy (Política de protocolo da origem) para a origem.

2. Instale um certificado SSL/TLS no servidor de origem (isso não é necessário ao usar uma origem do Amazon S3 ou algumas outras origens da AWS).

Tópicos

- [Exigir HTTPS para origens personalizadas](#)
- [Instalar um certificado SSL/TLS na origem personalizada](#)

Exigir HTTPS para origens personalizadas

O procedimento a seguir explica como configurar o CloudFront para usar HTTPS para se comunicar com um balanceador de carga do Elastic Load Balancing, uma instância do Amazon EC2 ou outra origem personalizada. Para obter informações sobre como usar a API do CloudFront para atualizar uma distribuição, consulte [UpdateDistribution](#) na Referência da API do Amazon CloudFront.

Como configurar o CloudFront para exigir HTTPS entre o CloudFront e a origem personalizada

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel superior do console do CloudFront, escolha o ID da distribuição que você deseja atualizar.
3. Na guia Behaviors (Comportamentos), selecione a origem que você deseja atualizar e, em seguida, escolha Edit (Editar).
4. Atualize as seguintes configurações:

Política de protocolo da origem

Altere Origin Protocol Policy para as origens aplicáveis à sua distribuição:

- **HTTPS Only (Somente HTTPS):** o CloudFront usa HTTPS para se comunicar com a origem personalizada.
- **Match Viewer (Corresponder visualizador):** o CloudFront se comunica com a origem personalizada usando HTTP ou HTTPS, dependendo do protocolo da solicitação do visualizador. Por exemplo, se você escolher Match Viewer (Corresponder visualizador) para Origin Protocol Policy (Política de protocolo de origem) e o visualizador usar HTTPS para solicitar um objeto do CloudFront, o CloudFront também usará HTTPS para encaminhar a solicitação para a origem.

Escolha Match Viewer somente se você especificar Redirect HTTP to HTTPS ou HTTPS Only em Viewer Protocol Policy.

Observe que o CloudFront armazenará o objeto em cache somente uma vez se os visualizadores fizerem solicitações usando protocolos HTTP e HTTPS.

Origin SSL Protocols

Escolha os Origin SSL Protocols para as origens aplicáveis à sua distribuição. O protocolo SSLv3 é menos seguro, portanto, recomendamos que você escolha SSLv3 somente se a origem não for compatível com TLSv1 ou posterior. O handshake TLSv1 é compatível com as versões anteriores e posteriores do SSLv3, mas não com o TLSv1.1 e posterior. Quando você escolhe SSLv3, o CloudFront envia somente solicitações de handshake SSLv3.

5. Escolha Salvar alterações.
6. Repita as etapas 3 a 5 para cada origem adicional para a qual você deseja exigir HTTPS entre o CloudFront e a origem personalizada.
7. Antes de usar a configuração atualizada em um ambiente de produção, confirme:
 - Se o padrão de caminho de cada comportamento de cache se aplica apenas às solicitações nas quais os visualizadores devem usar HTTPS.
 - Se os comportamentos de cache estão listados na ordem em que você deseja que o CloudFront os avalie. Para ter mais informações, consulte [Padrão de caminho](#).
 - Os comportamentos de cache estão roteando solicitações para as origens nas quais você alterou a opção Origin Protocol Policy.

Instalar um certificado SSL/TLS na origem personalizada

Você pode usar um certificado SSL/TLS das seguintes fontes na sua origem personalizada:

- Se a origem for um balanceador de carga do Elastic Load Balancing, você poderá usar um certificado fornecido pelo AWS Certificate Manager (ACM). Você também pode usar um certificado assinado por uma autoridade de certificação terceirizada reconhecida e importado no ACM.
- Para origens diferentes de balanceadores de carga do Elastic Load Balancing, use um certificado assinado por uma autoridade de certificação (CA) confiável de terceiros, como a Comodo, a DigiCert ou a Symantec.

O certificado retornado da origem deve incluir um dos seguintes nomes de domínio:

- O nome de domínio no campo Origin domain (Domínio de origem) da origem (o campo `DomainName` na API do CloudFront).
- O nome do domínio no cabeçalho `Host`, se o comportamento do cache estiver configurado para encaminhar o cabeçalho `Host` para a origem.

Ao usar HTTPS para se comunicar com a origem, o CloudFront verifica se o certificado foi emitido por uma autoridade de certificação reconhecida. O CloudFront é compatível com as mesmas autoridades de certificação que o Mozilla. Para ver a lista atual, consulte [Mozilla Included CA Certificate List](#). Você não pode usar um certificado autoassinado para comunicação HTTPS entre o CloudFront e a origem.

Important

Se o servidor de origem retornar um certificado expirado, inválido ou autoassinado, ou a cadeia de certificados na ordem errada, o CloudFront interromperá a conexão TCP, retornará o código de status HTTP 502 (gateway inválido) ao visualizador e definirá o cabeçalho `X-Cache` como `Error from cloudfront`. Além disso, se toda a cadeia de certificados, inclusive o certificado intermediário, não estiver presente, o CloudFront interromperá a conexão TCP.

Exigir HTTPS na comunicação entre o CloudFront e a origem do Amazon S3

Quando a origem é um bucket do Amazon S3, as opções para usar HTTPS para comunicação com o CloudFront dependem de como você está usando o bucket. Se o bucket do Amazon S3 estiver configurado como um endpoint de site, não será possível configurar o CloudFront para usar HTTPS para comunicação com a origem, pois o Amazon S3 não é compatível com conexões HTTPS nessa configuração.

Quando a origem é um bucket do Amazon S3 compatível com a comunicação HTTPS, o CloudFront sempre encaminha as solicitações para o S3 usando o protocolo que os visualizadores usaram para enviar as solicitações. A definição padrão da configuração [Protocolo \(somente origens personalizadas\)](#) é `Match Viewer` (Corresponder visualizador) e não pode ser alterada.

Para exigir HTTPS para comunicação entre o CloudFront e o Amazon S3, você deve alterar o valor de Viewer Protocol Policy (Política de protocolo do visualizador) para Redirect HTTP to HTTPS (Redirecionar HTTP para HTTPS) ou HTTPS Only (Somente HTTPS). O procedimento mais adiante nesta seção explica como usar o console do CloudFront para alterar a Viewer Protocol Policy (Política de protocolo do visualizador). Para informações sobre como usar a API do CloudFront para atualizar o elemento `ViewerProtocolPolicy` de uma distribuição, consulte [UpdateDistribution](#) na Referência da API do Amazon CloudFront.

Quando você usa HTTPS com um bucket do Amazon S3 compatível com a comunicação HTTPS, o Amazon S3 fornece o certificado SSL/TLS, portanto, você não precisa fazer isso.

Exigir HTTPS para uma origem do Amazon S3

O procedimento a seguir mostra como configurar o CloudFront para exigir HTTPS para a origem do Amazon S3.

Como configurar o CloudFront para exigir HTTPS para a origem do Amazon S3

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel superior do console do CloudFront, escolha o ID da distribuição que você deseja atualizar.
3. Na guia Behaviors, escolha o comportamento de cache que você deseja atualizar e, em seguida, escolha Edit.
4. Especifique um dos seguintes valores para Viewer Protocol Policy:

Redirect HTTP to HTTPS

Os visualizadores podem usar os dois protocolos, mas solicitações HTTP são automaticamente redirecionadas para HTTPS. O CloudFront retorna o código de status HTTP 301 (movido permanentemente) com o novo URL HTTPS. Depois, o visualizador reenvia a solicitação para o CloudFront usando o URL de HTTPS.

Important

O CloudFront não redireciona solicitações DELETE, OPTIONS, PATCH, POST ou PUT de HTTP para HTTPS. Se você configurar o redirecionamento de um comportamento de cache para HTTPS, o CloudFront responderá às solicitações HTTP DELETE,

OPTIONS, PATCH, POST ou PUT para esse comportamento de cache com o código de status HTTP 403 (proibido).

Quando um visualizador faz uma solicitação HTTP que é redirecionada para uma solicitação HTTPS, o CloudFront cobra pelas duas solicitações. Para a solicitação HTTP, a cobrança é somente pela solicitação e cabeçalhos retornados pelo CloudFront para o visualizador. Para a solicitação HTTPS, a cobrança é pela solicitação e pelos cabeçalhos e objeto retornados por sua origem.

HTTPS Only

Os visualizadores só podem acessar seu conteúdo se estiverem usando HTTPS. Se um visualizador enviar uma solicitação HTTP, em vez de HTTPS, o CloudFront retornará o código de status HTTP 403 (proibido) e não retornará o objeto.

5. Escolha Yes, Edit.
6. Repita as etapas 3 a 5 para cada comportamento de cache adicional para o qual você deseja exigir HTTPS entre os visualizadores e o CloudFront, e entre o CloudFront e o S3.
7. Antes de usar a configuração atualizada em um ambiente de produção, confirme:
 - Se o padrão de caminho de cada comportamento de cache se aplica apenas às solicitações nas quais os visualizadores devem usar HTTPS.
 - Se os comportamentos de cache estão listados na ordem em que você deseja que o CloudFront os avalie. Para obter mais informações, consulte [Padrão de caminho](#).
 - Se os comportamentos de cache estão roteando as solicitações para as origens corretas.

Protocolos e cifras compatíveis entre visualizadores e o CloudFront

Ao [exigir HTTPS entre os visualizadores e a distribuição do CloudFront](#), escolha uma [política de segurança](#) que determine as seguintes configurações:

- O protocolo SSL/TLS mínimo que o CloudFront usa para se comunicar com os visualizadores
- As criptografias que o CloudFront pode usar para criptografar a comunicação com os visualizadores.

Para escolher uma política de segurança, especifique o valor aplicável para [Política de segurança \(versão mínima de SSL/TLS\)](#). A tabela a seguir lista os protocolos e as criptografias que o CloudFront pode usar para cada política de segurança.

Um visualizador deve ser compatível com pelo menos uma dessas criptografias compatíveis para estabelecer uma conexão HTTPS com o CloudFront. O CloudFront escolhe uma criptografia na ordem listada entre as criptografias compatíveis com o visualizador. Consulte também [Nomes de cifras OpenSSL, s2n e RFC](#).

	Política de segurança						
	SSLv3	TLSv1	TLSv1_2 6	TLSv1.1 016	TLSv1.2 018	TLSv1.2 019	TLSv1.2_2 021
Protocolos SSL/TLS compatíveis							
TLSv1.3	◆	◆	◆	◆	◆	◆	◆
TLSv1.2	◆	◆	◆	◆	◆	◆	◆
TLSv1.1	◆	◆	◆	◆			
TLSv1	◆	◆	◆				
SSLv3	◆						
Cifras TLSv1.3 compatíveis							
TLS_AES_128_GCM_SH A256	◆	◆	◆	◆	◆	◆	◆
TLS_AES_256_GCM_SH A384	◆	◆	◆	◆	◆	◆	◆
TLS_CHACHA20_POLY1 305_SHA256	◆	◆	◆	◆	◆	◆	◆
Cifras ECDSA compatíveis							

	Política de segurança						
	SSLv3	TLSv1	TLSv1.2_6	TLSv1.1_016	TLSv1.2_018	TLSv1.2_019	TLSv1.2_2021
ECDHE-ECDSA-AES128-GCM-SHA256	◆	◆	◆	◆	◆	◆	◆
ECDHE-ECDSA-AES128-SHA256	◆	◆	◆	◆	◆	◆	
ECDHE-ECDSA-AES128-SHA	◆	◆	◆	◆			
ECDHE-ECDSA-AES256-GCM-SHA384	◆	◆	◆	◆	◆	◆	◆
ECDHE-ECDSA-CHACHA20-POLY1305	◆	◆	◆	◆	◆	◆	◆
ECDHE-ECDSA-AES256-SHA384	◆	◆	◆	◆	◆	◆	
ECDHE-ECDSA-AES256-SHA	◆	◆	◆	◆			
Cifras RSA compatíveis							
ECDHE-RSA-AES128-GCM-SHA256	◆	◆	◆	◆	◆	◆	◆
ECDHE-RSA-AES128-SHA256	◆	◆	◆	◆	◆	◆	
ECDHE-RSA-AES128-SHA	◆	◆	◆	◆			
ECDHE-RSA-AES256-GCM-SHA384	◆	◆	◆	◆	◆	◆	◆

	Política de segurança						
	SSLv3	TLSv1	TLSv1.2_6	TLSv1.1_016	TLSv1.2_018	TLSv1.2_019	TLSv1.2_2_021
ECDHE-RSA-CHACHA20-POLY1305	◆	◆	◆	◆	◆	◆	◆
ECDHE-RSA-AES256-SHA384	◆	◆	◆	◆	◆	◆	
ECDHE-RSA-AES256-SHA	◆	◆	◆	◆			
AES128-GCM-SHA256	◆	◆	◆	◆	◆		
AES256-GCM-SHA384	◆	◆	◆	◆	◆		
AES128-SHA256	◆	◆	◆	◆	◆		
AES256-SHA	◆	◆	◆	◆			
AES128-SHA	◆	◆	◆	◆			
DES-CBC3-SHA	◆	◆					
RC4-MD5	◆						

Nomes de cifras OpenSSL, s2n e RFC

O OpenSSL e o [s2n](#) usam nomes diferentes para cifras que os padrões TLS usam ([RFC 2246](#), [RFC 4346](#), [RFC 5246](#) e [RFC 8446](#)). A tabela a seguir mapeia os nomes do OpenSSL e do s2n para o nome do RFC para cada cifra.

Para cifras com algoritmos de troca de chaves de curva elíptica, o CloudFront oferece suporte às seguintes curvas elípticas:

- prime256v1
- secp384r1
- X25519

Nome da cifra do OpenSSL e do s2n	Nome da criptografia RFC
Cifras TLSv1.3 compatíveis	
TLS_AES_128_GCM_SHA256	TLS_AES_128_GCM_SHA256
TLS_AES_256_GCM_SHA384	TLS_AES_256_GCM_SHA384
TLS_CHACHA20_POLY1305_SHA256	TLS_CHACHA20_POLY1305_SHA256
Cifras ECDSA compatíveis	
ECDHE-ECDSA-AES128-GCM-SHA256	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
ECDHE-ECDSA-AES128-SHA256	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
ECDHE-ECDSA-AES128-SHA	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
ECDHE-ECDSA-AES256-GCM-SHA384	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
ECDHE-ECDSA-CHACHA20-POLY1305	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256
ECDHE-ECDSA-AES256-SHA384	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
ECDHE-ECDSA-AES256-SHA	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
Cifras RSA compatíveis	
ECDHE-RSA-AES128-GCM-SHA256	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
ECDHE-RSA-AES128-SHA256	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256

Nome da cifra do OpenSSL e do s2n	Nome da criptografia RFC
ECDHE-RSA-AES128-SHA	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
ECDHE-RSA-AES256-GCM-SHA384	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
ECDHE-RSA-CHACHA20-POLY1305	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256
ECDHE-RSA-AES256-SHA384	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
ECDHE-RSA-AES256-SHA	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
AES128-GCM-SHA256	TLS_RSA_WITH_AES_128_GCM_SHA256
AES256-GCM-SHA384	TLS_RSA_WITH_AES_256_GCM_SHA384
AES128-SHA256	TLS_RSA_WITH_AES_128_CBC_SHA256
AES256-SHA	TLS_RSA_WITH_AES_256_CBC_SHA
AES128-SHA	TLS_RSA_WITH_AES_128_CBC_SHA
DES-CBC3-SHA	TLS_RSA_WITH_3DES_EDE_CBC_SHA
RC4-MD5	TLS_RSA_WITH_RC4_128_MD5

Esquemas de assinatura compatíveis entre visualizadores e o CloudFront

O CloudFront é compatível com os seguintes esquemas de assinatura para conexões entre visualizadores e o CloudFront.

- TLS_SIGNATURE_SCHEME_RSA_PSS_PSS_SHA256
- TLS_SIGNATURE_SCHEME_RSA_PSS_SHA384
- TLS_SIGNATURE_SCHEME_RSA_PSS_SHA512

- TLS_SIGNATURE_SCHEME_RSA_PSS_RSAE_SHA256
- TLS_SIGNATURE_SCHEME_RSA_PSS_RSAE_SHA384
- TLS_SIGNATURE_SCHEME_RSA_PSS_RSAE_SHA512
- TLS_SIGNATURE_SCHEME_RSA_PKCS1_SHA256
- TLS_SIGNATURE_SCHEME_RSA_PKCS1_SHA384
- TLS_SIGNATURE_SCHEME_RSA_PKCS1_SHA512
- TLS_SIGNATURE_SCHEME_RSA_PKCS1_SHA224
- TLS_SIGNATURE_SCHEME_ECDSA_SHA256
- TLS_SIGNATURE_SCHEME_ECDSA_SHA384
- TLS_SIGNATURE_SCHEME_ECDSA_SHA512
- TLS_SIGNATURE_SCHEME_ECDSA_SHA224
- TLS_SIGNATURE_SCHEME_ECDSA_SECP256R1_SHA256
- TLS_SIGNATURE_SCHEME_ECDSA_SECP384R1_SHA384
- TLS_SIGNATURE_SCHEME_RSA_PKCS1_SHA1
- TLS_SIGNATURE_SCHEME_ECDSA_SHA1

Protocolos e criptografias compatíveis entre o CloudFront e a origem

Se você optar por [exigir HTTPS entre o CloudFront e a origem](#), poderá decidir [qual protocolo SSL/TLS permitir](#) para a conexão segura, e o CloudFront poderá conectar à origem usando qualquer uma das cifras ECDSA ou RSA listadas na tabela a seguir. A origem deve ser compatível com pelo menos uma dessas criptografias do CloudFront para estabelecer uma conexão HTTPS com a origem.

O OpenSSL e o [s2n](#) usam nomes diferentes para cifras que os padrões TLS usam ([RFC 2246](#), [RFC 4346](#), [RFC 5246](#) e [RFC 8446](#)). A tabela a seguir inclui os nomes do OpenSSL e do s2n, bem como o nome da RFC, para cada cifra.

Para cifras com algoritmos de troca de chaves de curva elíptica, o CloudFront oferece suporte às seguintes curvas elípticas:

- prime256v1
- secp384r1
- X25519

Nome da cifra do OpenSSL e do s2n	Nome da criptografia RFC
Cifras ECDSA compatíveis	
ECDHE-ECDSA-AES256-GCM-SHA384	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
ECDHE-ECDSA-AES256-SHA384	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
ECDHE-ECDSA-AES256-SHA	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
ECDHE-ECDSA-AES128-GCM-SHA256	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
ECDHE-ECDSA-AES128-SHA256	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
ECDHE-ECDSA-AES128-SHA	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
Cifras RSA compatíveis	
ECDHE-RSA-AES256-GCM-SHA384	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
ECDHE-RSA-AES256-SHA384	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
ECDHE-RSA-AES256-SHA	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
ECDHE-RSA-AES128-GCM-SHA256	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
ECDHE-RSA-AES128-SHA256	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256

Nome da cifra do OpenSSL e do s2n	Nome da criptografia RFC
ECDHE-RSA-AES128-SHA	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
AES256-SHA	TLS_RSA_WITH_AES_256_CBC_SHA
AES128-SHA	TLS_RSA_WITH_AES_128_CBC_SHA
DES-CBC3-SHA	TLS_RSA_WITH_3DES_EDE_CBC_SHA
RC4-MD5	TLS_RSA_WITH_RC4_128_MD5

Esquemas de assinatura compatíveis entre o CloudFront e a origem

O CloudFront é compatível com os seguintes esquemas de assinatura para conexões entre o CloudFront e a origem.

- TLS_SIGNATURE_SCHEME_RSA_PKCS1_SHA256
- TLS_SIGNATURE_SCHEME_RSA_PKCS1_SHA384
- TLS_SIGNATURE_SCHEME_RSA_PKCS1_SHA512
- TLS_SIGNATURE_SCHEME_RSA_PKCS1_SHA224
- TLS_SIGNATURE_SCHEME_ECDSA_SHA256
- TLS_SIGNATURE_SCHEME_ECDSA_SHA384
- TLS_SIGNATURE_SCHEME_ECDSA_SHA512
- TLS_SIGNATURE_SCHEME_ECDSA_SHA224
- TLS_SIGNATURE_SCHEME_RSA_PKCS1_SHA1
- TLS_SIGNATURE_SCHEME_ECDSA_SHA1

Usar nomes de domínio alternativos e HTTPS

Se quiser usar seu nome de domínio nos URLs dos arquivos (por exemplo, `https://www.example.com/image.jpg`) e quiser que os visualizadores usem HTTPS, será necessário concluir as etapas dos tópicos a seguir. (Se você usar o nome de domínio de distribuição padrão do CloudFront nos seus URLs, por exemplo, `https://d1111111abcdef8.cloudfront.net/`

image.jpg, siga as orientações no tópico a seguir: [Exigir HTTPS na comunicação entre visualizadores e CloudFront.](#))

Important

Ao adicionar um certificado à sua distribuição, o CloudFront o propagará imediatamente para todos os pontos de presença. À medida que novos pontos de presença são disponibilizados, o CloudFront propaga o certificado para esses pontos também. Não é possível restringir os pontos de presença para os quais o CloudFront propaga os certificados.

Tópicos

- [Escolher como o CloudFront atende a solicitações HTTPS](#)
- [Requisitos para usar certificados SSL/TLS com o CloudFront](#)
- [Cotas no uso de certificados SSL/TLS com o CloudFront \(somente HTTPS entre visualizadores e o CloudFront\)](#)
- [Configurar nomes de domínio alternativos e HTTPS](#)
- [Determinar o tamanho da chave pública em um certificado RSA SSL/TLS](#)
- [Aumentar as cotas de certificados SSL/TLS](#)
- [Trocar certificados SSL/TLS](#)
- [Reverter um certificado SSL/TLS personalizado para o certificado padrão do CloudFront](#)
- [Alternar de um certificado SSL/TLS personalizado com endereços IP dedicados para SNI](#)

Escolher como o CloudFront atende a solicitações HTTPS

Se quiser que os visualizadores usem HTTPS e nomes de domínio alternativos para os arquivos, escolha uma das opções a seguir de como o CloudFront atende a solicitações HTTPS:

- Usar a [indicação de nome de servidor \(SNI\)](#): recomendado
- Use um endereço IP dedicado em cada ponto de presença

Esta seção explica como cada opção funciona.

Usar SNI para atender a solicitações HTTPS (funciona para a maioria dos clientes)

A [Indicação de nome de servidor \(SNI\)](#) é uma extensão do protocolo TLS, compatível com os navegadores e clientes lançados após 2010. Se você configurar o CloudFront para atender a solicitações HTTPS usando SNI, ele associará seu nome de domínio alternativo a um endereço IP para cada ponto de presença. Quando um visualizador envia uma solicitação HTTPS para seu conteúdo, o DNS a roteia para o endereço IP do ponto de presença correto. O endereço IP para o seu nome de domínio é determinado durante a negociação do handshake SSL/TLS. O endereço IP não é dedicado à sua distribuição.

A negociação SSL/TLS ocorre no início do processo de estabelecimento de uma conexão HTTPS. Se o CloudFront não conseguir determinar imediatamente qual é o domínio da solicitação, ele interromperá a conexão. Veja o que acontece quando um visualizador compatível com SNI envia uma solicitação HTTP para seu conteúdo:

1. O visualizador recebe automaticamente o nome de domínio do URL da solicitação e o adiciona à extensão SNI da mensagem de saudações ao cliente TLS.
2. Quando o CloudFront recebe a mensagem de saudações ao cliente TLS, ele usa o nome de domínio na extensão SNI para encontrar a distribuição correspondente do CloudFront e envia de volta o certificado TLS associado.
3. O visualizador e o CloudFront executam a negociação SSL/TLS.
4. O CloudFront retorna o conteúdo solicitado para o visualizador.

Para obter uma lista atual dos navegadores compatíveis com SNI, consulte a entrada da Wikipedia [Server Name Indication](#).

Se você quiser usar a SNI, mas o navegador de alguns dos seus usuário não forem compatíveis, há várias opções:

- Configure o CloudFront para atender a solicitações HTTPS usando endereços IP dedicados, em vez de SNI. Para obter mais informações, consulte [Usar um endereço IP dedicado para atender a solicitações HTTPS \(funciona para todos os clientes\)](#).
- Use o certificado SSL/TLS do CloudFront, em vez de um certificado personalizado. Isso requer o uso do nome de domínio do CloudFront da sua distribuição nos URLs dos seus arquivos, por exemplo, `https://d111111abcdef8.cloudfront.net/logo.png`.

Se você usar o certificado padrão do CloudFront, os visualizadores deverão oferecer suporte ao protocolo SSL TLSv1 ou posterior. O CloudFront não oferece suporte a SSLv3 com o certificado padrão do CloudFront.

Também é necessário alterar o certificado SSL/TLS usado pelo CloudFront de um certificado personalizado para o certificado padrão do CloudFront:

- Se você não usou sua distribuição para distribuir o conteúdo, poderá simplesmente alterar a configuração. Para obter mais informações, consulte [Atualizar uma distribuição](#).
- Caso contrário, será necessário criar uma distribuição do CloudFront e alterar os URLs dos seus arquivos para reduzir ou eliminar o tempo de indisponibilidade do conteúdo. Para obter mais informações, consulte [Reverter um certificado SSL/TLS personalizado para o certificado padrão do CloudFront](#).
- Se você puder controlar qual navegador seus usuários usam, peça que atualizem-no para um que seja compatível com SNI.
- Use HTTP, em vez de HTTPS.

Usar um endereço IP dedicado para atender a solicitações HTTPS (funciona para todos os clientes)

A Indicação de Nome de Servidor (SNI) é uma maneira de associar uma solicitação a um domínio. Outra maneira é usar um endereço IP dedicado. Se tiver usuários que não podem fazer a atualização para um navegador ou cliente lançado após 2010, você poderá usar um endereço IP dedicado para atender a solicitações HTTPS. Para obter uma lista atual dos navegadores compatíveis com SNI, consulte a entrada da Wikipedia [Server Name Indication](#).

Important

Se você configurar o CloudFront para atender a solicitações HTTPS usando endereços IP dedicados, será cobrado um adicional por mês. A cobrança começará quando você associar seu certificado SSL/TLS a uma distribuição e habilitar a distribuição. Para mais informações sobre os preços do CloudFront, consulte [Definição de preços do Amazon CloudFront](#). Além disso, consulte [Using the Same Certificate for Multiple CloudFront Distributions](#).

Ao configurar o CloudFront para atender a solicitações HTTPS usando endereços IP dedicados, ele associará seu certificado a um endereço IP dedicado em cada local da borda. Veja o que acontece quando um visualizador envia uma solicitação HTTP para seu conteúdo:

1. O DNS roteia a solicitação para o endereço IP da sua distribuição no ponto de presença aplicável.
2. Se uma solicitação do cliente fornecer a extensão SNI na mensagem ClientHello, o CloudFront procurará uma distribuição associada a essa SNI.
 - Se houver correspondência, o CloudFront responderá à solicitação com o certificado SSL/TLS.
 - Se não houver correspondência, o CloudFront usa o endereço IP para identificar sua distribuição e determinar qual certificado SSL/TLS retornar para o visualizador.
3. O visualizador e o CloudFront executam uma negociação SSL/TLS usando seu certificado SSL/TLS.
4. O CloudFront retorna o conteúdo solicitado para o visualizador.

Este método funciona para todas as solicitações HTTPS, independentemente do navegador ou outro visualizador usado pelo usuário.

Note

IPs dedicados não são IPs estáticos e podem mudar com o tempo. O endereço IP que é retornado para o local da borda é alocado dinamicamente dos intervalos de endereços IP da [lista de servidores de borda do CloudFront](#).

Os intervalos de endereço IP para os servidores de borda do CloudFront estão sujeitos a alterações. Para receber uma notificação sobre as alterações de endereço IP, [assine as alterações de endereço IP público da AWS via Amazon SNS](#).

Solicitar permissão para usar três ou mais certificados SSL/TLS com IP dedicado

Se precisar de permissão para associar de forma permanente três ou mais certificados SSL/TLS com IP dedicado ao CloudFront, execute o procedimento a seguir. Para mais detalhes sobre solicitações HTTPS, consulte [Escolher como o CloudFront atende a solicitações HTTPS](#).

Note

Esse procedimento serve para usar três ou mais certificados com IP dedicado em todas as suas distribuições do CloudFront. O valor padrão é 2. Lembre-se de que você não pode associar mais de um certificado SSL a uma distribuição.

É possível associar somente um único certificado SSL/TLS a uma distribuição do CloudFront por vez. Esse número é para o total de certificados SSL com IP dedicado que podem ser usados em todas as suas distribuições do CloudFront.

Como solicitar permissão para usar três ou mais certificados com uma distribuição do CloudFront

1. Acesse o [Support Center](#) e crie um caso.
2. Indique quantos certificados você precisa de permissão para usar e descreva as circunstâncias na solicitação. Atualizaremos sua conta o mais rápido possível.
3. Vá para o próximo procedimento.

Requisitos para usar certificados SSL/TLS com o CloudFront

Os requisitos para certificados SSL/TLS estão descritos neste tópico. Eles se aplicam a ambas as opções a seguir, exceto conforme observado:

- Certificados para o uso de HTTPS entre visualizadores e o CloudFront
- Certificados para o uso de HTTPS entre o CloudFront e sua origem

Tópicos

- [Emissor do certificado](#)
- [Região da AWS para AWS Certificate Manager](#)
- [Formato do certificado](#)
- [Certificados intermediários](#)
- [Tipo de chave](#)
- [Chave privada](#)
- [Permissões](#)
- [Tamanho da chave do certificado](#)

- [Tipos de certificados compatíveis](#)
- [Data de expiração e renovação do certificado](#)
- [Nomes de domínio na distribuição do CloudFront e no certificado](#)
- [Versão mínima do protocolo SSL/TLS](#)
- [Versões de HTTP compatíveis](#)

Emissor do certificado

Recomendamos usar um certificado emitido pelo [AWS Certificate Manager \(ACM\)](#). Para obter informações sobre como obter um certificado do ACM, consulte o [Manual do usuário do AWS Certificate Manager](#). Para usar um certificado do ACM com o CloudFront, solicite (ou importe) o certificado na região Leste dos EUA (Norte da Virgínia) (us-east-1).

O CloudFront é compatível com as mesmas autoridades de certificação (CAs) da Mozilla. Portanto, se você não usa o ACM, use um certificado emitido por uma CA na [Lista de certificados CA incluídos no Mozilla](#). Para obter mais informações sobre como obter e instalar um certificado, consulte a documentação do software do servidor HTTP e a documentação da CA.

Região da AWS para AWS Certificate Manager

Para usar um certificado no AWS Certificate Manager (ACM) a fim de exigir HTTPS entre visualizadores e o CloudFront, solicite (ou importe) o certificado na região Leste dos EUA (Norte da Virgínia) (us-east-1).

Para exigir HTTPS entre o CloudFront e a origem se estiver usando um balanceador de carga no Elastic Load Balancing como origem, você poderá solicitar ou importar um certificado de qualquer Região da AWS.

Formato do certificado

O certificado deve estar no formato X.509 PEM. Esse será o formato padrão se você estiver usando o AWS Certificate Manager.

Certificados intermediários

Se você estiver usando uma autoridade de certificação (CA) de terceiros, indique todos os certificados intermediários na cadeia existente no arquivo .pem, começando com um para a CA que assinou o certificado do seu domínio. Normalmente, é possível encontrar um arquivo no site da sua CA com os certificados raiz e intermediários na ordem adequada da cadeia.

⚠ Important

Não inclua o seguinte: o certificado raiz, certificados intermediários não presentes no caminho de relação de confiança nem o certificado de chave pública da sua CA.

Veja um exemplo abaixo:

```
-----BEGIN CERTIFICATE-----  
Intermediate certificate 2  
-----END CERTIFICATE-----  
-----BEGIN CERTIFICATE-----  
Intermediate certificate 1  
-----END CERTIFICATE-----
```

Tipo de chave

O CloudFront é compatível com pares de chaves RSA e ECDSA públicas/privadas.

O CloudFront oferece suporte a conexões HTTPS para visualizadores e origens usando certificados RSA e ECDSA. Com o [AWS Certificate Manager \(ACM\)](#), é possível solicitar e importar certificados RSA ou ECDSA e, em seguida, associá-los à sua distribuição do CloudFront.

Para obter as listas de criptografias RSA e ECDSA compatíveis com o CloudFront que você pode negociar em conexões HTTPS, consulte [the section called “Protocolos e cifras compatíveis entre visualizadores e o CloudFront”](#) e [the section called “Protocolos e criptografias compatíveis entre o CloudFront e a origem”](#).

Chave privada

Se você estiver usando um certificado de uma autoridade de certificação (CA) terceirizada, observe:

- A chave privada deve ser correspondente à chave pública que está no certificado.
- A chave privada deve estar no formato PEM.
- A chave privada não pode ser criptografada com senha.

Se o AWS Certificate Manager (ACM) forneceu o certificado, ele não liberará a chave privada. A chave privada é armazenada no ACM para uso pelos serviços da AWS integrados a ele.

Permissões

É necessário ter permissão para usar e importar o certificado SSL/TLS. Se você estiver usando o AWS Certificate Manager (ACM), recomendamos usar as permissões do AWS Identity and Access Management para restringir o acesso aos certificados. Para obter mais informações, consulte [Gerenciamento de identidade e acesso](#) no Manual do usuário do AWS Certificate Manager.

Tamanho da chave do certificado

O tamanho de chave de certificado aceito pelo CloudFront depende dos tipos da chave e do certificado.

Para certificados RSA:

O CloudFront é compatível com chaves RSA de 1.024 bits, 2.048 bits, 3.072 bits e 4.096 bits. O tamanho máximo de chave de um certificado RSA usado com o CloudFront é de 4.096 bits.

Observe que o ACM emite certificados RSA com chaves de até 2048 bits. Para usar um certificado RSA de 3.072 bits ou 4.096 bits, você precisa obter o certificado externamente e importá-lo para o ACM a fim de que ele esteja disponível para uso com o CloudFront.

Para obter informações sobre como determinar o tamanho da chave RSA, consulte [Determinar o tamanho da chave pública em um certificado RSA SSL/TLS](#).

Para certificados ECDSA:

O CloudFront é compatível com chaves de 256 bits. Para usar um certificado ECDSA no ACM para exigir HTTPS entre visualizadores e o CloudFront, use a curva elíptica prime256v1.

Tipos de certificados compatíveis

O CloudFront é compatível com todos os tipos de certificados emitidos por uma autoridade de certificação confiável.

Data de expiração e renovação do certificado

Se estiver usando certificados obtidos de uma autoridade de certificação (CA) de terceiros, você será responsável por monitorar as datas de validade e renovar os certificados importados para o AWS Identity and Access Management (ACM) ou carregá-los para o armazenamento de certificados do AWS Certificate Manager antes que eles expirem.

Se você estiver usando certificados fornecidos pelo ACM, ele gerenciará as renovações. Para obter mais informações, consulte [Renovação gerenciada](#) no Guia do usuário do AWS Certificate Manager.

Nomes de domínio na distribuição do CloudFront e no certificado

Quando você usa uma origem personalizada, o certificado SSL/TLS na sua origem incluirá um nome de domínio no campo Common Name (Nome comum) e, possivelmente, vários outros no campo Subject Alternative Names (Nomes alternativos do sujeito). (O CloudFront oferece suporte a caracteres curinga em nomes de domínio de certificados.)

Um dos nomes de domínio do certificado deve ser correspondente ao nome de domínio especificado em "Origin Domain Name". Se nenhum nome de domínio corresponder, o CloudFront retornará um código de status HTTP 502 (Bad Gateway) para o visualizador.

Important

Ao adicionar um nome de domínio alternativo a uma distribuição, o CloudFront verifica se o nome de domínio alternativo está coberto pelo certificado que foi anexado. O certificado deve abranger o nome de domínio alternativo no campo de nome alternativo de assunto (SAN) do certificado. Isso significa que o campo SAN deve conter uma correspondência exata para o nome de domínio alternativo ou conter um curinga no mesmo nível do nome de domínio alternativo que você está adicionando.

Para obter mais informações, consulte [Requisitos para o uso de nomes de domínio alternativos](#).

Versão mínima do protocolo SSL/TLS

Se estiver usando endereços IP dedicados, defina a versão mínima do protocolo SSL/TLS para a conexão entre os visualizadores e o CloudFront escolhendo uma política de segurança.

Para obter mais informações, consulte [Política de segurança \(versão mínima de SSL/TLS\)](#) no tópico [Referência de configurações da distribuição](#).

Versões de HTTP compatíveis

Se você associar um certificado a mais de uma distribuição do CloudFront, todas as distribuições associadas ao certificado deverão usar a mesma opção para [Versões de HTTP compatíveis](#). Especifique essa opção ao criar ou atualizar uma distribuição do CloudFront.

Cotas no uso de certificados SSL/TLS com o CloudFront (somente HTTPS entre visualizadores e o CloudFront)

Observe as cotas a seguir para o uso de certificados SSL/TLS com o CloudFront. Essas cotas se aplicam somente a certificados SSL/TLS provisionados usando o AWS Certificate Manager (ACM), importados para o ACM ou carregados no armazenamento de certificados do IAM para a comunicação HTTPS entre visualizadores e o CloudFront.

Para ter mais informações, consulte [Aumentar as cotas de certificados SSL/TLS](#).

Número máximo de certificados por distribuição do CloudFront

É possível associar, no máximo, um certificado SSL/TLS a cada distribuição do CloudFront.

Número máximo de certificados que podem ser importados para o ACM ou carregados no armazenamento de certificados do IAM

Se você obteve os certificados SSL/TLS de uma CA de terceiros, deverá armazená-los em um dos seguintes locais:

- **AWS Certificate Manager:** para a cota atual para o número de certificados do ACM, consulte [Cotas](#) no Guia do usuário do AWS Certificate Manager. A cota informada é um total que inclui os certificados provisionados usando o ACM e os certificados importados para o ACM.
- **Armazenamento de certificados do IAM:** para a cota atual (anteriormente conhecida como limite) sobre o número de certificados que podem ser carregados no armazenamento de certificados do IAM para uma conta da AWS, consulte [Limites do IAM e do STS](#) no Guia do usuário do IAM. É possível solicitar um aumento de cotas no console do Service Quotas.

Número máximo de certificados por conta da AWS (somente endereços IP dedicados)

Se você quiser atender a solicitações HTTPS usando endereços IP dedicados, observe:

- Por padrão, o CloudFront concede permissão para usar dois certificados com sua conta da AWS: um para uso diário e um para quando você precisar alternar certificados para várias distribuições.
- Se precisar de mais de dois certificados SSL/TLS personalizados para a conta da AWS, solicite uma cota maior no console do Service Quotas.

Usar o mesmo certificado para distribuições do CloudFront criadas com diferentes contas da AWS

Se estiver usando uma CA de terceiros e quiser usar o mesmo certificado com várias distribuições do CloudFront criadas usando diferentes contas da AWS, você deverá importar o

certificado para o ACM ou fazer upload dele no armazenamento de certificados do IAM de cada conta da AWS.

Se estiver usando certificados fornecidos pelo ACM, você não poderá configurar o CloudFront para usar os certificados criados por outra conta da AWS.

Usar o mesmo certificado para o CloudFront e para outros serviços da AWS

Se adquiriu um certificado de uma autoridade de certificação confiável, como Comodo, DigiCert ou Symantec, você poderá usar o mesmo certificado para o CloudFront e para outros produtos da AWS. Se você estiver importando o certificado para o ACM, precisará fazê-lo somente uma vez para usá-lo para vários produtos da AWS.

Se estiver usando certificados fornecidos pelo ACM, eles serão armazenados nele.

Usar o mesmo certificado para várias distribuições do CloudFront

É possível usar o mesmo certificado para uma ou todas as distribuições do CloudFront que você estiver usando para atender a solicitações HTTPS. Observe o seguinte:

- Você pode usar o mesmo certificado para atender a solicitações usando endereços IP dedicados e a SNI.
- Você pode associar apenas um certificado a cada distribuição.
- Cada distribuição deve incluir um ou mais nomes de domínio alternativos que também aparecem no campo Common Name (Nome comum) ou Subject Alternative Names (Nomes alternativos de assunto) do certificado.
- Se você estiver atendendo a solicitações HTTPS usando endereços IP dedicados e criou todas as suas distribuições usando a mesma conta da AWS, poderá reduzir significativamente seus custos usando o mesmo certificado para todas as distribuições. O CloudFront cobra por certificado, não por distribuição.

Por exemplo, imagine que você criou três distribuições usando a mesma conta da AWS e usa o mesmo certificado para todas elas. Você será cobrado apenas uma taxa pelo uso de endereços IP dedicados.

No entanto, se você atende a solicitações HTTPS usando endereços IP dedicados e o mesmo certificado para criar distribuições do CloudFront em diferentes contas da AWS, é cobrada a taxa pelo uso de endereços IP dedicados de cada conta. Por exemplo, se você criar três distribuições usando três contas diferentes da AWS e usar o mesmo certificado para todas as distribuições, será cobrada a taxa total pelo uso de endereços IP dedicados de cada conta.

Configurar nomes de domínio alternativos e HTTPS

Para usar nomes de domínio alternativos nos URLs dos seus arquivos e usar HTTPS entre os visualizadores e o CloudFront, execute os procedimentos aplicáveis.

Tópicos

- [Receber um certificado SSL/TLS](#)
- [Importar um certificado SSL/TLS](#)
- [Atualizar a distribuição do CloudFront](#)

Receber um certificado SSL/TLS

Obtenha um certificado SSL/TLS, se você ainda não tiver um. Para obter mais informações, consulte a documentação aplicável:

- Para usar um certificado fornecido pelo AWS Certificate Manager (ACM), consulte o [Guia do usuário do AWS Certificate Manager](#). Em seguida, vá para [Atualizar a distribuição do CloudFront](#).

Note

É recomendável que você use o ACM para provisionar, gerenciar e implantar os certificados SSL/TLS nos recursos gerenciados da AWS. Você deve solicitar um certificado ACM na região Leste dos EUA (Norte da Virgínia).

- Para obter um certificado de uma autoridade de certificação (CA) terceirizada, consulte a documentação fornecida por ela. Se você tiver o certificado, continue no próximo procedimento.

Importar um certificado SSL/TLS

Se você obteve seu certificado de uma CA de terceiros, importe-o para o ACM ou faça upload dele no armazenamento de certificados do IAM:

ACM (recomendado)

O ACM permite importar certificados de terceiros pelo console do ACM, bem como de forma programática. Para obter informações sobre como importar um certificado para o ACM, consulte [Importação de certificados no AWS Certificate Manager](#) no Guia do usuário do AWS Certificate Manager. Você deve importar o certificado na região Leste dos EUA (Norte da Virgínia).

Armazenamento de certificados do IAM

(Não recomendado) Use o comando da AWS CLI a seguir para carregar o certificado de terceiros no armazenamento de certificados do IAM.

```
aws iam upload-server-certificate \  
  --server-certificate-name CertificateName \  
  --certificate-body file://public_key_certificate_file \  
  --private-key file://privatekey.pem \  
  --certificate-chain file://certificate_chain_file \  
  --path /cloudfront/path/
```

Observe o seguinte:

- Conta da AWS: é necessário fazer upload do certificado no armazenamento de certificados do IAM com a mesma conta da AWS usada para criar sua distribuição do CloudFront.
- Parâmetro `--path`: ao fazer upload do certificado no IAM, o valor do parâmetro `--path` (caminho do certificado) deve começar com `/cloudfront/`, por exemplo, `/cloudfront/production/` ou `/cloudfront/test/`. O caminho deve terminar com `"/`.
- Certificados existentes: é necessário especificar valores para os parâmetros `--server-certificate-name` e `--path` diferentes dos valores associados aos certificados existentes.
- Uso do console do CloudFront: o valor especificado para o parâmetro `--server-certificate-name` na AWS CLI, por exemplo, `myServerCertificate`, aparece na lista SSL Certificate (Certificado SSL) no console do CloudFront.
- Uso da API do CloudFront: anote a string alfanumérica retornada pela AWS CLI. Por exemplo, `AS1A2M3P4L5E67SIIXR3J`. Esse é o valor especificado no elemento `IAMCertificateId`. O ARN do IAM, que também é retornado pela CLI, não é necessário.

Para obter mais informações sobre a AWS CLI, consulte o [Guia do usuário da AWS Command Line Interface](#) e a [Referência de comandos da AWS CLI](#).

Atualizar a distribuição do CloudFront

Para atualizar as configurações da sua distribuição, execute o seguinte procedimento:

Como configurar sua distribuição do CloudFront para nomes de domínio alternativos

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.

2. Escolha o ID da distribuição que você deseja atualizar.
3. Na guia General, escolha Edit.
4. Atualize os seguintes valores:

Nome de domínio alternativo (CNAMEs)

Selecione Add item (Adicionar item) para adicionar os nomes de domínio alternativos aplicáveis. Separe os nomes de domínio com vírgulas ou digite cada nome de domínio em uma nova linha.

Certificado SSL personalizado

Selecione um certificado na lista suspensa.

Até 100 certificados estão listados aqui. Se você tiver mais de 100 certificados e não estiver visualizando o certificado que quer adicionar, digite um ARN de certificado no campo para escolhê-lo.

Se você fez o upload de um certificado para o armazenamento de certificados do IAM, mas ele não está listado e você não puder escolhê-lo digitando o nome no campo, revise o procedimento [Importar um certificado SSL/TLS](#) para verificar se você carregou corretamente o certificado.

Important

Depois que associar seu certificado SSL/TLS à sua distribuição do CloudFront, não o exclua do ACM nem do armazenamento de certificados do IAM enquanto não removê-lo de todas as distribuições e todas as distribuições sejam implantadas.

5. Escolha Salvar alterações.
6. Configurar o CloudFront para exigir HTTPS entre visualizadores e o CloudFront:
 - a. Na guia Behaviors, escolha o comportamento de cache que você deseja atualizar e, em seguida, escolha Edit.
 - b. Especifique um dos seguintes valores para Viewer Protocol Policy:

Redirect HTTP to HTTPS

Os visualizadores podem usar os dois protocolos, mas solicitações HTTP são automaticamente redirecionadas para HTTPS. O CloudFront retorna o código de status

HTTP 301 (Moved Permanently) junto com o novo URL de HTTPS. Depois, o visualizador reenvia a solicitação para o CloudFront usando o URL de HTTPS.

 Important

O CloudFront não redireciona solicitações DELETE, OPTIONS, PATCH, POST ou PUT de HTTP para HTTPS. Se você configurar um comportamento de cache para redirecionar para HTTPS, o CloudFront responderá a solicitações HTTP DELETE, OPTIONS, PATCH, POST ou PUT desse comportamento de cache com o código de status HTTP 403 (Forbidden).

Quando um visualizador faz uma solicitação HTTP que é redirecionada para uma solicitação HTTPS, o CloudFront cobra pelas duas solicitações. Para a solicitação HTTP, a cobrança é somente pela solicitação e cabeçalhos retornados pelo CloudFront para o visualizador. Para a solicitação HTTPS, a cobrança é pela solicitação e pelos cabeçalhos e arquivo retornados por sua origem.

HTTPS Only

Os visualizadores só podem acessar seu conteúdo se estiverem usando HTTPS. Se um visualizador enviar uma solicitação HTTP, em vez de HTTPS, o CloudFront retornará o código de status HTTP 403 (Forbidden) e não retornará o arquivo.

- c. Escolha Yes, Edit.
 - d. Repita as etapas "a" a "c" para cada comportamento de cache adicional para o qual você deseja exigir HTTPS entre os visualizadores e o CloudFront.
7. Antes de usar a configuração atualizada em um ambiente de produção, confirme:
- Se o padrão de caminho de cada comportamento de cache se aplica apenas às solicitações nas quais os visualizadores devem usar HTTPS.
 - Se os comportamentos de cache estão listados na ordem em que você deseja que o CloudFront os avalie. Para obter mais informações, consulte [Padrão de caminho](#).
 - Se os comportamentos de cache estão roteando as solicitações para as origens corretas.

Determinar o tamanho da chave pública em um certificado RSA SSL/TLS

Ao usar nomes de domínio alternativos do CloudFront e HTTPS, o tamanho máximo da chave pública em um certificado RSA SSL/TLS é de 4.096 bits. (Esse é o tamanho da chave, e não é o número de caracteres da chave pública.) Se você usa o AWS Certificate Manager para os certificados, embora o ACM seja compatível com chaves RSA maiores, não é possível usar as chaves maiores com o CloudFront.

Você pode determinar o tamanho da chave pública RSA executando o seguinte comando OpenSSL:

```
openssl x509 -in path and filename of SSL/TLS certificate -text -noout
```

Onde:

- `-in` especifica o caminho e nome do arquivo do certificado RSA SSL/TLS.
- `-text` faz com que o OpenSSL exiba o tamanho da chave pública RSA em bits.
- `-noout` impede que o OpenSSL exiba a chave pública.

Exemplos de resultado:

```
Public-Key: (2048 bit)
```

Aumentar as cotas de certificados SSL/TLS

Há cotas para o número de certificados SSL que você pode importar para o AWS Certificate Manager (ACM) ou dos quais fazer upload para o AWS Identity and Access Management (IAM). Também há uma cota para o número de certificados SSL/TLS que podem ser usados com uma Conta da AWS ao configurar o CloudFront para atender a solicitações HTTPS usando endereços IP dedicados. No entanto, você pode solicitar cotas mais altas.

Tópicos

- [Aumentar a cota de certificados importados para o ACM](#)
- [Aumentar a cota de certificados carregados para o IAM](#)
- [Aumentar a cota de certificados usados com endereços IP dedicados](#)

Aumentar a cota de certificados importados para o ACM

Para obter a cota do número de certificados que podem ser importados para o ACM, consulte [Cotas](#) no Guia do usuário do AWS Certificate Manager.

Para solicitar aumento de cota, use o console do Service Quotas. Para obter mais informações, consulte [Solicitando um Aumento de Cota](#) no Guia do Usuário do Service Quotas.

Aumentar a cota de certificados carregados para o IAM

Para saber a cota (anteriormente conhecida como limite) sobre o número de certificados que podem ser carregados no IAM, consulte [Limites do IAM e do STS](#) no Guia do usuário do IAM.

Para solicitar aumento de cota, use o console do Service Quotas. Para obter mais informações, consulte [Solicitando um Aumento de Cota](#) no Guia do Usuário do Service Quotas.

Aumentar a cota de certificados usados com endereços IP dedicados

Em relação à cota do número de certificados SSL que podem ser usados para cada Conta da AWS ao atender a solicitações HTTPS usando endereços IP dedicados, consulte [Cotas para certificados SSL](#).

Para solicitar aumento de cota, use o console do Service Quotas. Para obter mais informações, consulte [Solicitando um Aumento de Cota](#) no Guia do Usuário do Service Quotas.

Trocar certificados SSL/TLS

Se você estiver usando certificados fornecidos pelo AWS Certificate Manager (ACM), não precisará fazer o rodízio de certificados SSL/TLS. O ACM gerencia as renovações dos certificados para você. Para obter mais informações, consulte [Renovação gerenciada](#) no Guia do usuário do AWS Certificate Manager.

Note

O ACM não gerencia as renovações de certificados adquiridos de autoridades de certificação de terceiros e importados para o ACM.

Se estiver usando uma autoridade de certificação de terceiros e tiver importado certificados para o ACM (recomendado) ou feito upload desses certificados no armazenamento de certificados do

IAM, ocasionalmente, será necessário substituir um certificado por outro. Por exemplo, é necessário substituir um certificado quando a data de expiração do certificado está se aproximando.

Important

Se tiver configurado o CloudFront para atender a solicitações HTTPS usando endereços IP dedicados, você poderá receber uma cobrança adicional proporcional pelo uso de um ou mais certificados adicionais ao alterná-los. Recomendamos que você atualize suas distribuições imediatamente para minimizar o custo adicional.

Trocar certificados SSL/TLS

Para alternar certificados, execute o procedimento a seguir. Os visualizadores podem continuar acessando seu conteúdo enquanto você alterna os certificados e após a conclusão do processo.

Para alternar certificados SSL/TLS

1. [Aumentar as cotas de certificados SSL/TLS](#) para determinar se você precisa de permissão para usar mais certificados SSL. Em caso afirmativo, solicite permissão e aguarde a concessão antes de continuar na etapa 2.
2. Importar o novo certificado para o ACM ou fazer upload dele no IAM. Para obter mais informações, consulte [Importar um certificado SSL/TLS](#) no Guia do desenvolvedor do Amazon CloudFront.
3. Atualize suas distribuições individualmente para usar o novo certificado. Para obter mais informações, consulte [Listar, visualizar e atualizar distribuições do CloudFront](#) no Guia do desenvolvedor do Amazon CloudFront.
4. (Opcional) Depois que atualizar todas as suas distribuições do CloudFront, você poderá excluir o certificado antigo do ACM ou do IAM.

Important

Não exclua um certificado SSL/TLS enquanto não os remover de todas as distribuições e o status das distribuições atualizadas não for alterado para Deployed.

Reverter um certificado SSL/TLS personalizado para o certificado padrão do CloudFront

Se configurou o CloudFront para usar HTTPS entre visualizadores e o CloudFront e configurou o CloudFront para usar um certificado SSL/TLS personalizado, você poderá alterar sua configuração para usar o certificado SSL/TLS padrão do CloudFront. O processo dependerá se você já tiver usado ou não a distribuição para distribuir seu conteúdo:

- Se você não usou sua distribuição para distribuir o conteúdo, poderá simplesmente alterar a configuração. Para obter mais informações, consulte [Atualizar uma distribuição](#).
- Caso contrário, será necessário criar uma distribuição do CloudFront e alterar os URLs dos seus arquivos para reduzir ou eliminar o tempo de indisponibilidade do conteúdo. Para fazer isso, execute o procedimento a seguir.

Reverter para o certificado padrão do CloudFront

O procedimento a seguir mostra como reverter um certificado SSL/TLS personalizado para o certificado padrão do CloudFront.

Como reverter para o certificado padrão do CloudFront

1. Crie uma distribuição do CloudFront com a configuração desejada. Em SSL Certificate (Certificado SSL), escolha Default CloudFront Certificate (Certificado padrão do CloudFront) (*.cloudfront.net).

Para ter mais informações, consulte [Criar uma distribuição](#).

2. Para arquivos sendo distribuídos usando o CloudFront, atualize os URLs na aplicação para usar o nome de domínio atribuído pelo CloudFront à nova distribuição. Por exemplo, altere `https://www.example.com/images/logo.png` para `https://d111111abcdef8.cloudfront.net/images/logo.png`.
3. Exclua a distribuição associada a um certificado SSL/TLS personalizado ou atualize-a para alterar o valor de SSL Certificate (Certificado SSL) para Default CloudFront Certificate (Certificado padrão do CloudFront) (*.cloudfront.net). Para obter mais informações, consulte [Atualizar uma distribuição](#).

⚠ Important

Enquanto você não concluir essa etapa, a AWS continuará cobrando pelo uso de um certificado SSL/TLS personalizado.

4. (Opcional) Exclua seu certificado SSL/TLS personalizado.
 - a. Execute o comando `list-server-certificates` da AWS CLI para obter o ID do certificado que você deseja excluir. Para obter mais informações, consulte [list-server-certificates](#) na Referência de comandos da AWS CLI.
 - b. Execute o comando `delete-server-certificate` da AWS CLI para excluir o certificado. Para ter mais informações, consulte [delete-server-certificate](#) na Referência de comandos da AWS CLI.

Alternar de um certificado SSL/TLS personalizado com endereços IP dedicados para SNI

Se configurou o CloudFront para usar um certificado SSL/TLS personalizado com endereços IP dedicados, você poderá alternar para um certificado SSL/TLS personalizado com SNI e eliminar a cobrança associada a endereços IP dedicados. O procedimento a seguir mostra como fazer isso.

⚠ Important

Essa atualização na configuração do CloudFront não afeta os visualizadores que oferecem suporte à SNI. Os visualizadores podem acessar seu conteúdo antes e depois da alteração, bem como enquanto a alteração está se propagando para pontos de presença do CloudFront. Os visualizadores não compatíveis com SNI não poderão acessar seu conteúdo após a alteração. Para ter mais informações, consulte [Escolher como o CloudFront atende a solicitações HTTPS](#).

Alternar do certificado personalizado para SNI

O procedimento a seguir mostra como alternar de um certificado SSL/TLS personalizado com endereços IP dedicados para SNI.

Para alternar de um certificado SSL/TLS personalizado com endereços IP dedicados para SNI

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha o ID da distribuição que você deseja visualizar ou atualizar.
3. Escolha Distribution Settings.
4. Na guia General, escolha Edit.
5. Altere a configuração de Certificado SSL personalizado para Somente clientes que comportam Server Name Indication (SNI).
6. Escolha Yes, Edit.

Veicular conteúdo privado com URLs e cookies assinados

Várias empresas que distribuem conteúdo pela Internet querem restringir o acesso a documentos, dados de negócio, streams de mídia ou conteúdo destinado a usuários selecionados, por exemplo, que pagaram uma taxa. Para fornecer esse conteúdo privado com segurança usando o CloudFront, é possível:

- Solicitar que os usuários acessem o conteúdo privado usando signed URLs ou signed cookies especiais do CloudFront.
- Solicitar que os usuários acessem seu conteúdo usando URLs do CloudFront e não URLs que acessam conteúdo diretamente no servidor de origem (por exemplo, o Amazon S3 ou um servidor HTTP privado). Não é necessário solicitar URLs do CloudFront, mas recomendamos fazer isso para impedir que os usuários ignorem as restrições especificadas em signed URLs ou signed cookies.

Para ter mais informações, consulte [Restringir o acesso a arquivos](#).

Como fornecer conteúdo privado

Para configurar o CloudFront para fornecer conteúdo privado, siga estas etapas:

1. (Opcional, mas recomendado) Solicite que seus usuários acessem o conteúdo apenas pelo CloudFront. O método usado dependerá se você estiver usando o Amazon S3 ou origens personalizadas:

- Amazon S3: consulte [the section called “Restringir o acesso a uma origem do Amazon Simple Storage Service”](#).
- Origem personalizada: consulte [Restringir o acesso a arquivos em origens personalizadas](#).

As origens personalizadas incluem o Amazon EC2, buckets do Amazon S3 configurados como endpoints de site, o Elastic Load Balancing e seus próprios servidores web HTTP.

2. Especifique os grupos de chaves confiáveis ou assinantes confiáveis que você deseja usar para criar signed URLs ou cookies. Recomendamos que você use grupos de chaves confiáveis. Para obter mais informações, consulte [Especificar os assinantes que podem criar URLs e cookies assinados](#).
3. Crie seu aplicativo para responder a solicitações de usuários autorizados com signed URLs ou cabeçalhos Set-Cookie que definem signed cookies. Siga as etapas em um dos tópicos abaixo:
 - [Usar URLs assinados](#)
 - [Usar cookies assinados](#)

Se você não tiver certeza sobre qual método usar, consulte [Decidir usar URLs ou cookies assinados](#).

Tópicos

- [Restringir o acesso a arquivos](#)
- [Especificar os assinantes que podem criar URLs e cookies assinados](#)
- [Decidir usar URLs ou cookies assinados](#)
- [Usar URLs assinados](#)
- [Usar cookies assinados](#)
- [Comandos do Linux e OpenSSL para criptografia e codificação base64](#)
- [Exemplos de código para criar uma assinatura para um signed URL](#)

Restringir o acesso a arquivos

É possível controlar o acesso do usuário ao seu conteúdo privado de duas maneiras:

- [Restringir o acesso a arquivos em caches do CloudFront.](#)
- Restrinja o acesso a arquivos na sua origem de uma das seguintes maneiras:
 - [Configure um controle de acesso à origem \(OAC\) para seu bucket do Amazon S3.](#)
 - [Configurar cabeçalhos personalizados para um servidor HTTP privado \(uma origem personalizada\).](#)

Restringir o acesso a arquivos em caches do CloudFront

É possível configurar o CloudFront para exigir que os usuários acessem seus arquivos usando signed URLs ou signed cookies. Depois, desenvolva sua aplicação para criar e distribuir signed URLs para usuários autenticados ou para enviar cabeçalhos Set-Cookie que definem signed cookies para usuários autenticados. (Para conceder a alguns usuários acesso de longo prazo a um pequeno número de arquivos, você também pode criar URLs assinados manualmente.)

Ao criar signed URLs ou cookies para controlar o acesso a seus arquivos, você pode especificar as seguintes restrições:

- Uma data e hora de expiração do URL.
- (Opcional) A data e a hora em que o URL se torna válido.
- (Opcional) O endereço IP ou os endereços dos computadores que podem ser usados para acessar seu conteúdo.

É adicionado hash ou assinatura a parte de um signed URL ou signed cookie usando a chave privada de um par de chaves públicas/privadas. Quando alguém usa um signed URL ou signed cookie para acessar um arquivo, o CloudFront compara as partes assinada e não assinada do URL ou cookie. Se elas não corresponderem, o CloudFront não fornecerá o arquivo.

É necessário usar o RSA-SHA1 para assinar URLs ou cookies. O CloudFront não aceita outros algoritmos.

Restringir o acesso a arquivos em buckets do Amazon S3

Também é possível proteger o conteúdo no bucket do Amazon S3 para que os usuários possam acessá-lo por meio da distribuição especificada do CloudFront, mas não possam acessá-lo diretamente usando URLs do Amazon S3. Isso impede que alguém ignore o CloudFront e use o URL do Amazon S3 para obter o conteúdo ao qual você deseja restringir o acesso. Essa etapa não exige o uso de signed URLs, mas recomendamos que você o faça.

Para solicitar que os usuários acessem seu conteúdo por URLs do CloudFront, siga estas etapas:

- Conceda uma permissão de controle de acesso à origem do CloudFront para ler os arquivos no bucket do S3.
- Crie o controle de acesso à origem e associe-o à sua distribuição do CloudFront.
- Remova a permissão para usar URLs do Amazon S3 para ler os arquivos de todas as demais pessoas.

Para ter mais informações, consulte [the section called “Restringir o acesso a uma origem do Amazon Simple Storage Service”](#).

Restringir o acesso a arquivos em origens personalizadas

Se você usar uma origem personalizada, pode configurar os cabeçalhos personalizados para restringir o acesso. Para o CloudFront obter seus arquivos de uma origem personalizada, os arquivos devem estar acessíveis pelo CloudFront usando uma solicitação padrão HTTP (ou HTTPS). Mas, ao usar cabeçalhos personalizados, é possível restringir ainda mais o acesso ao conteúdo para que os usuários possam acessá-lo apenas por meio do CloudFront, e não diretamente. Essa etapa não exige o uso de signed URLs, mas recomendamos que você o faça.

Para exigir que os usuários acessem o conteúdo por meio do CloudFront, altere as seguintes configurações em suas distribuições do CloudFront:

Cabeçalhos personalizados de origem

Configure o CloudFront para encaminhar cabeçalhos personalizados para sua origem. Consulte [Configurar o CloudFront para adicionar cabeçalhos personalizados às solicitações de origem](#).

Política de protocolo do visualizador

Configure a distribuição para exigir que os visualizadores usem HTTPS para acessar o CloudFront. Consulte [Política de protocolo do visualizador](#).

Política de protocolo da origem

Configure sua distribuição para exigir que o CloudFront use o mesmo protocolo que os visualizadores para encaminhar solicitações para a origem. Consulte [Protocolo \(somente origens personalizadas\)](#).

Depois de fazer essas alterações, atualize sua aplicação na origem personalizada para aceitar somente solicitações que incluam os cabeçalhos personalizados que você configurou o CloudFront para enviar.

A combinação de Viewer Protocol Policy (Política de protocolo do visualizador) e Origin Protocol Policy (Política de protocolo da origem) garante que os cabeçalhos personalizados sejam criptografados em trânsito. No entanto, recomendamos que você execute periodicamente o seguinte, para alternar os cabeçalhos personalizados encaminhados pelo CloudFront para sua origem:

1. Atualize sua distribuição do CloudFront para começar a encaminhar um novo cabeçalho para sua origem personalizada.
2. Atualize sua aplicação para aceitar o novo cabeçalho como confirmação de que a solicitação é proveniente do CloudFront.
3. Quando as solicitações não incluírem mais o cabeçalho que você estiver substituindo, atualize a aplicação para não aceitar mais o cabeçalho antigo como confirmação de que a solicitação é proveniente do CloudFront.

Especificar os assinantes que podem criar URLs e cookies assinados

Tópicos

- [Escolher entre grupos de chaves confiáveis \(recomendado\) e Contas da AWS](#)
- [Criar pares de chaves para os assinantes](#)
- [Reformatar a chave privada \(somente .NET e Java\)](#)
- [Adicionar um assinante a uma distribuição](#)
- [Alternar pares de chaves](#)

Para criar signed URLs ou cookies, é necessário um assinante. Um assinante é um grupo de chaves confiável criado no CloudFront ou uma conta da AWS que contenha um par de chaves do CloudFront. Recomendamos que você use grupos de chaves confiáveis com signed URLs e cookies. Para obter mais informações, consulte [Escolher entre grupos de chaves confiáveis \(recomendado\) e Contas da AWS](#).

O assinante tem duas finalidades:

- Assim que você adicionar o assinante à sua distribuição, o CloudFront começará a exigir que os visualizadores usem signed URLs ou signed cookies para acessar seus arquivos.

- Ao criar signed URLs ou cookies, você usa a chave privada do par de chaves do assinante para assinar uma parte do URL ou do cookie. Quando alguém solicita um arquivo restrito, o CloudFront compara a assinatura no URL ou cookie com o URL ou cookie não assinado, para verificar se ele não foi adulterado. O CloudFront também verifica se o URL ou cookie é válido, ou seja, se a data e hora de expiração não passou, por exemplo.

Ao especificar um assinante, você também especifica indiretamente os arquivos que exigem signed URLs ou cookies adicionando o assinante a um comportamento de cache. Caso sua distribuição tenha somente um comportamento de cache, os visualizadores deverão usar signed URLs ou cookies para acessar qualquer arquivo na distribuição. Se criar vários comportamentos de cache e adicionar assinantes a alguns deles, mas não a outros, você poderá solicitar que os visualizadores usem signed URLs ou cookies para acessar alguns arquivos e não outros.

Para especificar os assinantes (as chaves privadas) com permissão para criar signed URLs ou signed cookies e adicionar os assinantes à sua distribuição do CloudFront, execute as seguintes tarefas:

1. Decida se pretende utilizar um grupo de chaves confiável ou uma Conta da AWS como assinante. Recomendamos o uso de um grupo de chaves confiável. Para obter mais informações, consulte [Escolher entre grupos de chaves confiáveis \(recomendado\) e Contas da AWS](#).
2. Para o assinante que você escolheu na etapa 1, crie um par de chaves pública/privada. Para obter mais informações, consulte [Criar pares de chaves para os assinantes](#).
3. Se você estiver usando .NET ou Java para criar signed URLs ou cookies, reformate a chave privada. Para obter mais informações, consulte [Reformatar a chave privada \(somente .NET e Java\)](#).
4. Na distribuição para a qual você está criando signed URLs ou cookies, especifique o assinante. Para ter mais informações, consulte [Adicionar um assinante a uma distribuição](#).

Escolher entre grupos de chaves confiáveis (recomendado) e Contas da AWS

Para usar signed URLs ou cookies, é necessário um assinante. Um assinante é um grupo de chaves confiável criado no CloudFront ou uma Conta da AWS que contenha um par de chaves do CloudFront. Recomendamos que você use grupos de chaves confiáveis, pelos seguintes motivos:

- Com os grupos de chaves do CloudFront, não é necessário usar o usuário root da conta da AWS para gerenciar as chaves públicas para URLs assinados e cookies do CloudFront. As

[práticas recomendadas da AWS](#) aconselham não usar o usuário root, exceto quando estritamente necessário.

- Com os grupos de chaves do CloudFront, é possível gerenciar chaves públicas, grupos de chaves e assinantes confiáveis usando a API do CloudFront. É possível usar a API para automatizar a criação de chaves e a alternância de chaves. Quando você usa o usuário-raiz da AWS, é necessário usar o AWS Management Console para gerenciar pares de chaves do CloudFront, por isso o processo não pode ser automatizado.
- Como você pode gerenciar grupos de chaves com a API do CloudFront, também é possível usar as políticas de permissões do AWS Identity and Access Management (IAM) para limitar as ações permitidas aos diferentes usuários. Por exemplo, é possível permitir que os usuários façam upload de chaves públicas, mas não excluí-las. Ou, é possível permitir que os usuários excluam chaves públicas, mas somente quando determinadas condições forem atendidas, como usar autenticação multifator, enviar a solicitação de uma determinada rede ou enviar a solicitação dentro de um determinado intervalo de data e hora.
- Com grupos de chaves do CloudFront, é possível associar um número maior de chaves públicas à sua distribuição do CloudFront, proporcionando mais flexibilidade na forma como você usa e gerencia as chaves públicas. Por padrão, é possível associar até quatro grupos de chaves a uma única distribuição e ter até cinco chaves públicas em um grupo de chaves.

Ao usar a conta da AWS do usuário-raiz para gerenciar pares de chaves do CloudFront, você só poderá ter até dois pares de chaves ativos do CloudFront por conta da AWS.

Criar pares de chaves para os assinantes

Cada assinante usado para criar signed URLs ou signed cookies do CloudFront deve ter um par de chaves pública/privada. O assinante usa sua chave privada para assinar o URL ou os cookies e o CloudFront usa a chave pública para verificar a assinatura.

A maneira como você cria um par de chaves depende se você usa um grupo de chaves confiável como assinante (recomendado) ou um par de chaves do CloudFront. Para obter mais informações, consulte as seções a seguir. O par de chaves que criar deve atender aos seguintes requisitos:

- Deve ser um par de chaves SSH-2 RSA.
- Ele deve estar no formato PEM codificado em base64.
- Deve ser um par de chaves de 2048 bits.

Para ajudar a proteger suas aplicações, recomendamos que você altere os pares de chaves periodicamente. Para obter mais informações, consulte [Alternar pares de chaves](#).


Criar um par de chaves para um grupo de chaves confiável (recomendado)

Para criar um par de chaves para um grupo de chaves confiável, execute as seguintes etapas:

1. Crie o par de chaves pública/privada.
2. Faça upload da chave pública no CloudFront.
3. Adicione a chave pública a um grupo de chaves do CloudFront.

Para obter mais informações, consulte os procedimentos a seguir.

Para criar um par de chaves

 Note

As etapas a seguir usam OpenSSL como um exemplo de uma maneira de criar um par de chaves. Há várias outras maneiras de criar um par de chaves RSA.

1. O comando de exemplo a seguir usa OpenSSL para gerar um par de chaves RSA com um tamanho de 2.048 bits e salvar no arquivo chamado `private_key.pem`.

```
openssl genrsa -out private_key.pem 2048
```

2. O arquivo resultante contém a chave pública e a privada. O comando de exemplo a seguir extrai a chave pública do arquivo chamado `private_key.pem`.

```
openssl rsa -pubout -in private_key.pem -out public_key.pem
```

Faça upload da chave pública (no arquivo `public_key.pem`) posteriormente, no procedimento a seguir.

Como carregar a chave pública no CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No menu de navegação, escolha Public keys (Chaves públicas).
3. Selecione Create public key (Criar chave pública).
4. Na janela Create public key (Criar chave pública), faça o seguinte:
 - a. Em Key name (Nome da chave), digite um nome para identificar a chave pública.
 - b. Em Key value (Valor da chave), cole a chave pública. Se você seguiu as etapas no procedimento anterior, a chave pública está no arquivo chamado `public_key.pem`. Para copiar e colar o conteúdo da chave pública, é possível:
 - Usar o comando `cat` na linha de comando do macOS ou do Linux, da seguinte forma:

```
cat public_key.pem
```

Copie a saída desse comando e cole-a no campo Key value (Valor da chave).

- b. Abra o arquivo `public_key.pem` com um editor de texto simples, como o Notepad (no Windows) ou o TextEdit (no macOS). Copie o conteúdo do arquivo e cole-o no campo Key value (Valor da chave).
- c. (Opcional) Em Comment (Comentário), adicione um comentário para descrever a chave pública.

Quando terminar, escolha Add (Adicionar).


5. Registre o ID da chave pública. Ele será usado posteriormente quando você criar signed URLs ou cookies como o valor do campo `Key-Pair-Id`.

Como adicionar a chave pública a um grupo de chaves

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No menu de navegação, escolha Key groups (Grupos de chaves).
3. Escolha Add key group (Adicionar grupo de chaves).
4. Na página Create key group (Criar grupo de chaves) faça o seguinte:

- a. Em Key group name (Nome do grupo de chaves), digite um nome para identificar o grupo de chaves.
 - b. (Opcional) Em Comment (Comentário), digite um comentário para descrever o grupo de chaves.
 - c. Em Public keys (Chaves públicas), selecione a chave pública a ser adicionada ao grupo de chaves e escolha Add (Adicionar). Repita essa etapa para cada chave pública que você deseja adicionar ao grupo de chaves.
5. Escolha Create key group (Criar grupo de chaves).
 6. Registre o nome do grupo de chaves. Ele será usado posteriormente para associar o grupo de chaves a um comportamento de cache em uma distribuição do CloudFront. (Na API do CloudFront, use o ID do grupo de chaves para associar o grupo de chaves a um comportamento de cache.)

Criar um par de chaves do CloudFront (não recomendado, requer o usuário-raiz da Conta da AWS)

 Important


Recomendamos que você crie uma chave pública para um grupo de chaves confiável em vez de seguir estas etapas. Para a maneira recomendada de criar chaves públicas para signed URLs e signed cookies, consulte [Criar um par de chaves para um grupo de chaves confiável \(recomendado\)](#).

É possível criar um par de chaves do CloudFront das seguintes maneiras:

- Crie um par de chaves no AWS Management Console e faça download da chave privada. Consulte o procedimento a seguir.
- Crie um par de chaves RSA usando uma aplicação, como o OpenSSL, e faça upload da chave pública no AWS Management Console. Para mais informações sobre como criar um par de chaves RSA, consulte [Criar um par de chaves para um grupo de chaves confiável \(recomendado\)](#).


Para criar pares de chaves do CloudFront no AWS Management Console

1. Faça login no AWS Management Console usando as credenciais do usuário-raiz da conta da AWS.

 Important


Usuários do IAM não podem criar pares de chaves do CloudFront. É necessário fazer login usando as credenciais de usuário-raiz para criar pares de chaves.

2. Escolha o nome da sua conta e selecione My Security Credentials (Minhas credenciais de segurança).
3. Escolha CloudFront key pairs (Pares de chaves do CloudFront).
4. Confirme se você não tem mais de um par de chaves ativo. Não será possível criar um par de chaves se você já tiver dois pares de chaves ativos.
5. Selecione Create a new key pair (Criar um par de chaves).

 Note

Também é possível criar um par de chaves próprio e fazer upload da chave pública. Os pares de chaves do CloudFront são compatíveis com chaves de 1024, 2048 ou 4096 bits.

6. Na caixa de diálogo Create Key Pair (Criar par de chaves), escolha Download Private Key File (Fazer download do arquivo de chave privada) e salve o arquivo no computador.

 Important

Salve a chave privada do par de chaves do CloudFront em um local seguro e defina as permissões no arquivo para que somente os usuários administradores desejados possam lê-lo. Se alguém obtiver sua chave privada, poderá gerar signed URLs e cookies válidos e fazer download do seu conteúdo. Não é possível gerar a chave privada novamente. Portanto, se perder ou excluí-la, crie outro par de chaves do CloudFront.

7. Anote o ID do seu par de chaves. (No AWS Management Console, ele é chamado de ID de chave de acesso.) Ele será usado quando você criar signed URLs ou cookies.

Reformatar a chave privada (somente .NET e Java)

Se estiver usando .NET ou Java para criar signed URLs ou cookies, você não poderá usar a chave privada do par de chaves no formato padrão PEM para criar a assinatura. Em vez disso, faça o seguinte:

- Framework .NET: converta a chave privada no formato XML usado pelo framework .NET. Várias ferramentas estão disponíveis.
- Java: converta a chave privada no formato DER. Uma maneira de fazer isso é com o comando OpenSSL a seguir. No comando a seguir, `private_key.pem` é o nome do arquivo que contém a chave privada formatada em PEM e `private_key.der` é o nome do arquivo que contém a chave privada formatada em DER depois que o comando é executado.

```
openssl pkcs8 -topk8 -nocrypt -in private_key.pem -inform PEM -out private_key.der -  
outform DER
```

Para garantir que o codificador funcione corretamente, adicione o JAR para as APIs de criptografia Java Bouncy Castle do seu projeto e adicione o provedor do Bouncy Castle.

Adicionar um assinante a uma distribuição

Um assinante é o grupo de chaves confiável (recomendado) ou o par de chaves do CloudFront que pode criar signed URLs e signed cookies para uma distribuição. Para usar signed URLs ou signed cookies com uma distribuição do CloudFront, é necessário especificar um assinante.

Os assinantes estão associados a comportamentos de cache. Isso permite exigir signed URLs ou cookies para alguns arquivos e não para outros na mesma distribuição. Uma distribuição requer signed URLs ou cookies somente para arquivos associados aos comportamentos de cache correspondentes.

Da mesma forma, um assinante só pode assinar URLs ou cookies para arquivos associados aos comportamentos de cache correspondentes. Por exemplo, se você tiver um assinante para um comportamento de cache e um assinante diferente para outro comportamento de cache, nenhum dos dois assinantes poderão criar signed URLs ou cookies para arquivos associados ao outro comportamento de cache.

Important

Antes de adicionar um assinante à sua distribuição, faça o seguinte:

- Defina os padrões de caminho nos comportamentos de cache e a sequência de comportamentos de cache cuidadosamente para que você não dê aos usuários acesso indesejado ao seu conteúdo ou impeça que eles acessem o conteúdo que você deseja que esteja disponível para todos.

Por exemplo, imagine que uma solicitação corresponda ao padrão de caminho de dois comportamentos de cache. O primeiro comportamento de cache não requer signed URLs ou cookies, mas o segundo comportamento de cache, sim. Os usuários poderão acessar os arquivos sem usar signed URLs ou signed cookies porque o CloudFront processa o comportamento de cache associado à primeira correspondência.

Para obter mais informações sobre padrões de caminho, consulte [Padrão de caminho](#).

- Para uma distribuição que você já esteja usando para distribuir conteúdo, certifique-se de que esteja pronto para começar a gerar signed URLs e cookies antes de adicionar um assinante. Quando você adiciona um assinante, o CloudFront rejeita as solicitações que não incluam um signed URL ou signed cookie válido.

É possível adicionar assinantes à sua distribuição usando o console ou a API do CloudFront.

Console

As etapas a seguir mostram como adicionar um grupo de chaves confiável como assinante. Também é possível adicionar uma Conta da AWS como signatário confiável, mas isso não é recomendado.

Como adicionar um assinante a uma distribuição usando o console

1. Registre o ID do grupo de chaves que você deseja usar como signatário confiável. Para obter mais informações, consulte [Criar um par de chaves para um grupo de chaves confiável \(recomendado\)](#).
2. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
3. Escolha a distribuição com os arquivos que você deseja proteger com signed URLs ou cookies.

Note

Para adicionar um assinante a uma nova distribuição, especifique as mesmas configurações descritas na etapa 6 ao criar a distribuição.

4. Escolha a guia Behaviors.
5. Selecione o comportamento de cache com o padrão de caminho corresponde aos arquivos que você deseja proteger com signed URLs ou cookies e escolha Edit (Editar).
6. Na página Edit Behavior (Editar Comportamento) faça o seguinte:
 - a. Em Restrict Viewer Access (Use Signed URLs or Signed Cookies) (Restringir acesso do visualizador (Usar signed URLs ou signed Cookies)), escolha Yes (Sim).
 - b. Em Trusted Key Groups or Trusted Signer (Grupos de chaves confiáveis ou signatário confiável), escolha Trusted Key Groups (Grupos de chaves confiáveis).
 - c. Em Trusted Key Groups (Grupos de chaves confiáveis), escolha o grupo de chaves a ser adicionado e escolha Add (Adicionar). Repita se quiser adicionar mais de um grupo de chaves.
7. Escolha Yes, Edit (Sim, editar) para atualizar o comportamento do cache.

API

É possível usar a API do CloudFront para adicionar um grupo de chaves confiável como assinante. É possível adicionar um assinante a uma distribuição existente ou a uma nova distribuição. Em qualquer caso, especifique os valores no elemento `TrustedKeyGroups`.

Também é possível adicionar uma Conta da AWS como signatário confiável, mas isso não é recomendado.

Consulte os seguintes tópicos na Referência da API do Amazon CloudFront:

- Atualizar uma distribuição existente: [UpdateDistribution](#)
- Criar uma nova distribuição: [CreateDistribution](#)

Alternar pares de chaves

Recomendamos que você alterne (mude) periodicamente seus pares de chaves para signed URLs e cookies. Para alternar os pares de chaves que você estiver usando para criar signed URLs ou cookies sem invalidar os URLs ou os cookies que ainda não expiraram, execute as seguintes tarefas:

1. Crie um par de chaves e adicione a chave pública a um grupo de chaves. Para obter mais informações, consulte [Criar um par de chaves para um grupo de chaves confiável \(recomendado\)](#).
2. Se você criou um grupo de chaves na etapa anterior, [adicione-o à distribuição como assinante](#).

Important

Não remova nenhuma chave pública existente do grupo de chaves ou nenhum grupo de chaves da distribuição ainda. Somente adicione os novos.

3. Atualize sua aplicação para criar assinaturas usando a chave privada do novo par de chaves. Confirme se os signed URLs ou cookies com as novas chaves privadas estão funcionando.
4. Espere passar a data de expiração dos URLs ou cookies que foram assinados usando a chave privada antiga. Depois, remova a chave pública antiga do grupo de chaves. Se você criou um grupo de chaves na etapa 2, remova o grupo de chaves antigo da sua distribuição.

Decidir usar URLs ou cookies assinados

Os signed URLs e signed cookies do CloudFront fornecem a mesma funcionalidade básica: eles permitem controlar quem pode acessar seu conteúdo. Se você quiser fornecer conteúdo privado pelo CloudFront e estiver tentando decidir se usará signed URLs ou signed cookies, considere as informações a seguir.

Use signed URLs nos seguintes casos:

- Você quer restringir o acesso a arquivos individuais, por exemplo, o download de uma instalação para seu aplicativo.
- Seus usuários estão usando um cliente (por exemplo, um cliente HTTP personalizado) incompatível com cookies.

Use signed cookies nos seguintes casos:

- Você quer fornecer acesso a vários arquivos restritos, por exemplo, todos os arquivos de um vídeo no formato HLS ou todos os arquivos da área de assinantes de um site.
- Você não quer alterar seus URLs atuais.

Se você não estiver usando URLs assinados e seus URLs (não assinados) contiverem um dos seguintes parâmetros de string de consulta, você não poderá usar signed URLs ou signed cookies:

- Expires
- Policy
- Signature
- Key-Pair-Id

O CloudFront pressupõe que os URLs que contenham um desses parâmetros de query string sejam signed URLs, portanto, não analisará os signed cookies.

Usar URLs e cookies assinados

Signed URLs têm precedência sobre signed cookies. Se você usar signed URLs e signed cookies para controlar o acesso aos mesmos arquivos e um visualizador usar um signed URL para solicitar um arquivo, o CloudFront determinará se retornará o arquivo para o visualizador com base somente no signed URL.

Usar URLs assinados

Um signed URL inclui informações adicionais, por exemplo, uma data e hora de expiração, que proporcionam a você mais controle sobre o acesso a seu conteúdo. Essas informações adicionais são descritas em uma declaração de política, que é baseada em uma política padrão ou personalizada. As diferenças entre a política padrão e a personalizada estão explicadas nas duas próximas seções.

Note

Você pode criar alguns signed URLs usando políticas padrão e outros usando políticas personalizadas para a mesma distribuição.

Tópicos

- [Decidir usar políticas predefinidas ou personalizadas para URLs assinados](#)
- [Como signed URLs funcionam](#)
- [Decidir o tempo de validade de URLs assinados](#)
- [Quando o CloudFront confere a data e hora de validade de um URL assinado?](#)
- [Código de exemplo e ferramentas de terceiros](#)
- [Criar um URL assinado usando uma política predefinida](#)
- [Criar um URL assinado usando uma política personalizada](#)

Decidir usar políticas predefinidas ou personalizadas para URLs assinados

Ao criar um signed URL, você grava uma declaração de política no formato JSON que especifica as restrições no signed URL, por exemplo, por quanto tempo o URL é válido. Você pode usar uma política padrão ou personalizada. Veja uma comparação entre as duas:

Descrição	Política padrão	Política personalizada
Você pode reutilizar a declaração de política para vários arquivos. Para reutilizar a declaração de política, é necessário usar caracteres curinga no objeto Resource. Para obter mais informações, consulte Valores especificados na declaração de política para um signed URL que usa uma política personalizada.)	Não	Sim
Você pode especificar a data e a hora em que os usuários podem começar a acessar seu conteúdo.	Não	Sim (opcional)
Você pode especificar a data e a hora em que os usuários não podem mais acessar seu conteúdo.	Sim	Sim
Você pode especificar o endereço IP ou vários endereços IP dos usuários que podem acessar seu conteúdo.	Não	Sim (opcional)

Descrição	Política padrão	Política personalizada
O signed URL inclui uma versão da política codificada em base64, resultando em um URL mais longo.	Não	Sim

Para obter informações sobre como criar signed URLs usando uma política padrão, consulte [Criar um URL assinado usando uma política predefinida](#).

Para obter informações sobre como criar signed URLs usando uma política personalizada, consulte [Criar um URL assinado usando uma política personalizada](#).

Como signed URLs funcionam

A seguir, uma visão geral de como configurar o CloudFront e o Amazon S3 para signed URLs e como o CloudFront responde quando um usuário usa um signed URL para solicitar um arquivo.

1. Na sua distribuição do CloudFront, especifique um ou mais grupos de chaves confiáveis, que contenham as chaves públicas que o CloudFront pode usar para verificar a assinatura do URL. Use as chaves privadas correspondentes para assinar os URLs.

Para obter mais informações, consulte [Especificar os assinantes que podem criar URLs e cookies assinados](#).

2. Desenvolva sua aplicação para determinar se um usuário deve ter acesso a seu conteúdo e criar signed URLs para os arquivos ou partes da aplicação às quais você deseja restringir o acesso. Para obter mais informações, consulte os tópicos a seguir:
 - [Criar um URL assinado usando uma política predefinida](#)
 - [Criar um URL assinado usando uma política personalizada](#)
3. Um usuário solicita um arquivo para o qual você deseja exigir signed URLs.
4. Seu aplicativo verifica se o usuário está autorizado a acessar o arquivo: ele fez login, pagou para acessar o conteúdo ou atendeu a outro requisito de acesso.
5. O aplicativo cria e retorna um signed URL para o usuário.
6. O signed URL permite que o usuário faça download ou transmita o conteúdo.

Essa etapa é automática; o usuário geralmente não precisa fazer nada a mais para acessar o conteúdo. Por exemplo, se um usuário estiver acessando seu conteúdo em um navegador

da Web, a aplicação retornará o signed URL para o navegador. O navegador imediatamente usa o signed URL para acessar o arquivo no ponto de presença de caches do CloudFront sem intervenção do usuário.

7. O CloudFront usa a chave pública para validar a assinatura e confirmar se o URL não foi adulterado. Se a assinatura for inválida, a solicitação será rejeitada.

Se a assinatura for válida, o CloudFront analisará a declaração de política no URL (ou criará uma se você estiver usando uma política padrão) para confirmar se a solicitação continua válida. Por exemplo, se você especificou uma data e hora de início e término para o URL, o CloudFront confirmará se o usuário está tentando acessar o conteúdo durante o período de acesso permitido.

Se a solicitação cumprir os requisitos da declaração de política, o CloudFront executará as operações padrão: determinar se o arquivo já está no ponto de presença de caches, encaminhar a solicitação para a origem, se necessário, e retornar o arquivo para o usuário.

Note

Se um URL não assinado contiver parâmetros de string de consulta, certifique-se de incluí-los na parte do URL que você assinar. Se você adicionar uma string de consulta a um signed URL depois de assiná-lo, o URL retornará um status HTTP 403.

Decidir o tempo de validade de URLs assinados

Você pode distribuir conteúdo privado usando um signed URL válido apenas por um período curto (possivelmente por apenas alguns minutos). URLs assinados válidos por um curto período são bons para distribuir conteúdo imediato a um usuário para uma finalidade específica, como a distribuição de aluguéis de filmes ou downloads de música para os clientes sob demanda. Caso seu signed URLs sejam válidos apenas por um curto período, gere-os automaticamente usando um aplicativo desenvolvido por você. Quando o usuário começar a fazer download de um arquivo ou a reproduzir um arquivo de mídia, o CloudFront comparará a hora de expiração do URL com a hora atual para determinar se o URL continua válido.

Você também pode distribuir conteúdo privado usando um signed URL válido por um período mais longo (possivelmente por anos). Signed URLs válidos por um período mais longo são úteis para distribuir conteúdo privado para usuários conhecidos, como a distribuição de um plano de negócios

para investidores ou de materiais de treinamento para funcionários. É possível desenvolver uma aplicação para gerar esses URLs assinados de longo prazo para você.

Quando o CloudFront confere a data e hora de validade de um URL assinado?

O CloudFront verifica a data e hora de expiração de um signed URL no momento da solicitação HTTP. Se um cliente começar a fazer download de um grande arquivo logo antes da hora de expiração, o download será concluído mesmo se passar a hora de expiração durante o download. Se a conexão TCP cair e o cliente tentar reiniciar o download após a hora de expiração, ocorrerá falha no download.

Se o cliente usar Range GETs para obter um arquivo em partes menores, ocorrerá falha em qualquer solicitação GET que ocorrer após a hora de expiração. Para obter mais informações sobre Range GETs, consulte [Como o CloudFront processa solicitações parciais de um objeto \(Range GETs\)](#).

Código de exemplo e ferramentas de terceiros

Para obter um código de exemplo que cria a parte assinada e com hash dos signed URLs, consulte os seguintes tópicos:

- [Criar uma assinatura de URL usando Perl](#)
- [Criar uma assinatura de URL usando PHP](#)
- [Criar uma assinatura de URL usando C# e o .NET Framework](#)
- [Criar uma assinatura de URL usando Java](#)

Criar um URL assinado usando uma política predefinida

Para criar um signed URL usando uma política enlatada, conclua as etapas a seguir.

Para criar um signed URL usando uma política padrão

1. Se você estiver usando o .NET ou Java para criar signed URLs e não tiver reformatado a chave privada do seu par de chaves para do formato padrão .pem para um formato compatível com o .NET ou Java, faça isso agora. Para ter mais informações, consulte [Reformatar a chave privada \(somente .NET e Java\)](#).
2. Concatene os seguintes valores na ordem listada, replicando o formato mostrado neste exemplo de URL assinado:

```
https://d111111abcdef8.cloudfront.net/  
image.jpg?color=red&size=medium&Expires=1357034400&Signature=nitfHRCrtziw02HwPFWw~yYDhUF5Ew  
j19DzZrvDh6hQ73lDx~-ar3UocvvRQVw6EkC~GdpGQyy0SKQim-  
TxAnW7d8F5Kkai9HVx0FIu-5jcQb0UEmatEXAMPLE3ReXySpLSMj0yCd3ZAB4UcBCAqEijkytL6f3fVYNGQI6&Key-  
Pair-Id=K2JCJMDEHXQW5F
```

Remova todos os espaços vazios (inclusive caracteres de nova linha e de tabulação). Pode ser necessário incluir caracteres de escape na string do código do aplicativo. Todos os valores têm um tipo de String.

1. URL base do arquivo

O URL base é o URL do CloudFront usado para acessar o arquivo se você não estivesse usando signed URLs, inclusive seus próprios parâmetros de query string, se houver. No exemplo anterior, o URL de base é `https://d111111abcdef8.cloudfront.net/image.jpg`. Para mais informações sobre o formato dos URLs para distribuições, consulte .

- O URL do CloudFront a seguir é para um arquivo de imagem em uma distribuição (usando o nome de domínio do CloudFront). Observe que `image.jpg` está em um diretório `images`. O caminho para o arquivo no URL deve corresponder ao caminho para o arquivo no servidor HTTP ou no bucket do Amazon S3.

```
https://d111111abcdef8.cloudfront.net/images/image.jpg
```

- O seguinte URL do CloudFront inclui uma query string:

```
https://d111111abcdef8.cloudfront.net/images/image.jpg?size=large
```

- Os URLs do CloudFront a seguir são para arquivos de imagem em uma distribuição. Ambos usam um nome de domínio alternativo. O segundo inclui uma string de consulta:

```
https://www.example.com/images/image.jpg
```

```
https://www.example.com/images/image.jpg?color=red
```

- O URL do CloudFront a seguir é para um arquivo de imagem em uma distribuição que usa um nome de domínio alternativo e o protocolo HTTPS:

```
https://www.example.com/images/image.jpg
```

2. ?

O ? indica que os parâmetros de string de consulta seguem o URL base. Inclua ? mesmo se você não tiver parâmetros de string de consulta.

3. ***Seus parâmetros de string de consulta, se houver&***

Este valor é opcional. Se você quiser adicionar seus próprios parâmetros de query string, por exemplo:

```
color=red&size=medium
```

e depois adicionar os parâmetros após ? e antes do parâmetro Expires. Em algumas circunstâncias raras, pode ser necessário inserir seus parâmetros de query string depois de Key-Pair-Id.

Important

Seus parâmetros não podem ser denominados Expires, Signature nem Key-Pair-Id.

Se você adicionar seus próprios parâmetros, inclua & depois de cada um deles, inclusive o último.

4. ***Expires=data e hora no formato de hora do Unix (em segundos) e no Tempo Universal Coordenado (UTC)***

A data e a hora em que você deseja que o URL pare de permitir acesso ao arquivo.

Especifique a data e hora de expiração no formato de hora do Unix (em segundos) e no Tempo Universal Coordenado (UTC). Por exemplo, 1º de janeiro de 2013 10h UTC é convertido em 1357034400 no formato de hora do Unix, conforme mostrado no exemplo no início deste tópico. Para usar o horário epoch, use um número inteiro de 32 bits para uma data que pode ser até 2147483647 (19 de janeiro de 2038 às 03:14:07 UTC). Para ter informações sobre UTC, consulte [RFC 3339, Date and Time on the Internet: Timestamps](#).

5. **&Signature=versão assinada e com hash da declaração de política**

Uma versão assinada, com hash e codificação base64 da declaração de política do JSON. Para ter mais informações, consulte [Criar uma assinatura para um URL assinado que use uma política predefinida](#).

6. **&Key-Pair-Id=ID da chave pública do CloudFront cuja chave privada correspondente está sendo usada para gerar a assinatura**

O ID de uma chave pública do CloudFront, por exemplo, K2JCMDEHXQW5F. O ID da chave pública informa ao CloudFront qual chave pública deve ser usada para validar o signed URL. O CloudFront compara as informações da assinatura com as informações da declaração de política para verificar se o URL não foi adulterado.

Essa chave pública deve pertencer a um grupo de chaves que seja um signatário confiável na distribuição. Para ter mais informações, consulte [Especificar os assinantes que podem criar URLs e cookies assinados](#).

Criar uma assinatura para um URL assinado que use uma política predefinida

Para criar a assinatura de um URL que use uma política predefinida, siga estes procedimentos.

Tópicos

- [Criar uma declaração de política para um URL assinado que use uma política predefinida](#)
- [Criar uma assinatura para um URL assinado que use uma política predefinida](#)

Criar uma declaração de política para um URL assinado que use uma política predefinida

Ao criar um signed URL usando uma política padrão, o parâmetro `Signature` será uma versão assinada e com hash de uma declaração de política. Para signed URLs que usam uma política padrão, a declaração de política não é incluída no URL, como é feito nos signed URLs que usam uma política personalizada. Para criar a declaração de política, siga o procedimento abaixo.

Para criar a declaração de política para um signed URL que usa uma política padrão

1. Crie a declaração de política usando o formato JSON a seguir e a codificação de caracteres UTF-8. Inclua todas as pontuações e outros valores literais exatamente como especificado. Para obter informações sobre os parâmetros `Resource` e `DateLessThan`, consulte [Valores especificados na declaração de política para um signed URL que usa uma política padrão](#).

```
{
  "Statement": [
    {
      "Resource": "base URL or stream name",
      "Condition": {
        "DateLessThan": {
          "AWS:EpochTime": ending date and time in Unix time format and
          UTC
        }
      }
    }
  ]
}
```

2. Remova todas os espaços em branco (inclusive caracteres de nova linha e de tabulação) da declaração de política. Pode ser necessário incluir caracteres de escape na string do código do aplicativo.

Valores especificados na declaração de política para um signed URL que usa uma política padrão

Ao criar uma declaração de política para uma política padrão, especifique os valores a seguir.

Recurso

Note

Você pode especificar apenas um valor para Resource.

O URL base, com suas query strings, se houver, sem os parâmetros Expires, Signature e Key-Pair-Id do CloudFront, por exemplo:

```
https://d1111111abcdef8.cloudfront.net/images/horizon.jpg?
size=large&license=yes
```

Observe o seguinte:

- Protocolo: o valor deve começar com `http://` ou `https://`.
- Parâmetros de query string :se você não tiver query strings, omita o ponto de interrogação.

- Nomes de domínio alternativos: se especificar um nome de domínio alternativo (CNAME) no URL, você deverá especificá-lo ao fazer referência ao arquivo na sua página da web ou aplicação. Não especifique o URL do Amazon S3 do objeto.

DateLessThan

A data e hora de expiração do URL no formato de hora do Unix (em segundos) e no Tempo Universal Coordenado (UTC). Por exemplo, 1º de janeiro de 2013 10h UTC é convertido para 1357034400 no formato de hora do Unix.

Esse valor deve corresponder ao valor do parâmetro de query string Expires do signed URL. Não coloque os valores entre aspas.

Para obter mais informações, consulte [Quando o CloudFront confere a data e hora de validade de um URL assinado?](#).

Exemplo de declaração de política para um signed URL que usa uma política padrão

Ao usar o seguinte exemplo de declaração de política em um signed URL, um usuário pode acessar o arquivo `https://d111111abcdef8.cloudfront.net/horizon.jpg` até 1º de janeiro de 2013, 10h UTC:

```
{
  "Statement": [
    {
      "Resource": "https://d111111abcdef8.cloudfront.net/horizon.jpg?size=large&license=yes",
      "Condition": {
        "DateLessThan": {
          "AWS:EpochTime": 1357034400
        }
      }
    }
  ]
}
```

Criar uma assinatura para um URL assinado que use uma política predefinida

Para criar o valor para o parâmetro Signature em um signed URL, assine e adicione um hash à declaração de política criada em [Criar uma declaração de política para um URL assinado que use uma política predefinida](#).


Para obter mais informações e exemplos de como adicionar hash, assinar e codificar a declaração de política, consulte:

- [Comandos do Linux e OpenSSL para criptografia e codificação base64](#)
- [Exemplos de código para criar uma assinatura para um signed URL](#)

Opção 1: Como criar uma assinatura usando uma política enlatada

1. Use a função de hash SHA-1 e o RSA para assinar e adicionar um hash à declaração de política criada no procedimento [Para criar a declaração de política para um signed URL que usa uma política padrão](#). Use a versão da declaração de política que não inclui mais espaços em branco.

Para a chave privada exigida pela função hash, use uma chave privada que tenha a chave pública em um grupo de chaves confiável ativo para a distribuição.

 Note

O método usado para assinar e adicionar um hash à declaração de política depende da sua linguagem de programação e plataforma. Para obter o código de exemplo, consulte [Exemplos de código para criar uma assinatura para um signed URL](#).

2. Remova os espaços em branco (inclusive caracteres de nova linha e de tabulação) da string assinada e com hash.
3. Codifique a string usando codificação base64 MIME. Para obter mais informações, consulte [Section 6.8, Base64 Content-Transfer-Encoding](#) em RFC 2045, MIME (Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies.
4. Substitua os caracteres inválidos da query string de um URL por caracteres válidos. A tabela a seguir indica os caracteres válidos e inválidos.

Substitua esses caracteres inválidos	Por esses caracteres válidos
+	- (hífen)
=	_ (sublinhado)
/	~ (til)

5. Inclua o valor resultante no seu signed URL depois de `&Signature=` e volte para [Para criar um signed URL usando uma política padrão](#) para concluir a concatenação das partes dele.

Criar um URL assinado usando uma política personalizada

Para criar um URL assinado usando uma política personalizada, execute o procedimento a seguir.

Para criar um signed URL usando uma política personalizada

1. Se você estiver usando o .NET ou Java para criar signed URLs e não tiver reformatado a chave privada do seu par de chaves para do formato padrão .pem para um formato compatível com o .NET ou Java, faça isso agora. Para ter mais informações, consulte [Reformatar a chave privada \(somente .NET e Java\)](#).
2. Concatene os seguintes valores na ordem listada, replicando o formato mostrado neste exemplo de URL assinado:

```
https://d111111abcdef8.cloudfront.net/  
image.jpg?color=red&size=medium&Policy=eyJANCIgICEXAMPLEW1lbnQiOiBbeyANCiAgICAgICJSZXNvdXJj  
j19DzZrvDh6hQ73LDx~-ar3UocvvRQVw6EkC~GdpGQyy0SKQim-  
TxAnW7d8F5Kkai9HVx0FIu-5jcQb0UEmatEXAMPLE3ReXySpLSMj0yCd3ZAB4UcBCAqEijkytL6f3fVYNGQI6&Key-  
Pair-Id=K2JCMDEHXQW5F
```

Remova todos os espaços vazios (inclusive caracteres de nova linha e de tabulação). Pode ser necessário incluir caracteres de escape na string do código do aplicativo. Todos os valores têm um tipo de `String`.

1. **URL base do arquivo**

O URL base é o URL do CloudFront usado para acessar o arquivo se você não estivesse usando signed URLs, inclusive seus próprios parâmetros de query string, se houver. No exemplo anterior, o URL de base é `https://d111111abcdef8.cloudfront.net/image.jpg`. Para mais informações sobre o formato dos URLs para distribuições, consulte .

Os exemplos a seguir mostram os valores que você especifica para suas distribuições.

- O URL do CloudFront a seguir é para um arquivo de imagem em uma distribuição (usando o nome de domínio do CloudFront). Observe que `image.jpg` está em um diretório `images`. O caminho para o arquivo no URL deve corresponder ao caminho para o arquivo no servidor HTTP ou no bucket do Amazon S3.

```
https://d111111abcdef8.cloudfront.net/images/image.jpg
```

- O seguinte URL do CloudFront inclui uma query string:

```
https://d111111abcdef8.cloudfront.net/images/image.jpg?size=large
```

- Os URLs do CloudFront a seguir são para arquivos de imagem em uma distribuição. Os dois usam um nome de domínio alternativo; o segundo inclui uma query string:

```
https://www.example.com/images/image.jpg
```

```
https://www.example.com/images/image.jpg?color=red
```

- O URL do CloudFront a seguir é para um arquivo de imagem em uma distribuição que usa um nome de domínio alternativo e o protocolo HTTPS:

```
https://www.example.com/images/image.jpg
```

2. ?

O ? indica que os parâmetros de string de consulta seguem o URL base. Inclua ? mesmo se você não tiver parâmetros de string de consulta.

3. ***Seus parâmetros de string de consulta, se houver&***

Este valor é opcional. Se você quiser adicionar seus próprios parâmetros de query string, por exemplo:

```
color=red&size=medium
```

e, depois, adicioná-los após ? e antes do parâmetro Policy. Em algumas circunstâncias raras, pode ser necessário inserir seus parâmetros de query string depois de Key-Pair-Id.

Important

Seus parâmetros não podem ser denominados Policy, Signature nem Key-Pair-Id.

Se você adicionar seus próprios parâmetros, inclua & depois de cada um deles, inclusive o último.

4. **Policy=versão da declaração de política codificada em base64**

Sua declaração de política no formato JSON, sem espaços em branco e com codificação base64. Para ter mais informações, consulte [Criar uma declaração de política para um URL assinado que use uma política personalizada](#).

A declaração de política controla o acesso que um signed URL concede a um usuário. Ela inclui o URL do arquivo, uma data e hora de expiração, uma data e hora opcionais em que o URL se torna válido e um endereço IP opcional ou intervalo de endereços IP que tenha permissão para acessar o arquivo.

5. **&Signature=versão assinada e com hash da declaração de política**

Uma versão assinada, com hash e codificação base64 da declaração de política do JSON. Para ter mais informações, consulte [Criar uma assinatura para um URL que use uma política personalizada](#).

6. **&Key-Pair-Id=ID da chave pública do CloudFront cuja chave privada correspondente está sendo usada para gerar a assinatura**

O ID de uma chave pública do CloudFront, por exemplo, K2JJCJMDEHXQW5F. O ID da chave pública informa ao CloudFront qual chave pública deve ser usada para validar o signed URL. O CloudFront compara as informações da assinatura com as informações da declaração de política para verificar se o URL não foi adulterado.

Essa chave pública deve pertencer a um grupo de chaves que seja um signatário confiável na distribuição. Para ter mais informações, consulte [Especificar os assinantes que podem criar URLs e cookies assinados](#).

Criar uma declaração de política para um URL assinado que use uma política personalizada

Conclua as etapas a seguir para criar uma declaração de política para um signed URL que usa uma política personalizada.

Para obter exemplos de declarações de política que controlam o acesso a arquivos de diversas formas, consulte [the section called “Exemplos de declaração de política para um signed URL que usa uma política personalizada”](#).

Para criar a declaração de política para um signed URL que usa uma política personalizada

1. Crie a declaração de política usando o formato JSON a seguir. Substitua os símbolos menor que (<) e maior que (>) e as descrições contidas neles por seus próprios valores. Para ter mais informações, consulte [the section called “Valores especificados na declaração de política para um signed URL que usa uma política personalizada”](#).

```
{
  "Statement": [
    {
      "Resource": "<Optional but recommended: URL of the file>",
      "Condition": {
        "DateLessThan": {
          "AWS:EpochTime": <Required: ending date and time in Unix time
format and UTC>
        },
        "DateGreaterThan": {
          "AWS:EpochTime": <Optional: beginning date and time in Unix time
format and UTC>
        },
        "IpAddress": {
          "AWS:SourceIp": "<Optional: IP address>"
        }
      }
    }
  ]
}
```

Observe o seguinte:

- É possível incluir somente uma declaração na política.
- Use a codificação de caracteres UTF-8.
- Inclua todas as pontuações e nomes de parâmetro exatamente como especificado. Abreviações de nomes de parâmetro não são aceitas.
- A ordem dos parâmetros na seção Condition não é importante.
- Para obter informações sobre os valores de Resource, DateLessThan, DateGreaterThan e IpAddress, consulte [the section called “Valores especificados na declaração de política para um signed URL que usa uma política personalizada”](#).

2. Remova todas os espaços em branco (inclusive caracteres de nova linha e de tabulação) da declaração de política. Pode ser necessário incluir caracteres de escape na string do código do aplicativo.
3. Codifique a declaração de política usando codificação base64 MIME. Para obter mais informações, consulte [Section 6.8, Base64 Content-Transfer-Encoding](#) em RFC 2045, MIME (Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies.
4. Substitua os caracteres inválidos da query string de um URL por caracteres válidos. A tabela a seguir indica os caracteres válidos e inválidos.

Substitua esses caracteres inválidos	Por esses caracteres válidos
+	- (hífen)
=	_ (sublinhado)
/	~ (til)

5. Inclua o valor resultante ao seu signed URL depois de Policy=.
6. Crie uma assinatura para o signed URL adicionando hash, assinando e codificando em Base64 a declaração de política. Para ter mais informações, consulte [the section called “Criar uma assinatura para um URL que use uma política personalizada”](#).

Valores especificados na declaração de política para um signed URL que usa uma política personalizada

Ao criar uma declaração de política para uma política personalizada, especifique os valores a seguir.

Recurso

O URL, incluindo quaisquer strings de consulta, mas excluindo os parâmetros Policy, Signature e Key-Pair-Id do CloudFront. Por exemplo:

```
https://d111111abcdef8.cloudfront.net/images/horizon.jpg?
size=large&license=yes
```

Você pode especificar somente um valor para Resource.

⚠ Important

Você pode omitir o parâmetro `Resource` de uma política, mas isso significa que qualquer pessoa com o signed URL poderá acessar todos os arquivos de qualquer distribuição associada ao par de chaves usado para criar o signed URL.

Observe o seguinte:

- Protocolo: o valor deve começar com `http://`, `https://` ou `*://`.
- Parâmetros da string de consulta: se o URL tiver parâmetros da string de consulta, use um caractere de barra invertida (`\`) para escapar do caractere de ponto de interrogação (`?`) que inicia a string de consulta. Por exemplo:

```
https://d1111111abcdef8.cloudfront.net/images/horizon.jpg\?  
size=large&license=yes
```

- Caracteres curinga: é possível usar caracteres curinga no URL da política. Os seguintes caracteres curinga são compatíveis:
 - asterisco (`*`), que corresponde a zero ou mais caracteres
 - ponto de interrogação (`?`) corresponde exatamente a um caractere

Quando o CloudFront combina o URL na política com o URL na solicitação HTTP, o URL na política é dividido em quatro seções (protocolo, domínio, caminho e string de consulta) da seguinte forma:

```
[protocol]://[domain]/[path]\?[query string]
```

Quando você usa um caractere curinga no URL da política, a correspondência de curingas se aplica somente dentro dos limites da seção que contém o curinga. Por exemplo, considere este signed URL em uma política:

```
https://www.example.com/hello*world
```

Neste exemplo, o caractere curinga asterisco (`*`) só se aplica à seção do caminho, portanto, ele corresponde aos URLs `https://www.example.com/helloworld` e `https://www.example.com/hello-world`, mas não corresponde ao URL `https://www.example.net/hello?world`.

As seguintes exceções se aplicam aos limites da seção para a correspondência de curingas:

- Um asterisco final na seção do caminho implica um asterisco na seção da string de consulta. Por exemplo, `http://example.com/hello*` equivale a `http://example.com/hello*\?*`.
- Um asterisco final na seção do domínio implica um asterisco nas seções do caminho e da string de consulta. Por exemplo, `http://example.com*` equivale a `http://example.com*/*\?*`.
- Um URL na política pode omitir a seção do protocolo e começar com um asterisco na seção do domínio. Nesse caso, a seção do protocolo é definida implicitamente como um asterisco. Por exemplo, o URL `*example.com` em uma política é equivalente a `*://*example.com/`.
- Um asterisco por si só (`"Resource": "*"`) corresponde a qualquer URL.

Por exemplo, o valor `https://d111111abcdef8.cloudfront.net/*game_download.zip*` em uma política corresponde a todos os seguintes URLs:

- `https://d111111abcdef8.cloudfront.net/game_download.zip`
- `https://d111111abcdef8.cloudfront.net/example_game_download.zip?license=yes`
- `https://d111111abcdef8.cloudfront.net/test_game_download.zip?license=temp`
- Nomes de domínio alternativos: se você especificar um nome de domínio alternativo (CNAME) no URL na política, A solicitação HTTP deverá usar o nome do domínio alternativo na sua página da web ou aplicação. Não especifique o URL do Amazon S3 para o arquivo em uma política.

DateLessThan

A data e hora de expiração do URL no formato de hora do Unix (em segundos) e no Tempo Universal Coordenado (UTC). Na política, não coloque os valores entre aspas. Para obter informações sobre UTC, consulte [Date and Time on the Internet: Timestamps](#).

Por exemplo, 31 de janeiro de 2023, 10h UTC é convertido em 1675159200 no formato de hora do Unix.

Esse é o único parâmetro obrigatório na seção `Condition`. O CloudFront requer esse valor para impedir que os usuários tenham acesso permanente ao seu conteúdo privado.

Para ter mais informações, consulte [the section called “Quando o CloudFront confere a data e hora de validade de um URL assinado?”](#).

DateGreaterThan (opcional)

Uma data e hora de início opcional do URL no formato de hora do Unix (em segundos) e no Tempo Universal Coordenado (UTC). Os usuários não podem acessar o arquivo antes ou na data e hora especificadas. Não coloque os valores entre aspas.

IpAddress (opcional)

O endereço IP do cliente que está fazendo a solicitação HTTP. Observe o seguinte:

- Para permitir o acesso de qualquer endereço IP ao arquivo, omita o parâmetro `IpAddress`.
- É possível especificar um ou vários endereços IP. Por exemplo, você não pode usar a política que permitir o acesso se o endereço IP do cliente estiver em um de dois intervalos separados.
- Para permitir o acesso de um único endereço IP, especifique:

"IPv4 IP address/32"

- Você deve especificar os intervalos de endereço IP no formato CIDR IPv4 padrão (por exemplo, `192.0.2.0/24`). Para obter mais informações, consulte [Encaminhamento Entre Domínios Sem Classificação \(CIDR\): A atribuição do endereço da Internet e o plano de agregação](#).

Important

Endereços IP no formato IPv6, como `2001:0db8:85a3::8a2e:0370:7334`, não são compatíveis.

Se você estiver usando uma política personalizada que inclui `IpAddress`, não permita o IPv6 para a distribuição. Se você quiser restringir o acesso a um conteúdo por endereço IP e oferecer suporte a solicitações IPv6 para outro tipo de conteúdo, crie duas distribuições. Para obter mais informações, consulte [the section called “Enable IPv6”](#) no tópico [the section called “Configurações de distribuição”](#).

Exemplos de declaração de política para um signed URL que usa uma política personalizada

Os exemplos de declaração de política a seguir mostram como controlar o acesso a um arquivo específico, todos os arquivos de um diretório ou todos os arquivos associados a um ID de par de chaves. Os exemplos também mostram como controlar o acesso de um único endereço IP ou um

intervalo de endereços IP e como evitar que os usuários usem o signed URL após a data e hora especificadas.

Se você copiar e colar qualquer um desses exemplos, remova os espaços em branco (inclusive caracteres de nova linha e de tabulação), substitua os valores pelos seus próprios valores e inclua um caractere de nova linha após a chave de fechamento (}).

Para ter mais informações, consulte [the section called “Valores especificados na declaração de política para um signed URL que usa uma política personalizada”](#).

Tópicos

- [Exemplo de declaração de política: acessar um arquivo de um intervalo de endereços IP](#)
- [Exemplo de declaração de política: acessar todos os arquivos de um diretório em um intervalo de endereços IP](#)
- [Exemplo de declaração de política: acessar todos os arquivos associados a um ID de par de chaves de um endereço IP](#)

Exemplo de declaração de política: acessar um arquivo de um intervalo de endereços IP

O exemplo a seguir de política personalizada em um signed URL especifica que um usuário pode acessar o arquivo `https://d111111abcdef8.cloudfront.net/game_download.zip` de endereços IP no intervalo `192.0.2.0/24` até 31 de janeiro de 2023, 10h UTC:

```
{
  "Statement": [
    {
      "Resource": "https://d111111abcdef8.cloudfront.net/game_download.zip",
      "Condition": {
        "IpAddress": {
          "AWS:SourceIp": "192.0.2.0/24"
        },
        "DateLessThan": {
          "AWS:EpochTime": 1675159200
        }
      }
    }
  ]
}
```

Exemplo de declaração de política: acessar todos os arquivos de um diretório em um intervalo de endereços IP

O exemplo a seguir de política personalizada permite criar signed URLs para qualquer arquivo no diretório `training`, conforme indicado pelo caractere curinga (*) no parâmetro `Resource`. Os usuários podem acessar o arquivo de um endereço IP no intervalo `192.0.2.0/24` até 31 de janeiro de 2023, 10h UTC:

```
{
  "Statement": [
    {
      "Resource": "https://d111111abcdef8.cloudfront.net/training/*",
      "Condition": {
        "IpAddress": {
          "AWS:SourceIp": "192.0.2.0/24"
        },
        "DateLessThan": {
          "AWS:EpochTime": 1675159200
        }
      }
    }
  ]
}
```

Cada signed URL com o qual você usa essa política tem um URL que identifica um arquivo específico, por exemplo:

`https://d111111abcdef8.cloudfront.net/training/orientation.pdf`

Exemplo de declaração de política: acessar todos os arquivos associados a um ID de par de chaves de um endereço IP

O exemplo de política personalizada a seguir permite que você crie signed URLs para qualquer arquivo associado a qualquer distribuição, conforme indicado pelo caractere curinga * no parâmetro `Resource`. O signed URL deve usar o protocolo `https://`, não o `http://`. O usuário deve usar o endereço IP `192.0.2.10/32`. (O valor `192.0.2.10/32` na notação CIDR se refere a um único endereço IP, `192.0.2.10`.) Os arquivos estão disponíveis somente de 31 de janeiro de 2023, 10h UTC, a 2 de fevereiro de 2023, 10h UTC:

```
{
  "Statement": [
```

```
{
  "Resource": "https://*",
  "Condition": {
    "IpAddress": {
      "AWS:SourceIp": "192.0.2.10/32"
    },
    "DateGreaterThan": {
      "AWS:EpochTime": 1675159200
    },
    "DateLessThan": {
      "AWS:EpochTime": 1675332000
    }
  }
}
```

Cada signed URL com o qual você usa essa política tem um URL que identifica um arquivo específico em uma distribuição específica do CloudFront, por exemplo:

```
https://d111111abcdef8.cloudfront.net/training/orientation.pdf
```

O signed URL também inclui um ID de par de chaves, que deve estar associado a um grupo de chaves confiável na distribuição (d111111abcdef8.cloudfront.net) especificada no URL.

Criar uma assinatura para um URL que use uma política personalizada

A assinatura de um signed URL que usa uma política personalizada é uma versão da declaração de política com hash, assinada e codificada em base64. Para criar uma assinatura para uma política personalizada, conclua as etapas a seguir.

Para obter mais informações e exemplos de como adicionar hash, assinar e codificar a declaração de política, consulte:

- [Comandos do Linux e OpenSSL para criptografia e codificação base64](#)
- [Exemplos de código para criar uma assinatura para um signed URL](#)

Opção 1: Como criar uma assinatura usando uma política personalizada

1. Use a função de hash SHA-1 e o RSA para assinar e adicionar um hash à declaração de política do JSON criada no procedimento [Para criar a declaração de política para um signed URL que](#)

[usa uma política personalizada](#). Use a versão da declaração de política que não inclui mais espaços em branco, mas que ainda não foi codificada em base64.

Para a chave privada exigida pela função hash, use uma chave privada que tenha a chave pública em um grupo de chaves confiável ativo para a distribuição.

Note

O método usado para assinar e adicionar um hash à declaração de política depende da sua linguagem de programação e plataforma. Para obter o código de exemplo, consulte [Exemplos de código para criar uma assinatura para um signed URL](#).

2. Remova os espaços em branco (inclusive caracteres de nova linha e de tabulação) da string assinada e com hash.
3. Codifique a string usando codificação base64 MIME. Para obter mais informações, consulte [Section 6.8, Base64 Content-Transfer-Encoding](#) em RFC 2045, MIME (Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies.
4. Substitua os caracteres inválidos da query string de um URL por caracteres válidos. A tabela a seguir indica os caracteres válidos e inválidos.

Substitua esses caracteres inválidos	Por esses caracteres válidos
+	- (hífen)
=	_ (sublinhado)
/	~ (til)

5. Inclua o valor resultante no seu signed URL depois de &Signature= e volte para [Para criar um signed URL usando uma política personalizada](#) para concluir a concatenação das partes dele.

Usar cookies assinados

Os signed cookies do CloudFront permitem controlar quem pode acessar seu conteúdo quando você não quiser alterar seus URLs atuais ou quando quiser fornecer acesso a vários arquivos restritos, por exemplo, todos os arquivos da área de assinantes de um site. Este tópico explica as considerações ao usar signed cookies e descreve como defini-los usando políticas padrão e personalizadas.

Tópicos

- [Decidir usar políticas predefinidas ou personalizadas para cookies assinados](#)
- [Como signed cookies funcionam](#)
- [Evitar o uso indevido de cookies assinados](#)
- [Quando o CloudFront confere a data e hora de validade de um cookie assinado](#)
- [Código de exemplo e ferramentas de terceiros](#)
- [Definir cookies assinados usando uma política predefinida](#)
- [Definir cookies assinados usando uma política personalizada](#)

Decidir usar políticas predefinidas ou personalizadas para cookies assinados

Ao criar um signed cookie, você grava uma declaração de política no formato JSON que especifica as restrições no signed cookie, por exemplo, por quanto tempo o cookie é válido. Você pode usar políticas padrão ou personalizadas. A tabela a seguir compara as políticas personalizadas e padrão:

Descrição	Política padrão	Política personalizada
Você pode reutilizar a declaração de política para vários arquivos. Para reutilizar a declaração de política, é necessário usar caracteres curinga no objeto Resource. Para obter mais informações, consulte Valores especificados na declaração de uma política personalizada para signed cookies.)	Não	Sim
Você pode especificar a data e a hora em que os usuários podem começar a acessar seu conteúdo	Não	Sim (opcional)
Você pode especificar a data e a hora em que os usuários não podem mais acessar seu conteúdo	Sim	Sim
Você pode especificar o endereço IP ou vários endereços IP dos usuários que podem acessar seu conteúdo	Não	Sim (opcional)

Para obter informações sobre como criar signed cookies usando uma política padrão, consulte [Definir cookies assinados usando uma política predefinida](#).

Para obter informações sobre como criar signed cookies usando uma política personalizada, consulte [Definir cookies assinados usando uma política personalizada](#).

Como signed cookies funcionam

A seguir, uma visão geral de como configurar o CloudFront para signed cookies e como o CloudFront responde quando um usuário envia uma solicitação que contenha um signed cookie.

1. Na sua distribuição do CloudFront, especifique um ou mais grupos de chaves confiáveis, que contenham as chaves públicas que o CloudFront pode usar para verificar a assinatura do URL. Use as chaves privadas correspondentes para assinar os URLs.

Para obter mais informações, consulte [Especificar os assinantes que podem criar URLs e cookies assinados](#).

2. Desenvolva seu aplicativo para determinar se um usuário deve ter acesso a seu conteúdo e, em caso afirmativo, para enviar três cabeçalhos Set-Cookie para o visualizador. (Cada cabeçalho Set-Cookie pode conter somente um par de nome/valor, e um signed cookie do CloudFront requer três pares de nome/valor.) Você deve enviar os cabeçalhos Set-Cookie para o visualizador antes de ele solicitar o conteúdo privado. Se você definir uma hora de expiração breve no cookie, envie mais três cabeçalhos Set-Cookie em resposta a solicitações subsequentes para que o usuário continue tendo acesso.

Normalmente, a distribuição do CloudFront terá pelo menos dois comportamentos de cache: um que não exige autenticação e um que exige. A página de erro da parte segura do site inclui um redirecionador ou link para uma página de login.

Se você configurar sua distribuição para armazenar arquivos em cache com base em cookies, o CloudFront não armazenará arquivos separados com base nos atributos nos signed cookies.

3. Um usuário faz login em seu site e paga pelo conteúdo ou cumpre outro requisito de acessar.
4. O aplicativo retorna os cabeçalhos Set-Cookie na resposta, e o visualizador armazena os pares de nome-valor.
5. O usuário solicita um arquivo.

O navegador do usuário ou outro visualizador obtém os pares de nome-valor da etapa 4 e os adiciona à solicitação em um cabeçalho Cookie. Esse é o signed cookie.

6. O CloudFront usa a chave pública para validar a assinatura no signed cookie e confirmar se o cookie não foi adulterado. Se a assinatura for inválida, a solicitação será rejeitada.

Se a assinatura do cookie for válida, o CloudFront analisará a declaração de política no cookie (ou criará uma se você estiver usando uma política padrão) para confirmar se a solicitação continua válida. Por exemplo, se você especificou uma data e hora de início e término para o cookie, o CloudFront confirmará se o usuário está tentando acessar o conteúdo durante o período de acesso permitido.

Se a solicitação cumprir os requisitos da declaração de política, o CloudFront fornecerá o conteúdo, como o faz com conteúdo não restrito: determina se o arquivo já está no ponto de presença de caches, encaminha a solicitação para a origem, se necessário, e retorna o arquivo para o usuário.

Evitar o uso indevido de cookies assinados

Se você especificar o parâmetro `Domain` em um cabeçalho `Set-Cookie`, especifique o valor mais preciso possível para reduzir o possível acesso por alguém com o mesmo nome de domínio raiz. Por exemplo, `app.example.com` é preferível a `example.com`, especialmente quando você não tem o controle sobre `example.com`. Isso ajuda a evitar que alguém acesse seu conteúdo de `www.example.com`.

Para ajudar a evitar esse tipo de ataque:

- Exclua os atributos de cookie `Expires` e `Max-Age` para que o cabeçalho `Set-Cookie` crie um cookie de sessão. Cookies de sessão são automaticamente excluídos quando o usuário fecha o navegador, diminuindo a possibilidade de alguém obter acesso não autorizado ao seu conteúdo.
- Inclua o atributo `Secure` para que o cookie seja criptografado quando o visualizador incluí-lo em uma solicitação.
- Quando possível, use uma política personalizada e inclua o endereço IP do visualizador.
- No atributo `CloudFront-Expires`, especifique o menor tempo de expiração razoável com base em quanto tempo você deseja que os usuários tenham acesso a seu conteúdo.

Quando o CloudFront confere a data e hora de validade de um cookie assinado

Para determinar se um signed cookie continua válido, o CloudFront verifica a data e hora de expiração dele no momento da solicitação HTTP. Se um cliente começar a fazer download de um

grande arquivo logo antes da hora de expiração, o download será concluído mesmo se passar a hora de expiração durante o download. Se a conexão TCP cair e o cliente tentar reiniciar o download após a hora de expiração, ocorrerá falha no download.

Se o cliente usar Range GETs para obter um arquivo em partes menores, ocorrerá falha em qualquer solicitação GET que ocorrer após a hora de expiração. Para obter mais informações sobre Range GETs, consulte [Como o CloudFront processa solicitações parciais de um objeto \(Range GETs\)](#).

Código de exemplo e ferramentas de terceiros

O código de exemplo do conteúdo privado mostra apenas como criar a assinatura para signed URLs. No entanto, o processo de criação de uma assinatura para um signed cookie é semelhante, ou seja, a maior parte do código de exemplo é relevante. Para obter mais informações, consulte os tópicos a seguir.

- [Criar uma assinatura de URL usando Perl](#)
- [Criar uma assinatura de URL usando PHP](#)
- [Criar uma assinatura de URL usando C# e o .NET Framework](#)
- [Criar uma assinatura de URL usando Java](#)

Definir cookies assinados usando uma política predefinida

Para definir um signed cookie usando uma política padrão, execute as seguintes etapas. Para criar a assinatura, consulte [Criar uma assinatura para um cookie assinado que use uma política predefinida](#).

Para definir um signed cookie usando uma política padrão

1. Se estiver usando o .NET ou o Java para criar signed cookies e não tiver reformatado a chave privada de seu par de chaves do formato padrão .pem para um formato compatível com o .NET ou o Java, faça isso agora. Para obter mais informações, consulte [Reformatar a chave privada \(somente .NET e Java\)](#).
2. Programe seu aplicativo para enviar três cabeçalhos Set-Cookie para os visualizadores aprovados. São necessários três cabeçalhos Set-Cookie porque cada cabeçalho Set-Cookie pode conter somente um par de nome/valor, e um signed cookie do CloudFront requer três pares. Os pares de nome-valor são: CloudFront-Expires, CloudFront-Signature e CloudFront-Key-Pair-Id. Os valores devem estar presentes no visualizador antes de um usuário fazer a primeira solicitação de um arquivo ao qual você deseja controlar o acesso.

Note

Em geral, recomendamos que você exclua os atributos Expires e Max-Age. A exclusão dos atributos faz com que o navegador exclua o cookie quando o usuário fecha o navegador, diminuindo a possibilidade de alguém obter acesso não autorizado ao seu conteúdo. Para obter mais informações, consulte [Evitar o uso indevido de cookies assinados](#).

Os nomes dos atributos de cookie fazem distinção entre letras maiúsculas e minúsculas.

As quebras de linha são incluídas apenas para tornar os atributos mais legíveis.

```
Set-Cookie:  
CloudFront-Expires=date and time in Unix time format (in seconds) and Coordinated  
Universal Time (UTC);  
Domain=optional domain name;  
Path=/optional directory path;  
Secure;  
HttpOnly  
  
Set-Cookie:  
CloudFront-Signature=hashed and signed version of the policy statement;  
Domain=optional domain name;  
Path=/optional directory path;  
Secure;  
HttpOnly  
  
Set-Cookie:  
CloudFront-Key-Pair-Id=public key ID for the CloudFront public key whose  
corresponding private key you're using to generate the signature;  
Domain=optional domain name;  
Path=/optional directory path;  
Secure;  
HttpOnly
```

(Opcional) Domain

O nome de domínio do arquivo solicitado. Se você não especificar um atributo Domain, o valor padrão será o nome de domínio do URL, e ele se aplica apenas ao nome de domínio

em questão, não aos subdomínios. Se você especificar um atributo `Domain`, ele também será aplicado aos subdomínios. Um ponto inicial no nome de domínio (por exemplo, `Domain=.example.com`) é opcional. Além disso, se você especificar um atributo `Domain`, o nome de domínio do URL e o valor do atributo `Domain` deverão ser correspondentes.

É possível especificar o nome de domínio atribuído pelo CloudFront à sua distribuição, por exemplo, `d111111abcdef8.cloudfront.net`, mas é possível especificar `*.cloudfront.net` para o nome de domínio.

Se você quiser usar um nome de domínio alternativo, como `example.com` nos URLs, deverá adicioná-lo à sua distribuição, independentemente de especificá-lo no atributo `Domain` ou não. Para obter mais informações, consulte [Nomes de domínio alternativos \(CNAMEs\)](#) no tópico [Referência de configurações da distribuição](#).

(Opcional) **Path**

O caminho do arquivo solicitado. Se você não especificar um atributo `Path`, o valor padrão será o caminho do URL.

Secure

Exige que o visualizador criptografe os cookies antes de enviar uma solicitação. Recomendamos que você envie o cabeçalho `Set-Cookie` por uma conexão HTTPS para garantir que os atributos de cookie estejam protegidos de ataques a intermediários.

HttpOnly

Define como o navegador (quando compatível) interage com o valor do cookie. Com `HttpOnly`, os valores dos cookies são inacessíveis ao JavaScript. Essa precaução pode ajudar a mitigar os ataques conhecidos como `cross-site scripting (XSS)`. Consulte mais informações em [Using HTTP cookies](#).

CloudFront-Expires

Especifique a data e hora de expiração no formato de hora do Unix (em segundos) e no Tempo Universal Coordenado (UTC). Por exemplo, 1º de janeiro de 2013 10h UTC é convertido para 1357034400 no formato de hora do Unix. Para usar o horário epoch, use um número inteiro de 32 bits para uma data que pode ser até 2147483647 (19 de janeiro de 2038 às 03:14:07 UTC). Para obter informações sobre UTC, consulte a RFC 3339, `Date and Time on the Internet: Timestamps`, <https://tools.ietf.org/html/rfc3339>.

CloudFront-Signature

Uma versão assinada, com hash e codificação base64 de uma declaração de política do JSON. Para obter mais informações, consulte [Criar uma assinatura para um cookie assinado que use uma política predefinida](#).

CloudFront-Key-Pair-Id

O ID de uma chave pública do CloudFront, por exemplo, K2JJCJMDEHXQW5F. O ID da chave pública informa ao CloudFront qual chave pública deve ser usada para validar o signed URL. O CloudFront compara as informações da assinatura com as informações da declaração de política para verificar se o URL não foi adulterado.

Essa chave pública deve pertencer a um grupo de chaves que seja um signatário confiável na distribuição. Para obter mais informações, consulte [Especificar os assinantes que podem criar URLs e cookies assinados](#).

O exemplo a seguir mostra cabeçalhos de Set-Cookie de um signed cookie quando você está usando o nome de domínio associado à sua distribuição nas URLs de seus arquivos:

```
Set-Cookie: CloudFront-Expires=1426500000; Domain=d111111abcdef8.cloudfront.net; Path=/images/*; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=yXrSIgyQoeE4FBI4eMKF6ho~CA8_; Domain=d111111abcdef8.cloudfront.net; Path=/images/*; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=K2JJCJMDEHXQW5F; Domain=d111111abcdef8.cloudfront.net; Path=/images/*; Secure; HttpOnly
```

O exemplo a seguir mostra cabeçalhos de Set-Cookie de um signed cookie quando você está usando o nome de domínio alternativo example.org nas URLs de seus arquivos:

```
Set-Cookie: CloudFront-Expires=1426500000; Domain=example.org; Path=/images/*; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=yXrSIgyQoeE4FBI4eMKF6ho~CA8_; Domain=example.org; Path=/images/*; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=K2JJCJMDEHXQW5F; Domain=example.org; Path=/images/*; Secure; HttpOnly
```

Se você quiser usar um nome de domínio alternativo, como example.com nos URLs, deverá adicioná-lo à sua distribuição, independentemente de especificá-lo no atributo Domain ou não. Para

obter mais informações, consulte [Nomes de domínio alternativos \(CNAMEs\)](#) no tópico [Referência de configurações da distribuição](#).

Criar uma assinatura para um cookie assinado que use uma política predefinida

Para criar a assinatura de um cookie que use uma política predefinida, siga estes procedimentos.

Tópicos

- [Criar uma declaração de política para um cookie assinado que use uma política predefinida](#)
- [Assinar a declaração de política para criar uma assinatura para um cookie assinado que use uma política predefinida](#)

Criar uma declaração de política para um cookie assinado que use uma política predefinida

Ao definir um signed cookie que usa uma política padrão, o atributo `CloudFront-Signature` será uma versão assinada e com hash de uma declaração de política. Para signed cookies que usam uma política padrão, a declaração de política não é incluída no cabeçalho `Set-Cookie`, como é feito nos signed cookies que usam uma política personalizada. Para criar a declaração de política, conclua as etapas a seguir.

Para criar uma declaração de política para um signed cookie que usa uma política padrão

1. Crie a declaração de política usando o formato JSON a seguir e a codificação de caracteres UTF-8. Inclua todas as pontuações e outros valores literais exatamente como especificado. Para obter informações sobre os parâmetros `Resource` e `DateLessThan`, consulte [Valores especificados na declaração de uma política padrão para signed cookies](#).

```
{
  "Statement": [
    {
      "Resource": "base URL or stream name",
      "Condition": {
        "DateLessThan": {
          "AWS:EpochTime": ending date and time in Unix time format and
          UTC
        }
      }
    }
  ]
}
```


2. Remova todas os espaços em branco (inclusive caracteres de nova linha e de tabulação) da declaração de política. Pode ser necessário incluir caracteres de escape na string do código do aplicativo.

Valores especificados na declaração de uma política padrão para signed cookies

Ao criar uma declaração de política para uma política padrão, especifique os valores a seguir:

Recurso

O URL base, inclusive suas query strings, se houver, por exemplo:

```
https://d1111111abcdef8.cloudfront.net/images/horizon.jpg?  
size=large&license=yes
```

Você pode especificar apenas um valor para Resource.

Observe o seguinte:

- Protocolo: o valor deve começar com `http://` ou `https://`.
- Parâmetros de query string :se você não tiver query strings, omita o ponto de interrogação.
- Nomes de domínio alternativos: se especificar um nome de domínio alternativo (CNAME) no URL, você deverá especificá-lo ao fazer referência ao arquivo na sua página da web ou aplicação. Não especifique o URL do Amazon S3 para o arquivo.

DateLessThan

A data e hora de expiração do URL no formato de hora do Unix (em segundos) e no Tempo Universal Coordenado (UTC). Não coloque os valores entre aspas.

Por exemplo, 16 de março de 2015, 10h UTC é convertido para 1426500000 no formato de hora do Unix.

Esse valor deve corresponder ao valor do atributo `CloudFront-Expires` no cabeçalho `Set-Cookie`. Não coloque os valores entre aspas.

Para obter mais informações, consulte [Quando o CloudFront confere a data e hora de validade de um cookie assinado](#).

Exemplo de declaração de política para uma política padrão

Ao usar o seguinte exemplo de declaração de política em um signed cookie, um usuário pode acessar o arquivo `https://d111111abcdef8.cloudfront.net/horizon.jpg` até 16 de março de 2015, 10h UTC:

```
{
  "Statement": [
    {
      "Resource": "https://d111111abcdef8.cloudfront.net/horizon.jpg?
size=large&license=yes",
      "Condition": {
        "DateLessThan": {
          "AWS:EpochTime": 1426500000
        }
      }
    }
  ]
}
```

Assinar a declaração de política para criar uma assinatura para um cookie assinado que use uma política predefinida

Para criar o valor para o atributo `CloudFront-Signature` em um cabeçalho `Set-Cookie`, assine e adicione um hash à declaração de política criada em [Para criar uma declaração de política para um signed cookie que usa uma política padrão](#).

Para obter mais informações e exemplos de como adicionar hash, assinar e codificar a declaração de política, consulte os seguintes tópicos:

- [Comandos do Linux e OpenSSL para criptografia e codificação base64](#)
- [Exemplos de código para criar uma assinatura para um signed URL](#)

Para criar uma assinatura para um signed cookie usando uma política padrão

1. Use a função de hash SHA-1 e o RSA para assinar e adicionar um hash à declaração de política criada no procedimento [Para criar uma declaração de política para um signed cookie que usa uma política padrão](#). Use a versão da declaração de política que não inclui mais espaços em branco.

Para a chave privada exigida pela função hash, use uma chave privada que tenha a chave pública em um grupo de chaves confiável ativo para a distribuição.

Note

O método usado para assinar e adicionar um hash à declaração de política depende da sua linguagem de programação e plataforma. Para obter o código de exemplo, consulte [Exemplos de código para criar uma assinatura para um signed URL](#).

2. Remova os espaços em branco (inclusive caracteres de nova linha e de tabulação) da string assinada e com hash.
3. Codifique a string usando codificação base64 MIME. Para obter mais informações, consulte [Section 6.8, Base64 Content-Transfer-Encoding](#) em RFC 2045, MIME (Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies.
4. Substitua os caracteres inválidos da query string de um URL por caracteres válidos. A tabela a seguir indica os caracteres válidos e inválidos.

Substitua esses caracteres inválidos	Por esses caracteres válidos
+	- (hífen)
=	_ (sublinhado)
/	~ (til)

5. Inclua o valor resultante no cabeçalho Set-Cookie para o par de nome-valor CloudFront-Signature. Em seguida, volte para [Para definir um signed cookie usando uma política padrão](#) e adicione o cabeçalho Set-Cookie em CloudFront-Key-Pair-Id.

Definir cookies assinados usando uma política personalizada

Para definir um signed cookie que usa uma política personalizada, execute as etapas a seguir.

Para definir um signed cookie usando uma política personalizada

1. Se você estiver usando o .NET ou Java para criar signed URLs e não tiver reformatado a chave privada do seu par de chaves para do formato padrão .pem para um formato compatível com

- o .NET ou Java, faça isso agora. Para obter mais informações, consulte [Reformatar a chave privada \(somente .NET e Java\)](#).
2. Programe seu aplicativo para enviar três cabeçalhos Set-Cookie para os visualizadores aprovados. São necessários três cabeçalhos Set-Cookie porque cada cabeçalho Set-Cookie pode conter somente um par de nome/valor, e um signed cookie do CloudFront requer três pares. Os pares de nome-valor são: CloudFront-Policy, CloudFront-Signature e CloudFront-Key-Pair-Id. Os valores devem estar presentes no visualizador antes de um usuário fazer a primeira solicitação de um arquivo ao qual você deseja controlar o acesso.

Note

Em geral, recomendamos que você exclua os atributos Expires e Max-Age. Isso faz com que o navegador exclua o cookie quando o usuário fecha o navegador, diminuindo a possibilidade de alguém obter acesso não autorizado ao seu conteúdo. Para obter mais informações, consulte [Evitar o uso indevido de cookies assinados](#).

Os nomes dos atributos de cookie fazem distinção entre letras maiúsculas e minúsculas.

As quebras de linha são incluídas apenas para tornar os atributos mais legíveis.

```
Set-Cookie:  
CloudFront-Policy=base64 encoded version of the policy statement;  
Domain=optional domain name;  
Path=/optional directory path;  
Secure;  
HttpOnly  
  
Set-Cookie:  
CloudFront-Signature=hashed and signed version of the policy statement;  
Domain=optional domain name;  
Path=/optional directory path;  
Secure;  
HttpOnly  
  
Set-Cookie:  
CloudFront-Key-Pair-Id=public key ID for the CloudFront public key whose  
corresponding private key you're using to generate the signature;  
Domain=optional domain name;
```

```
Path=/optional directory path;  
Secure;  
HttpOnly
```

(Opcional) **Domain**

O nome de domínio do arquivo solicitado. Se você não especificar um atributo `Domain`, o valor padrão será o nome de domínio do URL, e ele se aplica apenas ao nome de domínio em questão, não aos subdomínios. Se você especificar um atributo `Domain`, ele também será aplicado aos subdomínios. Um ponto inicial no nome de domínio (por exemplo, `Domain=.example.com`) é opcional. Além disso, se você especificar um atributo `Domain`, o nome de domínio do URL e o valor do atributo `Domain` deverão ser correspondentes.

É possível especificar o nome de domínio atribuído pelo CloudFront à sua distribuição, por exemplo, `d111111abcdef8.cloudfront.net`, mas é possível especificar `*.cloudfront.net` para o nome de domínio.

Se você quiser usar um nome de domínio alternativo, como `example.com` nos URLs, deverá adicioná-lo à sua distribuição, independentemente de especificá-lo no atributo `Domain` ou não. Para obter mais informações, consulte [Nomes de domínio alternativos \(CNAMEs\)](#) no tópico [Referência de configurações da distribuição](#).

(Opcional) **Path**

O caminho do arquivo solicitado. Se você não especificar um atributo `Path`, o valor padrão será o caminho do URL.

Secure

Exige que o visualizador criptografe os cookies antes de enviar uma solicitação. Recomendamos que você envie o cabeçalho `Set-Cookie` por uma conexão HTTPS para garantir que os atributos de cookie estejam protegidos de ataques a intermediários.

HttpOnly

Exige que o visualizador envie o cookie apenas em solicitações HTTP ou HTTPS.

CloudFront-Policy

Sua declaração de política no formato JSON, sem espaços em branco e com codificação base64. Para ter mais informações, consulte [Criar uma assinatura para um cookie assinado que use uma política personalizada](#).

A declaração de política controla o acesso que um signed cookie concede a um usuário. Ela inclui os arquivos que o usuário pode acessar, uma data e hora de expiração, uma data e hora opcionais em que o URL se torna válido e um endereço IP opcional ou intervalo de endereços IP que tenham permissão para acessar o arquivo.

CloudFront-Signature

Uma versão assinada, com hash e codificação base64 da declaração de política do JSON. Para obter mais informações, consulte [Criar uma assinatura para um cookie assinado que use uma política personalizada](#).

CloudFront-Key-Pair-Id

O ID de uma chave pública do CloudFront, por exemplo, K2JJCJMDEHXQW5F. O ID da chave pública informa ao CloudFront qual chave pública deve ser usada para validar o signed URL. O CloudFront compara as informações da assinatura com as informações da declaração de política para verificar se o URL não foi adulterado.

Essa chave pública deve pertencer a um grupo de chaves que seja um signatário confiável na distribuição. Para ter mais informações, consulte [Especificar os assinantes que podem criar URLs e cookies assinados](#).

Exemplos de cabeçalhos **Set-Cookie** para políticas personalizadas

Veja os exemplos a seguir de pares de cabeçalhos Set-Cookie.

Se quiser usar um nome de domínio alternativo, como exemplo.org nos URLs, deverá adicioná-lo à sua distribuição, independentemente de especificá-lo no atributo Domain ou não. Para obter mais informações, consulte [Nomes de domínio alternativos \(CNAMEs\)](#) no tópico [Referência de configurações da distribuição](#).

Example Exemplo 1

É possível usar os cabeçalhos Set-Cookie de um signed cookie quando estiver usando o nome de domínio associado à sua distribuição nos URLs dos seus arquivos.

```
Set-Cookie: CloudFront-  
Policy=eyJTdGF0ZW11bnQiO1t7I1Jlc291cmN1IjoiaHR0cDovL2QxMTEyMTFhYmNkZWY4LmNsb3VkZnJvbnQubmV0L2dh  
Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly  
Set-Cookie: CloudFront-Signature=dtKhpJ3aUYxqDIwepczPiDb9NXQ_  
Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
```

```
Set-Cookie: CloudFront-Key-Pair-Id=K2JJCJMDEHXQW5F;
Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
```

Example Exemplo 2

É possível usar cabeçalhos Set-Cookie de um signed cookie quando você estiver usando um nome de domínio alternativo (exemplo.org) nos URLs dos seus arquivos.

```
Set-Cookie: CloudFront-
Policy=eyJTdGF0ZWl1bnQiO1t7I1Jlc291cmNlIjoiaHR0cDovL2QxMTEyMTFhYmNkZWY4LmNsb3VkZnJvbnQubmV0L2dh
Domain=example.org; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=dtKhpJ3aUYxqDIwepczPiDb9NXQ_; Domain=example.org;
Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=K2JJCJMDEHXQW5F; Domain=example.org; Path=/; Secure;
HttpOnly
```

Example Exemplo 3

É possível usar os pares de cabeçalho Set-Cookie de uma solicitação assinada quando estiver usando o nome de domínio associado à sua distribuição nos URLs dos seus arquivos.

```
Set-Cookie: CloudFront-
Policy=eyJTdGF0ZWl1bnQiO1t7I1Jlc291cmNlIjoiaHR0cDovL2QxMTEyMTFhYmNkZWY4LmNsb3VkZnJvbnQubmV0L2dh
Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=dtKhpJ3aUYxqDIwepczPiDb9NXQ_;
Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=K2JJCJMDEHXQW5F;
Domain=dd111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
```

Example Exemplo 4

É possível usar os pares de cabeçalho Set-Cookie de uma solicitação assinada quando estiver usando um nome de domínio alternativo (exemplo.org) associado à sua distribuição nos URLs dos seus arquivos.

```
Set-Cookie: CloudFront-
Policy=eyJTdGF0ZWl1bnQiO1t7I1Jlc291cmNlIjoiaHR0cDovL2QxMTEyMTFhYmNkZWY4LmNsb3VkZnJvbnQubmV0L2dh
Domain=example.org; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=dtKhpJ3aUYxqDIwepczPiDb9NXQ_; Domain=example.org;
Path=/; Secure; HttpOnly
```

```
Set-Cookie: CloudFront-Key-Pair-Id=K2JJCJMDEHXQW5F; Domain=example.org; Path=/; Secure; HttpOnly
```

Criar uma declaração de política para um cookie assinado que use uma política personalizada

Para criar uma declaração de política para uma política personalizada, conclua as etapas a seguir. Para obter vários exemplos de declaração de política que controlam o acesso a arquivos de diversas formas, consulte [Exemplos de declaração de política para um signed cookie que usa uma política personalizada](#).

Para criar a declaração de política para um signed cookie que usa uma política personalizada

1. Crie a declaração de política usando o formato JSON a seguir.

```
{
  "Statement": [
    {
      "Resource": "URL of the file",
      "Condition": {
        "DateLessThan": {
          "AWS:EpochTime": "required ending date and time in Unix time
format and UTC"
        },
        "DateGreaterThan": {
          "AWS:EpochTime": "optional beginning date and time in Unix time
format and UTC"
        },
        "IpAddress": {
          "AWS:SourceIp": "optional IP address"
        }
      }
    }
  ]
}
```

Observe o seguinte:

- Você pode incluir apenas uma instrução.
- Use a codificação de caracteres UTF-8.
- Inclua todas as pontuações e nomes de parâmetro exatamente como especificado. Abreviações de nomes de parâmetro não são aceitas.

- A ordem dos parâmetros na seção `Condition` não é importante.
 - Para obter informações sobre os valores de `Resource`, `DateLessThan`, `DateGreaterThan` e `IpAddress`, consulte [Valores especificados na declaração de uma política personalizada para signed cookies](#).
2. Remova todas os espaços em branco (inclusive caracteres de nova linha e de tabulação) da declaração de política. Pode ser necessário incluir caracteres de escape na string do código do aplicativo.
 3. Codifique a declaração de política usando codificação base64 MIME. Para obter mais informações, consulte [Section 6.8, Base64 Content-Transfer-Encoding](#) em RFC 2045, MIME (Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies.
 4. Substitua os caracteres inválidos da query string de um URL por caracteres válidos. A tabela a seguir indica os caracteres válidos e inválidos.

Substitua esses caracteres inválidos	Por esses caracteres válidos
+	- (hífen)
=	_ (sublinhado)
/	~ (til)

5. Inclua o valor resultante em seu cabeçalho `Set-Cookie` depois de `CloudFront-Policy=`.
6. Crie uma assinatura para o cabeçalho `Set-Cookie` em `CloudFront-Signature` adicionando hash, assinando e codificando em base64 a declaração de política. Para obter mais informações, consulte [Criar uma assinatura para um cookie assinado que use uma política personalizada](#).

Valores especificados na declaração de uma política personalizada para signed cookies

Ao criar uma declaração de política para uma política personalizada, especifique os valores a seguir.

Recurso

O URL base, inclusive suas query strings, se houver:

```
https://d111111abcdef8.cloudfront.net/images/horizon.jpg?  
size=large&license=yes
```

Important

Se você omitir o parâmetro `Resource`, os usuários poderão acessar todos os arquivos associados a qualquer distribuição associada ao par de chaves usado para criar o signed URL.

Você pode especificar apenas um valor para `Resource`.

Observe o seguinte:

- Protocolo: o valor deve começar com `http://` ou `https://`.
- Parâmetros de query string :se você não tiver query strings, omita o ponto de interrogação.
- Curingas: é possível usar o caractere curinga que corresponde a zero ou mais caracteres (*) ou o caractere curinga que corresponde a exatamente um caractere (?) em qualquer lugar na string. Por exemplo, o valor:

```
https://d111111abcdef8.cloudfront.net/*game_download.zip*
```

incluiria os seguintes arquivos:

- `https://d111111abcdef8.cloudfront.net/game_download.zip`
- `https://d111111abcdef8.cloudfront.net/example_game_download.zip?license=yes`
- `https://d111111abcdef8.cloudfront.net/test_game_download.zip?license=temp`
- Nomes de domínio alternativos: se especificar um nome de domínio alternativo (CNAME) no URL, você deverá especificá-lo ao fazer referência ao arquivo na sua página da web ou aplicação. Não especifique o URL do Amazon S3 para o arquivo.

DateLessThan

A data e hora de expiração do URL no formato de hora do Unix (em segundos) e no Tempo Universal Coordenado (UTC). Não coloque os valores entre aspas.

Por exemplo, 16 de março de 2015, 10h UTC é convertido para 1426500000 no formato de hora do Unix.

Para obter mais informações, consulte [Quando o CloudFront confere a data e hora de validade de um cookie assinado](#).

DateGreaterThan (opcional)

Uma data e hora de início opcional do URL no formato de hora do Unix (em segundos) e no Tempo Universal Coordenado (UTC). Os usuários não podem acessar o arquivo antes ou na data e hora especificadas. Não coloque os valores entre aspas.

IpAddress (opcional)

O endereço IP do cliente que está fazendo a solicitação GET. Observe o seguinte:

- Para permitir o acesso de qualquer endereço IP ao arquivo, omita o parâmetro `IpAddress`.
- Você pode especificar um ou vários endereços IP. Por exemplo, você não pode definir que a política permita o acesso se o endereço IP do cliente estiver em um de dois intervalos separados.
- Para permitir o acesso de um único endereço IP, especifique:

"IPv4 IP address/32"

- Você deve especificar os intervalos de endereço IP no formato CIDR IPv4 padrão (por exemplo, `192.0.2.0/24`). Para obter mais informações, acesse a RFC 4632, Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan, <https://tools.ietf.org/html/rfc4632>.

Important

Endereços IP no formato IPv6, como `2001:0db8:85a3::8a2e:0370:7334`, não são compatíveis.

Se você estiver usando uma política personalizada que inclui `IpAddress`, não permita o IPv6 para a distribuição. Se você quiser restringir o acesso a um conteúdo por endereço IP e oferecer suporte a solicitações IPv6 para outro tipo de conteúdo, crie duas distribuições. Para obter mais informações, consulte [Enable IPv6](#) no tópico [Referência de configurações da distribuição](#).

Exemplos de declaração de política para um signed cookie que usa uma política personalizada

Os exemplos de declaração de política a seguir mostram como controlar o acesso a um arquivo específico, todos os arquivos de um diretório ou todos os arquivos associados a um ID de par de chaves. Os exemplos também mostram como controlar o acesso de um único endereço IP ou um intervalo de endereços IP e como evitar que os usuários usem o signed cookie após a data e hora especificadas.

Se você copiar e colar qualquer um desses exemplos, remova os espaços em branco (inclusive caracteres de nova linha e de tabulação), substitua os valores pelos seus próprios valores e inclua um caractere de nova linha após a chave de fechamento (}).

Para ter mais informações, consulte [Valores especificados na declaração de uma política personalizada para signed cookies](#).

Tópicos

- [Exemplo de declaração de política: acessar um arquivo de um intervalo de endereços IP](#)
- [Exemplo de declaração de política: acessar todos os arquivos de um diretório em um intervalo de endereços IP](#)
- [Exemplo de declaração de política: acessar todos os arquivos associados a um ID de par de chaves de um endereço IP](#)

Exemplo de declaração de política: acessar um arquivo de um intervalo de endereços IP

O exemplo a seguir de política personalizada em um signed cookie especifica que um usuário pode acessar o arquivo `https://d111111abcdef8.cloudfront.net/game_download.zip` de endereços IP no intervalo `192.0.2.0/24` até 1.º de janeiro de 2023, 10h UTC:

```
{
  "Statement": [
    {
      "Resource": "https://d111111abcdef8.cloudfront.net/game_download.zip",
      "Condition": {
        "IpAddress": {
          "AWS:SourceIp": "192.0.2.0/24"
        },
        "DateLessThan": {
          "AWS:EpochTime": 1357034400
        }
      }
    }
  ]
}
```

```

    }
  ]
}

```

Exemplo de declaração de política: acessar todos os arquivos de um diretório em um intervalo de endereços IP

O exemplo a seguir de política personalizada permite criar signed cookies para qualquer arquivo no diretório `training`, conforme indicado pelo caractere curinga "*" no parâmetro `Resource`. Os usuários podem acessar o arquivo de um endereço IP no intervalo `192.0.2.0/24` até 1º de janeiro de 2013, 10h UTC:

```

{
  "Statement": [
    {
      "Resource": "https://d111111abcdef8.cloudfront.net/training/*",
      "Condition": {
        "IpAddress": {
          "AWS:SourceIp": "192.0.2.0/24"
        },
        "DateLessThan": {
          "AWS:EpochTime": 1357034400
        }
      }
    }
  ]
}

```

Cada signed cookie em que você usa essa política inclui um URL base que identifica um arquivo específico, por exemplo:

`https://d111111abcdef8.cloudfront.net/training/orientation.pdf`

Exemplo de declaração de política: acessar todos os arquivos associados a um ID de par de chaves de um endereço IP

O exemplo a seguir de política personalizada permite definir signed cookies para qualquer arquivo associado a qualquer distribuição, conforme indicado pelo caractere curinga "*" no parâmetro `Resource`. O usuário deve usar o endereço IP `192.0.2.10/32`. (O valor `192.0.2.10/32` na notação CIDR se refere a um único endereço IP, `192.0.2.10`.) Os arquivos estão disponíveis apenas de 1º de janeiro de 2013, 10h UTC, a 2 de janeiro de 2013, 10h UTC:

```
{
  "Statement": [
    {
      "Resource": "https://*",
      "Condition": {
        "IpAddress": {
          "AWS:SourceIp": "192.0.2.10/32"
        },
        "DateGreaterThan": {
          "AWS:EpochTime": 1357034400
        },
        "DateLessThan": {
          "AWS:EpochTime": 1357120800
        }
      }
    }
  ]
}
```

Cada signed cookie em que você usa essa política inclui um URL base que identifica um arquivo específico em uma distribuição específica do CloudFront, por exemplo:

<https://d111111abcdef8.cloudfront.net/training/orientation.pdf>

O signed cookie também inclui o ID de um par de chaves, que deve estar associado a um grupo de chaves confiável na distribuição (d111111abcdef8.cloudfront.net) especificada no URL base.

Criar uma assinatura para um cookie assinado que use uma política personalizada

A assinatura de um signed cookie que usa uma política personalizada é uma versão da declaração de política com hash, assinada e codificada em base64.

Para obter mais informações e exemplos de como adicionar hash, assinar e codificar a declaração de política, consulte:

- [Comandos do Linux e OpenSSL para criptografia e codificação base64](#)
- [Exemplos de código para criar uma assinatura para um signed URL](#)

Para criar uma assinatura para um signed cookie usando uma política personalizada

1. Use a função de hash SHA-1 e o RSA para assinar e adicionar um hash à declaração de política do JSON criada no procedimento [Para criar a declaração de política para um signed URL que usa uma política personalizada](#). Use a versão da declaração de política que não inclui mais espaços em branco, mas que ainda não foi codificada em base64.

Para a chave privada exigida pela função hash, use uma chave privada que tenha a chave pública em um grupo de chaves confiável ativo para a distribuição.

Note

O método usado para assinar e adicionar um hash à declaração de política depende da sua linguagem de programação e plataforma. Para obter o código de exemplo, consulte [Exemplos de código para criar uma assinatura para um signed URL](#).

2. Remova os espaços em branco (inclusive caracteres de nova linha e de tabulação) da string assinada e com hash.
3. Codifique a string usando codificação base64 MIME. Para obter mais informações, consulte [Section 6.8, Base64 Content-Transfer-Encoding](#) em RFC 2045, MIME (Multipurpose Internet Mail Extensions) Part One: Format of Internet Message Bodies.
4. Substitua os caracteres inválidos da query string de um URL por caracteres válidos. A tabela a seguir indica os caracteres válidos e inválidos.

Substitua esses caracteres inválidos	Por esses caracteres válidos
+	- (hífen)
=	_ (sublinhado)
/	~ (til)

5. Inclua o valor resultante no cabeçalho Set-Cookie para o par de nome-valor CloudFront-Signature= e volte para [Para definir um signed cookie usando uma política personalizada](#) adicionar o cabeçalho Set-Cookie em CloudFront-Key-Pair-Id.

Comandos do Linux e OpenSSL para criptografia e codificação base64

Você pode usar o comando de linha de comando do Linux a seguir e o OpenSSL para adicionar hash e assinar a declaração de política, codificar a assinatura em base64 e substituir caracteres inválidos dos parâmetros de query string do URL por caracteres válidos.

Para obter informações sobre o OpenSSL, acesse <https://www.openssl.org>.

```
cat policy | tr -d "\n" | tr -d " \t\n\r" | openssl sha1 -sign private_key.pem |  
openssl base64 -A | tr -- '+=/' '-_~'
```

No comando anterior:

- `cat` lê o arquivo `policy`.
- `tr -d "\n" | tr -d " \t\n\r"` remove os espaços em branco e o caractere de nova linha que foram adicionados por `cat`.
- O OpenSSL adiciona hash ao arquivo usando SHA-1 e o assina usando RSA e o arquivo de chave privada `private_key.pem`.
- O OpenSSL codifica em base64 a instrução de política assinada e com hash.
- O `tr` substitui os caracteres inválidos dos parâmetros de string de consulta do URL pelos caracteres válidos.

Consulte mais exemplos de código que demonstram como criar uma assinatura em [Exemplos de código para criar uma assinatura para um signed URL](#).

Exemplos de código para criar uma assinatura para um signed URL

Esta seção inclui exemplos de aplicativos para download que demonstram como criar assinaturas para signed URLs. Os exemplos estão disponíveis em Perl, PHP, C# e Java. Você pode usar qualquer um dos exemplos para criar signed URLs. O script Perl é executado nas plataformas Linux e macOS. O exemplo PHP funciona em qualquer servidor com PHP. O exemplo C# usa o .NET Framework.

Para obter um código de exemplo em JavaScript (Node.js), consulte [Criar URLs assinados do Amazon CloudFront em Node.js](#) no Blog do desenvolvedor da AWS.

Para obter um código de exemplo em Python, consulte [Gerar um signed URL para o Amazon CloudFront](#) na Referência da API do AWS SDK for Python (Boto3) e [este código de exemplo](#) no repositório GitHub do Boto3.

Tópicos

- [Criar uma assinatura de URL usando Perl](#)
- [Criar uma assinatura de URL usando PHP](#)
- [Criar uma assinatura de URL usando C# e o .NET Framework](#)
- [Criar uma assinatura de URL usando Java](#)

Criar uma assinatura de URL usando Perl

Esta seção inclui um script Perl para plataformas Linux/Mac que você pode usar para criar a assinatura para conteúdo privado. Para criar a assinatura, execute o script com argumentos de linha de comando que especificam o URL do CloudFront, o caminho para a chave privada do assinante, o ID da chave e uma data de expiração para o URL. A ferramenta também pode decodificar signed URLs.

Note

A criação de uma assinatura de URL é apenas uma parte do processo de fornecimento de conteúdo privado usando um signed URL. Para obter mais informações sobre o todo o processo, consulte [Usar URLs assinados](#).

Tópicos

- [Fonte do script Perl para criar um signed URL](#)

Fonte do script Perl para criar um signed URL

O seguinte código-fonte Perl pode ser usado para criar uma URL assinada para CloudFront. Os comentários no código incluem informações sobre as opções da linha de comando e os recursos da ferramenta.

```
#!/usr/bin/perl -w
```

```
# Copyright 2008 Amazon Technologies, Inc. Licensed under the Apache License, Version
2.0 (the "License");
# you may not use this file except in compliance with the License. You may obtain a
copy of the License at:
#
# https://aws.amazon.com/apache2.0
#
# This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
KIND, either express or implied.
# See the License for the specific language governing permissions and limitations under
the License.
```

```
=head1 cfsign.pl
```

cfsign.pl - A tool to generate and verify Amazon CloudFront signed URLs

```
=head1 SYNOPSIS
```

This script uses an existing RSA key pair to sign and verify Amazon CloudFront signed URLs

View the script source for details as to which CPAN packages are required beforehand.

For help, try:

```
cfsign.pl --help
```

URL signing examples:

```
cfsign.pl --action encode --url https://images.my-website.com/gallery1.zip --policy
sample_policy.json --private-key privkey.pem --key-pair-id mykey
```

```
cfsign.pl --action encode --url https://images.my-website.com/gallery1.zip --expires
1257439868 --private-key privkey.pem --key-pair-id mykey
```

URL decode example:

```
cfsign.pl --action decode --url "http://mydist.cloudfront.net/?Signature=AG0-
PgXkYo99MkJFHvjfGXjG1QDEXeaDb4Qtzmy85wqyJjK7eKojQWa4BCRCow__&Policy=eyJTdGF0ZWw1bnQiO1t7I1Jlc29
Pair-Id=mykey"
```

To generate an RSA key pair, you can use openssl and the following commands:

```
# Generate a 2048 bit key pair
openssl genrsa -out private-key.pem 2048
openssl rsa -in private-key.pem -pubout -out public-key.pem
```

=head1 OPTIONS

=over 8

=item B<--help>

Print a help message and exits.

=item B<--action> [action]

The action to execute. action can be one of:

- encode - Generate a signed URL (using a canned policy or a user policy)
- decode - Decode a signed URL

=item B<--url>

The URL to en/decode

=item B<--stream>

The stream to en/decode

=item B<--private-key>

The path to your private key.

=item B<--key-pair-id>

The key pair identifier.

=item B<--policy>

The CloudFront policy document.

=item B<--expires>

The Unix epoch time when the URL is to expire. If both this option and the --policy option are specified, --policy will be used. Otherwise, this

```
option alone will use a canned policy.

=back

=cut

use strict;
use warnings;

# you might need to use CPAN to get these modules.
# run perl -MCPAN -e "install <module>" to get them.
# The openssl command line will also need to be in your $PATH.
use File::Temp qw/tempfile/;
use File::Slurp;
use Getopt::Long;
use IPC::Open2;
use MIME::Base64 qw(encode_base64 decode_base64);
use Pod::Usage;
use URI;

my $CANNED_POLICY
    = '{"Statement":[{"Resource":"<RESOURCE>","Condition":{"DateLessThan":
{"AWS:EpochTime":<EXPIRES>}}}}]';

my $POLICY_PARAM      = "Policy";
my $EXPIRES_PARAM     = "Expires";
my $SIGNATURE_PARAM  = "Signature";
my $KEY_PAIR_ID_PARAM = "Key-Pair-Id";

my $verbose = 0;
my $policy_filename = "";
my $expires_epoch = 0;
my $action = "";
my $help = 0;
my $key_pair_id = "";
my $url = "";
my $stream = "";
my $private_key_filename = "";

my $result = GetOptions("action=s"      => \$action,
                       "policy=s"      => \$policy_filename,
                       "expires=i"     => \$expires_epoch,
                       "private-key=s" => \$private_key_filename,
                       "key-pair-id=s" => \$key_pair_id,
```

```
        "verbose"      => \$verbose,
        "help"        => \$help,
        "url=s"       => \$url,
        "stream=s"    => \$stream,
    );

if ($help or !$result) {
    pod2usage(1);
    exit;
}

if ($url eq "" and $stream eq "") {
    print STDERR "Must include a stream or a URL to encode or decode with the --stream
or --url option\n";
    exit;
}

if ($url ne "" and $stream ne "") {
    print STDERR "Only one of --url and --stream may be specified\n";
    exit;
}

if ($url ne "" and !is_url_valid($url)) {
    exit;
}

if ($stream ne "") {
    exit unless is_stream_valid($stream);

    # The signing mechanism is identical, so from here on just pretend we're
    # dealing with a URL
    $url = $stream;
}

if ($action eq "encode") {
    # The encode action will generate a private content URL given a base URL,
    # a policy file (or an expires timestamp) and a key pair id parameter
    my $private_key;
    my $public_key;
    my $public_key_file;

    my $policy;
    if ($policy_filename eq "") {
        if ($expires_epoch == 0) {
```

```
        print STDERR "Must include policy filename with --policy argument or an
expires" .
            "time using --expires\n";
    }

    $policy = $CANNED_POLICY;
    $policy =~ s/<EXPIRES>/$expires_epoch/g;
    $policy =~ s/<RESOURCE>/$url/g;
} else {
    if (! -e $policy_filename) {
        print STDERR "Policy file $policy_filename does not exist\n";
        exit;
    }
    $expires_epoch = 0; # ignore if set
    $policy = read_file($policy_filename);
}

if ($private_key_filename eq "") {
    print STDERR "You must specific the path to your private key file with --
private-key\n";
    exit;
}

if (! -e $private_key_filename) {
    print STDERR "Private key file $private_key_filename does not exist\n";
    exit;
}

if ($key_pair_id eq "") {
    print STDERR "You must specify a key pair id with --key-pair-id\n";
    exit;
}

my $encoded_policy = url_safe_base64_encode($policy);
my $signature = rsa_sha1_sign($policy, $private_key_filename);
my $encoded_signature = url_safe_base64_encode($signature);

my $generated_url = create_url($url, $encoded_policy, $encoded_signature,
$key_pair_id, $expires_epoch);

if ($stream ne "") {
    print "Encoded stream (for use within a swf):\n" . $generated_url . "\n";
}
```

```
        print "Encoded and escaped stream (for use on a webpage):\n" .
escape_url_for_webpage($generated_url) . "\n";
    } else {
        print "Encoded URL:\n" . $generated_url . "\n";
    }
} elsif ($action eq "decode") {
    my $decoded = decode_url($url);
    if (!$decoded) {
        print STDERR "Improperly formed URL\n";
        exit;
    }

    print_decoded_url($decoded);
} else {
    # No action specified, print help.  But only if this is run as a program (caller
    will be empty)
    pod2usage(1) unless caller();
}

# Decode a private content URL into its component parts
sub decode_url {
    my $url = shift;

    if ($url =~ /(.*?)\?(.*)/) {
        my $base_url = $1;
        my $params = $2;

        my @unparsed_params = split(/&/, $params);
        my %params = ();
        foreach my $param (@unparsed_params) {
            my ($key, $val) = split(/=/, $param);
            $params{$key} = $val;
        }

        my $encoded_signature = "";
        if (exists $params{$SIGNATURE_PARAM}) {
            $encoded_signature = $params{"Signature"};
        } else {
            print STDERR "Missing Signature URL parameter\n";
            return 0;
        }

        my $encoded_policy = "";
        if (exists $params{$POLICY_PARAM}) {
```

```
    $encoded_policy = $params{$POLICY_PARAM};
} else {
    if (!exists $params{$EXPIRES_PARAM}) {
        print STDERR "Either the Policy or Expires URL parameter needs to be
specified\n";
        return 0;
    }

    my $expires = $params{$EXPIRES_PARAM};

    my $policy = $CANNED_POLICY;
    $policy =~ s/<EXPIRES>/$expires/g;

    my $url_without_cf_params = $url;
    $url_without_cf_params =~ s/$SIGNATURE_PARAM=[^&]*&?//g;
    $url_without_cf_params =~ s/$POLICY_PARAM=[^&]*&?//g;
    $url_without_cf_params =~ s/$EXPIRES_PARAM=[^&]*&?//g;
    $url_without_cf_params =~ s/$KEY_PAIR_ID_PARAM=[^&]*&?//g;

    if ($url_without_cf_params =~ /(.*?)\?$/) {
        $url_without_cf_params = $1;
    }

    $policy =~ s/<RESOURCE>/$url_without_cf_params/g;

    $encoded_policy = url_safe_base64_encode($policy);
}

my $key = "";
if (exists $params{$KEY_PAIR_ID_PARAM}) {
    $key = $params{$KEY_PAIR_ID_PARAM};
} else {
    print STDERR "Missing $KEY_PAIR_ID_PARAM parameter\n";
    return 0;
}

my $policy = url_safe_base64_decode($encoded_policy);

my %ret = ();
$ret{"base_url"} = $base_url;
$ret{"policy"} = $policy;
$ret{"key"} = $key;

return \%ret;
```



```
    } else {
        return 0;
    }
}

# Print a decoded URL out
sub print_decoded_url {
    my $decoded = shift;

    print "Base URL: \n" . $decoded->{"base_url"} . "\n";
    print "Policy: \n" . $decoded->{"policy"} . "\n";
    print "Key: \n" . $decoded->{"key"} . "\n";
}

# Encode a string with base 64 encoding and replace some invalid URL characters
sub url_safe_base64_encode {
    my ($value) = @_;

    my $result = encode_base64($value);
    $result =~ tr|+="/|-_~|;

    return $result;
}

# Decode a string with base 64 encoding. URL-decode the string first
# followed by reversing any special character ("+="/) translation.
sub url_safe_base64_decode {
    my ($value) = @_;

    $value =~ s/%([0-9A-Fa-f]{2})/chr(hex($1))/eg;
    $value =~ tr|_~|+="/|;

    my $result = decode_base64($value);

    return $result;
}

# Create a private content URL
sub create_url {
    my ($path, $policy, $signature, $key_pair_id, $expires) = @_;

    my $result;
    my $separator = $path =~ /\?/ ? '&' : '?';
    if ($expires) {
```

```
    $result = "$path$separator$EXPIRES_PARAM=$expires&$SIGNATURE_PARAM=$signature&
$KEY_PAIR_ID_PARAM=$key_pair_id";
    } else {
        $result = "$path$separator$POLICY_PARAM=$policy&$SIGNATURE_PARAM=$signature&
$KEY_PAIR_ID_PARAM=$key_pair_id";
    }
    $result =~ s/\n//g;

    return $result;
}

# Sign a document with given private key file.
# The first argument is the document to sign
# The second argument is the name of the private key file
sub rsa_sha1_sign {
    my ($to_sign, $pvkFile) = @_;
    print "openssl sha1 -sign $pvkFile $to_sign\n";

    return write_to_program($pvkFile, $to_sign);
}

# Helper function to write data to a program
sub write_to_program {
    my ($keyfile, $data) = @_;
    unlink "temp_policy.dat" if (-e "temp_policy.dat");
    unlink "temp_sign.dat" if (-e "temp_sign.dat");

    write_file("temp_policy.dat", $data);

    system("openssl dgst -sha1 -sign \"\$keyfile\" -out temp_sign.dat temp_policy.dat");

    my $output = read_file("temp_sign.dat");

    return $output;
}

# Read a file into a string and return the string
sub read_file {
    my ($file) = @_;

    open(INFILE, "<$file") or die("Failed to open $file: $!");
    my $str = join('', <INFILE>);
    close INFILE;
}
```

```
    return $str;
}

sub is_url_valid {
    my ($url) = @_;

    # HTTP distributions start with http[s]:// and are the correct thing to sign
    if ($url =~ /^https?:\\\/\\\/) {
        return 1;
    } else {
        print STDERR "CloudFront requires absolute URLs for HTTP distributions\n";
        return 0;
    }
}

sub is_stream_valid {
    my ($stream) = @_;

    if ($stream =~ /^rtmp:\\\/\\\/ or $stream =~ /^\\\/?cfx\\\/st/) {
        print STDERR "Streaming distributions require that only the stream name is
signed.\n";
        print STDERR "The stream name is everything after, but not including, cfx/st/
\n";
        return 0;
    } else {
        return 1;
    }
}

# flash requires that the query parameters in the stream name are url
# encoded when passed in through javascript, etc. This sub handles the minimal
# required url encoding.
sub escape_url_for_webpage {
    my ($url) = @_;

    $url =~ s/\?/%3F/g;
    $url =~ s/=/%3D/g;
    $url =~ s/&%26/g;

    return $url;
}

1;
```

Criar uma assinatura de URL usando PHP

Qualquer servidor da web que execute PHP pode usar esse exemplo de código PHP para criar declarações de política e assinaturas para distribuições privadas do CloudFront. O exemplo completo cria uma página da web ativa com links de signed URL que reproduzem um streaming de vídeo usando streaming do CloudFront. É possível fazer download do exemplo completo em <https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/samples/demo-php.zip>.

Você também pode criar URLs assinados usando a classe `UrlSigner` no AWS SDK for PHP. Para obter mais informações, consulte [Classe UrlSigner](#) na Referência de API do AWS SDK for PHP.

Note

A criação de uma assinatura de URL é apenas uma parte do processo de fornecimento de conteúdo privado usando um signed URL. Para obter mais informações sobre o todo o processo, consulte [Usar URLs assinados](#).

Tópicos

- [Exemplo: assinatura RSA SHA-1](#)
- [Exemplo: criar uma política predefinida](#)
- [Exemplo: criar uma política personalizada](#)
- [Exemplo de código completo](#)

Exemplo: assinatura RSA SHA-1

No exemplo de código a seguir, a função `rsa_sha1_sign` adiciona hash e assina a declaração de política. Os argumentos necessários são uma declaração de política e a chave privada que corresponde a uma chave pública que está em um grupo de chaves confiável para sua distribuição. Em seguida, a função `url_safe_base64_encode` cria uma versão da assinatura segura para URL.

```
function rsa_sha1_sign($policy, $private_key_filename) {
    $signature = "";

    // load the private key
    $fp = fopen($private_key_filename, "r");
    $priv_key = fread($fp, 8192);
    fclose($fp);
    $pkeyid = openssl_get_privatekey($priv_key);
```

```

// compute signature
openssl_sign($policy, $signature, $pkeyid);

// free the key from memory
openssl_free_key($pkeyid);

return $signature;
}

function url_safe_base64_encode($value) {
    $encoded = base64_encode($value);
    // replace unsafe characters +, = and / with
    // the safe characters -, _ and ~
    return str_replace(
        array('+', '=', '/'),
        array('-', '_', '~'),
        $encoded);
}

```

Exemplo: criar uma política predefinida

O código de exemplo a seguir cria uma declaração de política padrão para a assinatura. Para obter mais informações sobre políticas padrão, consulte [Criar um URL assinado usando uma política predefinida](#).

Note

A variável `$expires` é um carimbo de data/hora que deve ser um número inteiro, não uma string.

```

function get_canned_policy_stream_name($video_path, $private_key_filename,
    $key_pair_id, $expires) {
    // this policy is well known by CloudFront, but you still need to sign it,
    // since it contains your parameters
    $canned_policy = '{"Statement":[{"Resource":"' . $video_path . '", "Condition":
{"DateLessThan":{"AWS:EpochTime":' . $expires . '}}]}';

    // sign the canned policy
    $signature = rsa_sha1_sign($canned_policy, $private_key_filename);
    // make the signature safe to be included in a url

```

```

$encoded_signature = url_safe_base64_encode($signature);

// combine the above into a stream name
$stream_name = create_stream_name($video_path, null, $encoded_signature,
$key_pair_id, $expires);
// url-encode the query string characters to work around a flash player bug
return encode_query_params($stream_name);
}

```

Exemplo: criar uma política personalizada

O código de exemplo a seguir cria uma declaração de política personalizada para a assinatura. Para obter mais informações sobre políticas personalizadas, consulte [Criar um URL assinado usando uma política personalizada](#).

```

function get_custom_policy_stream_name($video_path, $private_key_filename,
$key_pair_id, $policy) {
    // sign the policy
    $signature = rsa_sha1_sign($policy, $private_key_filename);
    // make the signature safe to be included in a url
    $encoded_signature = url_safe_base64_encode($signature);

    // combine the above into a stream name
    $stream_name = create_stream_name($video_path, $encoded_policy, $encoded_signature,
$key_pair_id, null);
    // url-encode the query string characters to work around a flash player bug
    return encode_query_params($stream_name);
}

```

Exemplo de código completo

O exemplo de código a seguir fornece uma demonstração completa de como criar signed URLs do CloudFront com PHP. É possível fazer download desse exemplo completo em <https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/samples/demo-php.zip>.

No exemplo a seguir, é possível modificar o elemento `$policy Condition` para permitir os intervalos de endereços do IPv4 e do IPv6. Por exemplo, consulte [Como usar endereços do IPv6 em políticas do IAM](#) no Guia do usuário do Amazon Simple Storage Service.

```

<?php

function rsa_sha1_sign($policy, $private_key_filename) {

```

```
$signature = "";

// load the private key
$fp = fopen($private_key_filename, "r");
$priv_key = fread($fp, 8192);
fclose($fp);
$pkeyid = openssl_get_privatekey($priv_key);

// compute signature
openssl_sign($policy, $signature, $pkeyid);

// free the key from memory
openssl_free_key($pkeyid);

return $signature;
}

function url_safe_base64_encode($value) {
    $encoded = base64_encode($value);
    // replace unsafe characters +, = and / with the safe characters -, _ and ~
    return str_replace(
        array('+', '=', '/'),
        array('-', '_', '~'),
        $encoded);
}

function create_stream_name($stream, $policy, $signature, $key_pair_id, $expires) {
    $result = $stream;
    // if the stream already contains query parameters, attach the new query parameters
    to the end
    // otherwise, add the query parameters
    $separator = strpos($stream, '?') == FALSE ? '?' : '&';
    // the presence of an expires time means we're using a canned policy
    if($expires) {
        $result .= $path . $separator . "Expires=" . $expires . "&Signature=" .
        $signature . "&Key-Pair-Id=" . $key_pair_id;
    }
    // not using a canned policy, include the policy itself in the stream name
    else {
        $result .= $path . $separator . "Policy=" . $policy . "&Signature=" .
        $signature . "&Key-Pair-Id=" . $key_pair_id;
    }

    // new lines would break us, so remove them
```

```
    return str_replace('\n', '', $result);
}

function encode_query_params($stream_name) {
    // Adobe Flash Player has trouble with query parameters being passed into it,
    // so replace the bad characters with their URL-encoded forms
    return str_replace(
        array('?', '=', '&'),
        array('%3F', '%3D', '%26'),
        $stream_name);
}

function get_canned_policy_stream_name($video_path, $private_key_filename,
    $key_pair_id, $expires) {
    // this policy is well known by CloudFront, but you still need to sign it, since it
    // contains your parameters
    $canned_policy = '{"Statement":[{"Resource":"' . $video_path . '", "Condition":
{"DateLessThan":{"AWS:EpochTime":'. $expires . '}}]}]';
    // the policy contains characters that cannot be part of a URL, so we base64 encode
    // it
    $encoded_policy = url_safe_base64_encode($canned_policy);
    // sign the original policy, not the encoded version
    $signature = rsa_sha1_sign($canned_policy, $private_key_filename);
    // make the signature safe to be included in a URL
    $encoded_signature = url_safe_base64_encode($signature);

    // combine the above into a stream name
    $stream_name = create_stream_name($video_path, null, $encoded_signature,
    $key_pair_id, $expires);
    // URL-encode the query string characters to support Flash Player
    return encode_query_params($stream_name);
}

function get_custom_policy_stream_name($video_path, $private_key_filename,
    $key_pair_id, $policy) {
    // the policy contains characters that cannot be part of a URL, so we base64 encode
    // it
    $encoded_policy = url_safe_base64_encode($policy);
    // sign the original policy, not the encoded version
    $signature = rsa_sha1_sign($policy, $private_key_filename);
    // make the signature safe to be included in a URL
    $encoded_signature = url_safe_base64_encode($signature);

    // combine the above into a stream name
```



```
$stream_name = create_stream_name($video_path, $encoded_policy, $encoded_signature,
$key_pair_id, null);
// URL-encode the query string characters to support Flash Player
return encode_query_params($stream_name);
}

// Path to your private key. Be very careful that this file is not accessible
// from the web!

$private_key_filename = '/home/test/secure/example-priv-key.pem';
$key_pair_id = 'K2JCJMDEHXQW5F';

$video_path = 'example.mp4';

$expires = time() + 300; // 5 min from now
$canned_policy_stream_name = get_canned_policy_stream_name($video_path,
$private_key_filename, $key_pair_id, $expires);

$client_ip = $_SERVER['REMOTE_ADDR'];
$policy =
'{'.
  '"Statement":['.
    '{'.
      '"Resource":' . $video_path . ',' .
      '"Condition":{'.
        '"IpAddress":{"AWS:SourceIp":"' . $client_ip . '/32"},'.
        '"DateLessThan":{"AWS:EpochTime":"' . $expires . '}'.
      '}'.
    '}'.
  ']' .
'}';

$custom_policy_stream_name = get_custom_policy_stream_name($video_path,
$private_key_filename, $key_pair_id, $policy);

?>

<html>

<head>
  <title>CloudFront</title>
<script type='text/javascript' src='https://example.cloudfront.net/player/
swfobject.js'></script>
</head>
```

```
<body>
  <h1>Amazon CloudFront</h1>
  <h2>Canned Policy</h2>
  <h3>Expires at <? = gmdate('Y-m-d H:i:s T', $expires) ?></h3>
  <br />

  <div id='canned'>The canned policy video will be here</div>

  <h2>Custom Policy</h2>
  <h3>Expires at <? = gmdate('Y-m-d H:i:s T', $expires) ?> only viewable by IP <? =
$client_ip ?></h3>
  <div id='custom'>The custom policy video will be here</div>

  <!-- ***** Have to update the player.swf path to a real JWPlayer instance.
  The fake one means that external people cannot watch the video right now -->
  <script type='text/javascript'>
var so_canned = new SWFObject('https://files.example.com/
player.swf', 'mpl', '640', '360', '9');
so_canned.addParam('allowfullscreen', 'true');
so_canned.addParam('allowscriptaccess', 'always');
so_canned.addParam('wmode', 'opaque');
so_canned.addVariable('file', '<? = $canned_policy_stream_name ?>');
so_canned.addVariable('streamer', 'rtmp://example.cloudfront.net/cfx/st');
so_canned.write('canned');

var so_custom = new SWFObject('https://files.example.com/
player.swf', 'mpl', '640', '360', '9');
so_custom.addParam('allowfullscreen', 'true');
so_custom.addParam('allowscriptaccess', 'always');
so_custom.addParam('wmode', 'opaque');
so_custom.addVariable('file', '<? = $custom_policy_stream_name ?>');
so_custom.addVariable('streamer', 'rtmp://example.cloudfront.net/cfx/st');
so_custom.write('custom');
  </script>
</body>

</html>
```

Consulte também:

- [Criar uma assinatura de URL usando Perl](#)
- [Criar uma assinatura de URL usando C# e o .NET Framework](#)

- [Criar uma assinatura de URL usando Java](#)

Criar uma assinatura de URL usando C# e o .NET Framework

Os exemplos de C# desta seção implementam um exemplo de aplicação que demonstra como criar assinaturas para distribuições privadas do CloudFront usando declarações de política padrão e personalizada. Alguns dos exemplos são as funções de utilitário com base no [AWS SDK for .NET](#) que podem ser úteis em aplicações .NET.

Você também pode criar URLs assinados e cookies com o uso do AWS SDK for .NET. Na Referência de API do AWS SDK for .NET, consulte os seguintes tópicos:

- URLs assinados: [AmazonCloudFrontUrlSigner](#)
- Cookies assinados: [AmazonCloudFrontCookieSigner](#)

Para fazer download do código, acesse [Código de assinatura em C#](#).

Note

A criação de uma assinatura de URL é apenas uma parte do processo de fornecimento de conteúdo privado usando um signed URL. Para obter mais informações sobre o todo o processo, consulte [Usar URLs assinados](#). Para ter mais informações sobre como usar cookies assinados, consulte [Usar cookies assinados](#).

Usar uma chave RSA no .NET Framework

Para usar uma chave RSA no .NET Framework, converta o arquivo .pem da AWS fornecido no formato XML usado pelo .NET Framework.

Após a conversão, o arquivo de chave privada do RSA ficará no seguinte formato:

Example : chave privada do RSA no formato XML do .NET Framework

```
<RSAKeyValue>
  <Modulus>
    w05IvYCP5UcoCKDo1dcspoMehWBZcyfs9QEzGi60e5y+ewGr1oW+vB2GPB
    ANBiVPcUHTFWhwaIBd3oglmF01GQ1jP/j0fmXHUK2kUUnLnJp+o0BL2NiuFtqcW6h/L51IpD8Yq+NRHg
    Ty4zDsy12880MvXv88yEFURckqEXAMPLE=
```

```

</Modulus>
<Exponent>AQAB</Exponent>
<P>
  5bmKDaTz
  npENGVqz4Cea8XPH+sxt+2VaAwYnsarVUoSBeVt8WL1oVuZGG9IZYmH5KteXEu7fZveYd9UEXAMPLE==
</P>
<Q>
  1v9l/WN1a1N3r0K4VGoCokx7kR2SyTMSbZgF9IWJN0ugR/WZw7HTnqip03c9dy1Ms9pUKwUF4
  6d7049EXAMPLE==
</Q>
<DP>
  RgrSKuLWXMyBH+/l1Dx/I4tXuAJIrlPyo+Vmi0c7b5NzHptkSHEPFR9s1
  0K0VqjknclqCJ3Ig860MEtEXAMPLE==
</DP>
<DQ>
  pjPjvSFw+RoaTu0pgCA/jwW/FGyfn6iim1RFbkT4
  z49DZb2IM885f3vf35eLTaEYRYUHqgZtChNEV0TEXAMPLE==
</DQ>
<InverseQ>
  nkV0JTg5QtGNgWb9i
  cVtzrL/1pFE0HbJXwEJdU99N+7sMK+1066DL/HSBUCD63qD4USpnf0myc24in0EXAMPLE==</InverseQ>
<D>
  Bc7mp7XYHyNuPZxChjWNJZiQ+A73gm0ASDv6At7F8Vi9r0xU1Qe/v0AQS3ycN8Q1yR4XMbzMLYk
  3yJxFDXo4ZKQt0GzLGteCU2srANiLv26/imXA8FvidZftTAtLviWQZBVPTeYIA69ATUYPEq0a5u5wjGy
  U0ij90WyuEXAMPLE=
</D>
</RSAKeyValue>

```

Método de assinatura de política padrão em C#

O seguinte código C# cria um URL assinado que usa uma política padrão executando as seguintes etapas:

- Cria uma declaração de política.
- Adiciona hash à declaração de política usando SHA1 e assina o resultado usando RSA e a chave privada com a chave pública correspondente que está em um grupo de chaves confiável.
- Codifica em base64 a declaração de política assinada e com hash e substitui os caracteres especiais para tornar a string segura para ser usada como parâmetro de solicitação de URL.
- Concatena os valores.

Para obter a implementação completa, consulte o exemplo em [Código de assinatura em C#](#).

Note

O keyId é exibido quando você faz upload de uma chave pública no CloudFront. Para ter mais informações, consulte



[&Key-Pair-Id.](#)

Example : método de assinatura de política predefinida em C#

```
public static string ToUrlSafeBase64String(byte[] bytes)
{
    return System.Convert.ToBase64String(bytes)
        .Replace('+', '-')
        .Replace('=', '_')
        .Replace('/', '~');
}

public static string CreateCannedPrivateURL(string urlString,
    string durationUnits, string durationNumber, string pathToPolicyStmnt,
    string pathToPrivateKey, string keyId)
{
    // args[] 0-thisMethod, 1-resourceUrl, 2-seconds-minutes-hours-days
    // to expiration, 3-numberOfPreviousUnits, 4-pathToPolicyStmnt,
    // 5-pathToPrivateKey, 6-keyId

    TimeSpan timeSpanInterval = GetDuration(durationUnits, durationNumber);

    // Create the policy statement.
    string strPolicy = CreatePolicyStatement(pathToPolicyStmnt,
        urlString,
        DateTime.Now,
        DateTime.Now.Add(timeSpanInterval),
        "0.0.0.0/0");
    if ("Error!" == strPolicy) return "Invalid time frame." +
        "Start time cannot be greater than end time.";

    // Copy the expiration time defined by policy statement.
    string strExpiration = CopyExpirationTimeFromPolicy(strPolicy);

    // Read the policy into a byte buffer.
    byte[] bufferPolicy = Encoding.ASCII.GetBytes(strPolicy);
}
```

```
// Initialize the SHA1CryptoServiceProvider object and hash the policy data.
using (SHA1CryptoServiceProvider
    cryptoSHA1 = new SHA1CryptoServiceProvider())
{
    bufferPolicy = cryptoSHA1.ComputeHash(bufferPolicy);

    // Initialize the RSACryptoServiceProvider object.
    RSACryptoServiceProvider providerRSA = new RSACryptoServiceProvider();
    XmlDocument xmlPrivateKey = new XmlDocument();

    // Load your private key, which you created by converting your
    // .pem file to the XML format that the .NET framework uses.
    // Several tools are available.
    xmlPrivateKey.Load(pathToPrivateKey);

    // Format the RSACryptoServiceProvider providerRSA and
    // create the signature.
    providerRSA.FromXmlString(xmlPrivateKey.InnerXml);
    RSAPKCS1SignatureFormatter rsaFormatter =
        new RSAPKCS1SignatureFormatter(providerRSA);
    rsaFormatter.SetHashAlgorithm("SHA1");
    byte[] signedPolicyHash = rsaFormatter.CreateSignature(bufferPolicy);

    // Convert the signed policy to URL-safe base64 encoding and
    // replace unsafe characters + = / with the safe characters - _ ~
    string strSignedPolicy = ToUrlSafeBase64String(signedPolicyHash);

    // Concatenate the URL, the timestamp, the signature,
    // and the key pair ID to form the signed URL.
    return urlString +
        "?Expires=" +
        strExpiration +
        "&Signature=" +
        strSignedPolicy +
        "&Key-Pair-Id=" +
        keyId;
}
}
```

Método de assinatura de política personalizada em C#

O seguinte código C# cria um URL assinado que usa uma política personalizada executando as seguintes etapas:

1. Cria uma declaração de política.
2. Codifica a declaração de política em base64 e substitui os caracteres especiais para tornar a string segura para ser usada como parâmetro de solicitação de URL.
3. Adiciona hash a declaração de política usando SHA1 e criptografa o resultado usando RSA e a chave privada com a chave pública correspondente que está em um grupo de chaves confiável.
4. Codifica em base64 a declaração de política com hash e substitui os caracteres especiais para tornar a string segura para ser usada como parâmetro de solicitação de URL.
5. Concatena os valores.

Para obter a implementação completa, consulte o exemplo em [Código de assinatura em C#](#).

Note

O keyId é exibido quando você faz upload de uma chave pública no CloudFront. Para ter mais informações, consulte



[&Key-Pair-Id](#).

Example : método de assinatura de política personalizada em C#

```
public static string ToUrlSafeBase64String(byte[] bytes)
{
    return System.Convert.ToBase64String(bytes)
        .Replace('+', '-')
        .Replace('=', '_')
        .Replace('/', '~');
}

public static string CreateCustomPrivateURL(string urlString,
    string durationUnits, string durationNumber, string startIntervalFromNow,
    string ipAddress, string pathToPolicyStmnt, string pathToPrivateKey,
    string keyId)
{
```

```
// args[] 0-thisMethod, 1-resourceUrl, 2-seconds-minutes-hours-days
// to expiration, 3-numberOfPreviousUnits, 4-starttimeFromNow,
// 5-ip_address, 6-pathToPolicyStmnt, 7-pathToPrivateKey, 8-keyId

TimeSpan timeSpanInterval = GetDuration(durationUnits, durationNumber);
TimeSpan timeSpanToStart = GetDurationByUnits(durationUnits,
    startIntervalFromNow);
if (null == timeSpanToStart)
    return "Invalid duration units." +
        "Valid options: seconds, minutes, hours, or days";

string strPolicy = CreatePolicyStatement(
    pathToPolicyStmnt, urlString, DateTime.Now.Add(timeSpanToStart),
    DateTime.Now.Add(timeSpanInterval), ipaddress);

// Read the policy into a byte buffer.
byte[] bufferPolicy = Encoding.ASCII.GetBytes(strPolicy);

// Convert the policy statement to URL-safe base64 encoding and
// replace unsafe characters + = / with the safe characters - _ ~

string urlSafePolicy = ToUrlSafeBase64String(bufferPolicy);

// Initialize the SHA1CryptoServiceProvider object and hash the policy data.
byte[] bufferPolicyHash;
using (SHA1CryptoServiceProvider cryptoSHA1 =
    new SHA1CryptoServiceProvider())
{
    bufferPolicyHash = cryptoSHA1.ComputeHash(bufferPolicy);

    // Initialize the RSACryptoServiceProvider object.
    RSACryptoServiceProvider providerRSA = new RSACryptoServiceProvider();
    XmlDocument xmlPrivateKey = new XmlDocument();

    // Load your private key, which you created by converting your
    // .pem file to the XML format that the .NET framework uses.
    // Several tools are available.
    xmlPrivateKey.Load(pathToPrivateKey);

    // Format the RSACryptoServiceProvider providerRSA
    // and create the signature.
    providerRSA.FromXmlString(xmlPrivateKey.InnerXml);
    RSAPKCS1SignatureFormatter RSAFormatter =
        new RSAPKCS1SignatureFormatter(providerRSA);
```



```

    RSAFormatter.SetHashAlgorithm("SHA1");
    byte[] signedHash = RSAFormatter.CreateSignature(bufferPolicyHash);

    // Convert the signed policy to URL-safe base64 encoding and
    // replace unsafe characters + = / with the safe characters - _ ~
    string strSignedPolicy = ToUrlSafeBase64String(signedHash);

    return urlString +
        "?Policy=" +
        urlSafePolicy +
        "&Signature=" +
        strSignedPolicy +
        "&Key-Pair-Id=" +
        keyId;
}
}

```

Métodos utilitários para geração de assinaturas

Os métodos a seguir obtêm a declaração de política de um arquivo e analisam os intervalos de tempo para a geração de assinatura.

Example : métodos utilitários para geração de assinaturas

```

public static string CreatePolicyStatement(string policyStmnt,
    string resourceUrl,
    DateTime startTime,
    DateTime endTime,
    string ipAddress)

{
    // Create the policy statement.
    FileStream streamPolicy = new FileStream(policyStmnt, FileMode.Open,
    FileAccess.Read);
    using (StreamReader reader = new StreamReader(streamPolicy))
    {
        string strPolicy = reader.ReadToEnd();

        TimeSpan startTimeSpanFromNow = (startTime - DateTime.Now);
        TimeSpan endTimeSpanFromNow = (endTime - DateTime.Now);
        TimeSpan intervalStart =
            (DateTime.UtcNow.Add(startTimeSpanFromNow)) -
            new DateTime(1970, 1, 1, 0, 0, 0, DateTimeKind.Utc);
        TimeSpan intervalEnd =

```

```
(DateTime.UtcNow.Add(endTimeSpanFromNow)) -
    new DateTime(1970, 1, 1, 0, 0, 0, DateTimeKind.Utc);

int startTimestamp = (int)intervalStart.TotalSeconds; // START_TIME
int endTimestamp = (int)intervalEnd.TotalSeconds; // END_TIME

if (startTimestamp > endTimestamp)
    return "Error!";

// Replace variables in the policy statement.
strPolicy = strPolicy.Replace("RESOURCE", resourceUrl);
strPolicy = strPolicy.Replace("START_TIME", startTimestamp.ToString());
strPolicy = strPolicy.Replace("END_TIME", endTimestamp.ToString());
strPolicy = strPolicy.Replace("IP_ADDRESS", ipAddress);
strPolicy = strPolicy.Replace("EXPIRES", endTimestamp.ToString());
return strPolicy;
}
}

public static TimeSpan GetDuration(string units, string numUnits)
{
    TimeSpan timeSpanInterval = new TimeSpan();
    switch (units)
    {
        case "seconds":
            timeSpanInterval = new TimeSpan(0, 0, 0, int.Parse(numUnits));
            break;
        case "minutes":
            timeSpanInterval = new TimeSpan(0, 0, int.Parse(numUnits), 0);
            break;
        case "hours":
            timeSpanInterval = new TimeSpan(0, int.Parse(numUnits), 0, 0);
            break;
        case "days":
            timeSpanInterval = new TimeSpan(int.Parse(numUnits), 0, 0, 0);
            break;
        default:
            Console.WriteLine("Invalid time units;" +
                "use seconds, minutes, hours, or days");
            break;
    }
    return timeSpanInterval;
}
```

```
private static TimeSpan GetDurationByUnits(string durationUnits,
    string startIntervalFromNow)
{
    switch (durationUnits)
    {
        case "seconds":
            return new TimeSpan(0, 0, int.Parse(startIntervalFromNow));
        case "minutes":
            return new TimeSpan(0, int.Parse(startIntervalFromNow), 0);
        case "hours":
            return new TimeSpan(int.Parse(startIntervalFromNow), 0, 0);
        case "days":
            return new TimeSpan(int.Parse(startIntervalFromNow), 0, 0, 0);
        default:
            return new TimeSpan(0, 0, 0, 0);
    }
}

public static string CopyExpirationTimeFromPolicy(string policyStatement)
{
    int startExpiration = policyStatement.IndexOf("EpochTime");
    string strExpirationRough = policyStatement.Substring(startExpiration +
        "EpochTime".Length);
    char[] digits = { '0', '1', '2', '3', '4', '5', '6', '7', '8', '9' };

    List<char> listDigits = new List<char>(digits);
    StringBuilder buildExpiration = new StringBuilder(20);

    foreach (char c in strExpirationRough)
    {
        if (listDigits.Contains(c))
            buildExpiration.Append(c);
    }
    return buildExpiration.ToString();
}
```

Consulte também

- [Criar uma assinatura de URL usando Perl](#)
- [Criar uma assinatura de URL usando PHP](#)
- [Criar uma assinatura de URL usando Java](#)

Criar uma assinatura de URL usando Java

Além do exemplo de código a seguir, você pode usar [a classe de utilitário CloudFrontUrlSigner no AWS SDK for Java \(versão 1\)](#) para criar [signed URLs do CloudFront](#).

Para ver mais exemplos, consulte [Crie URLs e cookies assinados usando um SDK da AWS](#) na Biblioteca de códigos de exemplos de códigos do SDK da AWS.

Note

Criar um signed URL é somente uma parte do processo de [fornecimento de conteúdo privado com o CloudFront](#). Para obter mais informações sobre o todo o processo, consulte [Usar URLs assinados](#).

O exemplo a seguir mostra como criar um signed URL do CloudFront.

Exemplo Política Java e métodos de criptografia de assinatura

```
package org.example;

import java.time.Instant;
import java.time.temporal.ChronoUnit;
import software.amazon.awssdk.services.cloudfront.CloudFrontUtilities;
import software.amazon.awssdk.services.cloudfront.model.CannedSignerRequest;
import software.amazon.awssdk.services.cloudfront.url.SignedUrl;

public class Main {

    public static void main(String[] args) throws Exception {
        CloudFrontUtilities cloudFrontUtilities = CloudFrontUtilities.create();
        Instant expirationDate = Instant.now().plus(7, ChronoUnit.DAYS);
        String resourceUrl = "https://a1b2c3d4e5f6g7.cloudfront.net";
        String keyPairId = "K1UA3WV15I7JSD";
        CannedSignerRequest cannedRequest = CannedSignerRequest.builder()
            .resourceUrl(resourceUrl)
            .privateKey(new java.io.File("/path/to/private_key.pem").toPath())
            .keyPairId(keyPairId)
            .expirationDate(expirationDate)
            .build();
        SignedUrl signedUrl =
            cloudFrontUtilities.getSignedUrlWithCannedPolicy(cannedRequest);
    }
}
```

```
String url = signedUrl.url();
System.out.println(url);

    }
}
```

Consulte também:

- [Criar uma assinatura de URL usando Perl](#)
- [Criar uma assinatura de URL usando PHP](#)
- [Criar uma assinatura de URL usando C# e o .NET Framework](#)

Restringir o acesso a uma origem da AWS

É possível configurar o CloudFront e algumas origens da AWS de uma forma que ofereça os seguintes benefícios:

- Restringir o acesso à origem da AWS para que ela não seja acessível ao público
- Garantir que os visualizadores (usuários) possam acessar o conteúdo na origem da AWS somente por meio da distribuição especificada do CloudFront, ou seja, impedir que eles acessem o conteúdo diretamente do bucket ou por meio de uma distribuição não intencional do CloudFront

Para fazer isso, configure o CloudFront para enviar solicitações autenticadas para a origem da AWS e configure a origem da AWS para permitir acesso às solicitações autenticadas do CloudFront. Para ter mais informações, consulte os tópicos a seguir para obter os tipos compatíveis de origens da AWS.

Tópicos

- [Restringir o acesso a uma origem da AWS Elemental MediaPackage v2](#)
- [Restringir o acesso a uma origem da AWS Elemental MediaStore](#)
- [Restringir o acesso a uma origem do URL de função do AWS Lambda](#)
- [Restringir o acesso a uma origem do Amazon Simple Storage Service](#)

Restringir o acesso a uma origem da AWS Elemental MediaPackage v2

O CloudFront fornece controle de acesso à origem (OAC) para restringir o acesso a uma origem do MediaPackage v2.

Note

O OAC do CloudFront só é compatível com o MediaPackage v2. O MediaPackage v1 não é compatível.

Tópicos

- [Criar um novo OAC](#)
- [Configurações avançadas para controle de acesso à origem](#)

Criar um novo OAC

Conclua as etapas descritas nos tópicos a seguir para configurar um novo OAC no CloudFront.

Tópicos

- [Pré-requisitos](#)
- [Conceder permissão de controle de acesso à origem para acessar a origem do MediaPackage v2](#)
- [Criar o OAC](#)

Pré-requisitos

Antes de criar e configurar o controle de acesso à origem, você deve ter uma distribuição do CloudFront com uma origem do MediaPackage v2. Para ter mais informações, consulte [Usar um contêiner do MediaStore ou um canal do MediaPackage](#).

Conceder permissão de controle de acesso à origem para acessar a origem do MediaPackage v2

Antes de criar um controle de acesso à origem ou configurá-lo em uma distribuição do CloudFront, o OAC deve ter permissão para acessar a origem do MediaPackage v2. Faça isso depois de criar uma distribuição do CloudFront, mas antes de adicionar o OAC à origem do MediaPackage v2 na configuração de distribuição.

Para conceder permissão ao OAC para acessar a origem do MediaPackage v2, use uma política do IAM para possibilitar que a entidade principal de serviço do CloudFront (`cloudfront.amazonaws.com`) acesse a origem. O elemento `Condition` na política permite que o CloudFront acesse a origem do MediaPackage v2 somente quando a solicitação for em nome da distribuição do CloudFront que contém a origem do MediaPackage v2.

Exemplo : política do IAM que permite acesso somente leitura a uma distribuição do CloudFront

A política a seguir permite que a distribuição do CloudFront (`E1PDK09ESKHJWT`) acesse a origem do MediaPackage v2. A origem é o ARN especificado para o elemento `Resource`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCloudFrontServicePrincipal",
      "Effect": "Allow",
      "Principal": {"Service": "cloudfront.amazonaws.com"},
      "Action": "mediapackagev2:GetObject",
      "Resource": "arn:aws:mediapackagev2:us-
east-1:123456789012:channelGroup/channel-group-name/channel/channel-name/
originEndpoint/origin_endpoint_name",
      "Condition": {
        "StringEquals": {"AWS:SourceArn":
"arn:aws:cloudfront::123456789012:distribution/E1PDK09ESKHJWT"}
      }
    }
  ]
}
```

Note

Se você criar uma distribuição que não tenha permissão para sua origem do MediaPackage v2, é possível escolher Copy policy (Copiar política) no console do CloudFront e, depois, escolher Update endpoint permissions (Atualizar permissões de endpoint). Em seguida, é possível anexar a permissão copiada ao endpoint. Para ter mais informações, consulte [Endpoint policy fields](#) (Campos de política de endpoint) no Guia do usuário do AWS Elemental MediaPackage.

Criar o OAC

Para criar um OAC, é possível usar o AWS Management Console, o AWS CloudFormation, a AWS CLI ou a API do CloudFront.

Console

Para criar um OAC

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Origem access (Acesso à origem).
3. Selecione Create control setting (Criar configuração de controle).
4. No formulário Create new OAC (Criar novo OAC), faça o seguinte:
 - a. Informe um Name (Nome) e (opcionalmente) uma Description (Descrição) para o OAC.
 - b. Em Signing behavior (Comportamento de assinatura), recomendamos que deixe a configuração padrão (Sign requests (recommended) [Solicitações de assinatura (recomendado)]). Para ter mais informações, consulte [the section called “Configurações avançadas para controle de acesso à origem”](#).
5. Em Origin type (Tipo de origem), selecione MediaPackage v2.
6. Escolha Criar.

Tip

Depois que criar OAC, anote o Name (Nome). Você precisará dele no procedimento a seguir.

Como adicionar um OAC a uma origem do MediaPackage v2 em uma distribuição

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha uma distribuição com uma origem do MediaPackage v2 à qual você deseja adicionar o OAC e, depois, selecione a guia Origins (Origens).
3. Selecione a origem do MediaPackage v2 à qual você deseja adicionar o OAC e, depois, Edit (Editar).
4. Selecione HTTPS only (Somente HTTPS) para o Protocol (Protocolo) de sua origem.

5. No menu suspenso Origin access control (Controle de acesso à origem), selecione o nome do OAC que você deseja usar.
6. Escolha Salvar alterações.

A distribuição começa a ser implantada em todos os locais da borda do CloudFront. Quando um local da borda recebe a nova configuração, ele assina todas as solicitações enviadas à origem do MediaPackage v2.

CloudFormation

Para criar um OAC com o AWS CloudFormation, use o tipo de recurso `AWS::CloudFront::OriginAccessControl`. O exemplo a seguir mostra a sintaxe do modelo AWS CloudFormation no formato YAML, para criar um OAC.

```
Type: AWS::CloudFront::OriginAccessControl
Properties:
  OriginAccessControlConfig:
    Description: An optional description for the origin access control
    Name: ExampleOAC
    OriginAccessControlOriginType: mediapackagev2
    SigningBehavior: always
    SigningProtocol: sigv4
```

Para obter mais informações, consulte [AWS::CloudFront::OriginAccessControl](#) no Guia do usuário do AWS CloudFormation.

CLI

Para criar um controle de acesso à origem com a AWS Command Line Interface (AWS CLI), use o comando `aws cloudfront create-origin-access-control`. É possível usar um arquivo de entrada para fornecer os parâmetros de entrada do comando, em vez de especificar cada parâmetro individual como entrada na linha de comando.

Como criar um controle de acesso à origem (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `origin-access-control.yaml`. Esse arquivo contém todos os parâmetros de entrada para o comando `create-origin-access-control`.

```
aws cloudfront create-origin-access-control --generate-cli-skeleton yml-input >
origin-access-control.yml
```

2. Abra o arquivo `origin-access-control.yml` que você acabou de criar. Edite o arquivo para adicionar um nome para o OAC, uma descrição (opcional) e alterar `SigningBehavior` para `always`. Salve o arquivo.

Para obter mais informações sobre outras configurações de OAC, consulte [the section called “Configurações avançadas para controle de acesso à origem”](#).

3. Use o comando a seguir para criar o controle de acesso à origem usando parâmetros de entrada do arquivo `origin-access-control.yml`.

```
aws cloudfront create-origin-access-control --cli-input-yml file://origin-
access-control.yml
```

Anote o valor do `Id` na saída do comando. Você precisa dele para adicionar o OAC a uma origem do `MediaPackage v2` em uma distribuição do CloudFront.

Como anexar um OAC a uma origem do `MediaPackage v2` em uma distribuição existente (CLI com arquivo de entrada)

1. Use o comando a seguir para salvar a configuração da distribuição do CloudFront à qual você deseja adicionar o OAC. A distribuição deve ter uma origem do `MediaPackage v2`.

```
aws cloudfront get-distribution-config --id <CloudFront distribution ID> --
output yml > dist-config.yml
```

2. Abra o arquivo chamado `dist-config.yml` que você acabou de criar. Edite o arquivo fazendo as seguintes alterações:
 - No objeto `Origins`, adicione o ID do OAC ao campo chamado `OriginAccessControlId`.
 - Remova o valor do campo chamado `OriginAccessIdentity`, se houver.
 - Renomeie o campo `ETag` para `IfMatch`, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

3. Use o comando a seguir para atualizar a distribuição para usar o controle de acesso à origem.

```
aws cloudfront update-distribution --id <CloudFront distribution ID> --cli-input-yaml file://dist-config.yaml
```

A distribuição começa a ser implantada em todos os locais da borda do CloudFront. Quando um local da borda recebe a nova configuração, ele assina todas as solicitações enviadas à origem do MediaPackage v2.

API

Para criar um OAC com a API do CloudFront, use [CreateOriginAccessControl](#). Para obter mais informações sobre os campos especificados nessa chamada de API, consulte a documentação de referência de API do seu AWS SDK ou de outro cliente de API.

Assim que criar um controle de acesso à origem, é possível anexá-lo a uma origem do MediaPackage v2 em uma distribuição usando uma das seguintes chamadas de API:

- Para anexá-lo a uma distribuição existente, use [UpdateDistribution](#).
- Para anexá-lo a uma nova distribuição, use [CreateDistribution](#).

Para as duas chamadas de API, forneça o ID de OAC no campo `OriginAccessControlId`, dentro de uma origem. Para mais informações sobre os outros campos especificados nessas chamadas de API, consulte [Referência de configurações da distribuição](#) e a documentação de referência da API do AWS SDK ou de outro cliente de API.

Configurações avançadas para controle de acesso à origem

O atributo de OAC do CloudFront inclui configurações avançadas destinadas somente a casos de uso específicos. Use as configurações recomendadas, a menos que você precise usar as configurações avançadas para uma necessidade específica.

O OAC contém uma configuração chamada `Signing behavior` (Comportamento de assinatura) (no console) ou `SigningBehavior` (na API, na CLI e no AWS CloudFormation). Essa configuração fornece as seguintes opções:

Always sign origin requests (recommended setting) [Sempre assinar solicitações de origem (configuração recomendada)]

Recomendamos usar essa configuração, chamada Sign requests (recommended) [Assinar solicitações (recomendado)] no console, ou `always` na API, na CLI e no AWS CloudFormation. Com essa configuração, o CloudFront sempre assina todas as solicitações enviadas à origem do MediaPackage v2.

Never sign origin requests (Nunca assinar solicitações de origem)

Essa configuração é chamada Do not sign requests (Não assinar solicitações) no console ou `never` na API, na CLI e no AWS CloudFormation. Use essa configuração para desativar o OAC para todas as origens em todas as distribuições que usam esse OAC. Isso pode economizar tempo e esforço em comparação com a remoção individual de um OAC de todas as origens e distribuições que o usam. Com essa configuração, o CloudFront não assina nenhuma solicitação enviada à origem do MediaPackage v2.

Warning

Para usar essa configuração, a origem do MediaPackage v2 deve estar acessível de forma pública. Se você usar essa configuração com uma origem do MediaPackage v2 que não esteja acessível de forma pública, o CloudFront não poderá acessar a origem. A origem do MediaPackage v2 retorna erros ao CloudFront e o CloudFront transmite esses erros aos visualizadores. Para ter mais informações, consulte o exemplo de política do MediaPackage v2 para [Políticas e permissões no MediaPackage](#) no Guia do usuário do AWS Elemental MediaPackage.

Não substituir o cabeçalho **Authorization** do visualizador (cliente)

Essa configuração é chamada Do not override authorization header (Não substituir o cabeçalho de autorização) no console ou `no-override` na API, na CLI e no AWS CloudFormation. Use essa configuração quando quiser que o CloudFront assine solicitações de origem somente quando a solicitação do visualizador correspondente não incluir um cabeçalho `Authorization`. Com essa configuração, o CloudFront transmite o cabeçalho `Authorization` da solicitação do visualizador quando houver, mas assina a solicitação de origem (adicionando seu próprio cabeçalho `Authorization`) quando a solicitação do visualizador não inclui um cabeçalho `Authorization`.

⚠ Warning

Para transmitir o cabeçalho `Authorization` da solicitação do visualizador, você deve adicionar o cabeçalho `Authorization` a uma [política de cache](#) para todos os comportamentos de cache que usam origens do MediaPackage v2 associadas a esse controle de acesso à origem.

Restringir o acesso a uma origem da AWS Elemental MediaStore

O CloudFront fornece controle de acesso à origem (OAC) para restringir o acesso a uma origem do AWS Elemental MediaStore.

Tópicos

- [Criar um controle de acesso à origem](#)
- [Configurações avançadas para controle de acesso à origem](#)

Criar um controle de acesso à origem

Conclua as etapas descritas nos tópicos a seguir para configurar um novo controle de acesso à origem no CloudFront.

Tópicos

- [Pré-requisitos](#)
- [Conceder permissão de controle de acesso à origem para acessar a origem do MediaStore](#)
- [Criar o controle de acesso à origem](#)

Pré-requisitos

Antes de criar e configurar o controle de acesso à origem, você deve ter uma distribuição do CloudFront com uma origem do MediaStore.

Conceder permissão de controle de acesso à origem para acessar a origem do MediaStore

Antes de criar um controle de acesso à origem ou configurá-lo em uma distribuição do CloudFront, o OAC deve ter permissão para acessar a origem do MediaStore. Faça isso depois de criar uma

distribuição do CloudFront, mas antes de adicionar o OAC à origem do MediaStore na configuração de distribuição.

Para conceder permissão ao OAC para acessar a origem do MediaStore, use uma política de contêiner do MediaStore para possibilitar que a entidade principal de serviço do CloudFront (`cloudfront.amazonaws.com`) acesse a origem. Use um elemento `Condition` na política para permitir que o CloudFront acesse o contêiner do MediaStore somente quando a solicitação for em nome da distribuição do CloudFront que contém a origem do MediaStore.

Veja a seguir exemplos de políticas de contêiner do MediaStore que permitem que um OAC do CloudFront acesse uma origem do MediaStore.

Example Política de contêiner do MediaStore que permite acesso somente leitura a um OAC do CloudFront

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCloudFrontServicePrincipalReadOnly",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": [
        "mediastore:GetObject"
      ],
      "Resource":
"arn:aws:mediastore:<region>:111122223333:container/<container name>/*",
      "Condition": {
        "StringEquals": {
          "AWS:SourceArn":
"arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
        },
        "Bool": {
          "aws:SecureTransport": "true"
        }
      }
    }
  ]
}
```

Exemplo Política de contêiner do MediaStore que permite acesso de leitura e gravação a um OAC do CloudFront

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCloudFrontServicePrincipalReadWrite",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": [
        "mediastore:GetObject",
        "mediastore:PutObject"
      ],
      "Resource":
"arn:aws:mediastore:<region>:111122223333:container/<container name>/*",
      "Condition": {
        "StringEquals": {
          "AWS:SourceArn":
"arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
        },
        "Bool": {
          "aws:SecureTransport": "true"
        }
      }
    }
  ]
}
```

Note

Para permitir o acesso de gravação, você deve configurar Allowed HTTP methods (Métodos HTTP permitidos) para incluir PUT nas configurações de comportamento de sua distribuição do CloudFront.

Criar o controle de acesso à origem

Para criar um OAC, você pode usar o AWS Management Console, o AWS CloudFormation, a AWS CLI ou a API do CloudFront.

Console

Como criar um controle de acesso à origem

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Origem access (Acesso à origem).
3. Selecione Create control setting (Criar configuração de controle).
4. No formulário Create control setting (Criar configuração de controle), faça o seguinte:
 - a. No painel Details (Detalhes), insira um Name (Nome) e (opcionalmente) uma Description (Descrição) para o controle de acesso à origem.
 - b. No painel Settings (Configurações), recomendamos que você deixe a configuração padrão (Sign requests (recommended) [Solicitações de assinatura (recomendado)]). Para ter mais informações, consulte [the section called “Configurações avançadas para controle de acesso à origem”](#).
5. Selecione MediaStore no menu suspenso Origin type (Tipo de origem).
6. Escolha Criar.

Depois que o OAC for criado, anote o Name (Nome). Você precisará dele no procedimento a seguir.

Como adicionar um controle de acesso à origem a uma origem do MediaStore em uma distribuição

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha uma distribuição com uma origem do MediaStore à qual você deseja adicionar o OAC e, depois, selecione a guia Origins (Origens).
3. Selecione a origem do MediaStore à qual você deseja adicionar o OAC e, depois, Edit (Editar).
4. Selecione HTTPS only (Somente HTTPS) para o Protocol (Protocolo) de sua origem.
5. No menu suspenso Origin access control (Controle de acesso à origem), selecione o OAC que você deseja usar.
6. Escolha Salvar alterações.

A distribuição começa a ser implantada em todos os locais da borda do CloudFront. Quando um local da borda recebe a nova configuração, ele assina todas as solicitações enviadas à origem de bucket do MediaStore.

CloudFormation

Para criar um controle de acesso à origem (OAC) com o AWS CloudFormation, use o tipo de recurso `AWS::CloudFront::OriginAccessControl`. O exemplo a seguir mostra a sintaxe do modelo AWS CloudFormation no formato YAML, para criar um controle de acesso à origem.

```
Type: AWS::CloudFront::OriginAccessControl
Properties:
  OriginAccessControlConfig:
    Description: An optional description for the origin access control
    Name: ExampleOAC
    OriginAccessControlOriginType: mediastore
    SigningBehavior: always
    SigningProtocol: sigv4
```

Para obter mais informações, consulte [AWS::CloudFront::OriginAccessControl](#) no Guia do usuário do AWS CloudFormation.

CLI

Para criar um controle de acesso à origem com a AWS Command Line Interface (AWS CLI), use o comando `aws cloudfront create-origin-access-control`. É possível usar um arquivo de entrada para fornecer os parâmetros de entrada do comando, em vez de especificar cada parâmetro individual como entrada na linha de comando.

Como criar um controle de acesso à origem (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `origin-access-control.yaml`. Esse arquivo contém todos os parâmetros de entrada para o comando `create-origin-access-control`.

```
aws cloudfront create-origin-access-control --generate-cli-skeleton yaml-input >
origin-access-control.yaml
```

2. Abra o arquivo `origin-access-control.yaml` que você acabou de criar. Edite o arquivo para adicionar um nome para o OAC, uma descrição (opcional) e alterar `SigningBehavior` para `always`. Salve o arquivo.

Para obter mais informações sobre outras configurações de OAC, consulte [the section called “Configurações avançadas para controle de acesso à origem”](#).

3. Use o comando a seguir para criar o controle de acesso à origem usando parâmetros de entrada do arquivo `origin-access-control.yaml`.

```
aws cloudfront create-origin-access-control --cli-input-yaml file://origin-access-control.yaml
```

Anote o valor do Id na saída do comando. Você precisa dele para adicionar o OAC a uma origem do MediaStore em uma distribuição do CloudFront.

Como anexar um OAC a uma origem do MediaStore em uma distribuição existente (CLI com arquivo de entrada)

1. Use o comando a seguir para salvar a configuração da distribuição do CloudFront à qual você deseja adicionar o OAC. A distribuição deve ter uma origem do MediaStore.

```
aws cloudfront get-distribution-config --id <CloudFront distribution ID> --output yaml > dist-config.yaml
```

2. Abra o arquivo chamado `dist-config.yaml` que você acabou de criar. Edite o arquivo fazendo as seguintes alterações:

- No objeto `Origins`, adicione o ID do OAC ao campo chamado `OriginAccessControlId`.
- Remova o valor do campo chamado `OriginAccessIdentity`, se houver.
- Renomeie o campo `ETag` para `IfMatch`, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

3. Use o comando a seguir para atualizar a distribuição para usar o controle de acesso à origem.

```
aws cloudfront update-distribution --id <CloudFront distribution ID> --cli-input-yaml file://dist-config.yaml
```

A distribuição começa a ser implantada em todos os locais da borda do CloudFront. Quando um local da borda recebe a nova configuração, ele assina todas as solicitações enviadas à origem do MediaStore.

API

Para criar um controle de acesso à origem com a API do CloudFront, use [CreateOriginAccessControl](#). Para obter mais informações sobre os campos especificados nessa chamada de API, consulte a documentação de referência de API do seu AWS SDK ou de outro cliente de API.

Assim que criar um controle de acesso à origem, você pode anexá-lo a uma origem do MediaStore em uma distribuição usando uma das seguintes chamadas de API:

- Para anexá-lo a uma distribuição existente, use [UpdateDistribution](#).
- Para anexá-lo a uma nova distribuição, use [CreateDistribution](#).

Para as duas chamadas de API, forneça o ID de controle de acesso à origem no campo `OriginAccessControlId`, dentro de uma origem. Para mais informações sobre os outros campos especificados nessas chamadas de API, consulte [Referência de configurações da distribuição](#) e a documentação de referência da API do AWS SDK ou de outro cliente de API.

Configurações avançadas para controle de acesso à origem

O recurso de controle de acesso à origem do CloudFront inclui configurações avançadas destinadas somente a casos de uso específicos. Use as configurações recomendadas, a menos que você precise usar as configurações avançadas para uma necessidade específica.

O controle de acesso à origem contém uma configuração chamada `Signing behavior` (Comportamento de assinatura) (no console) ou `SigningBehavior` (na API, na CLI e no AWS CloudFormation). Essa configuração fornece as seguintes opções:

Always sign origin requests (recommended setting) [Sempre assinar solicitações de origem (configuração recomendada)]

Recomendamos usar essa configuração, chamada Sign requests (recommended) [Assinar solicitações (recomendado)] no console, ou `always` na API, na CLI e no AWS CloudFormation. Com essa configuração, o CloudFront sempre assina todas as solicitações enviadas à origem do MediaStore.

Never sign origin requests (Nunca assinar solicitações de origem)

Essa configuração é chamada Do not sign requests (Não assinar solicitações) no console ou `never` na API, na CLI e no AWS CloudFormation. Use essa configuração para desativar o controle de acesso à origem para todas as origens em todas as distribuições que usam esse controle. Isso pode economizar tempo e esforço em comparação com a remoção individual de um controle de acesso à origem de todas as origens e distribuições que o usam. Com essa configuração, o CloudFront não assina nenhuma solicitação enviada à origem do MediaStore.

Warning

Para usar essa configuração, a origem do MediaStore deve estar acessível ao público. Se você usar essa configuração com uma origem do MediaStore que não esteja acessível ao público, o CloudFront não poderá acessar a origem. A origem do MediaStore retorna erros ao CloudFront e o CloudFront transmite esses erros aos visualizadores. Para ter mais informações, consulte o exemplo de política de contêiner do MediaStore para [acesso público de leitura por HTTPS](#).

Não substituir o cabeçalho **Authorization** do visualizador (cliente)

Essa configuração é chamada Do not override authorization header (Não substituir o cabeçalho de autorização) no console ou `no-override` na API, na CLI e no AWS CloudFormation. Use essa configuração quando quiser que o CloudFront assine solicitações de origem somente quando a solicitação do visualizador correspondente não incluir um cabeçalho **Authorization**. Com essa configuração, o CloudFront transmite o cabeçalho **Authorization** da solicitação do visualizador quando houver, mas assina a solicitação de origem (adicionando seu próprio cabeçalho **Authorization**) quando a solicitação do visualizador não inclui um cabeçalho **Authorization**.

⚠ Warning

Para transmitir o cabeçalho `Authorization` da solicitação do visualizador, você deve adicionar o cabeçalho `Authorization` a uma [política de cache](#) para todos os comportamentos de cache que usam origens do MediaStore associadas a esse controle de acesso à origem.

Restringir o acesso a uma origem do URL de função do AWS Lambda

O CloudFront fornece controle de acesso à origem (OAC) para restringir o acesso a uma origem do URL da função do Lambda.

Tópicos

- [Criar um OAC](#)
- [Configurações avançadas para controle de acesso à origem](#)

Criar um OAC

Conclua as etapas descritas nos tópicos a seguir para configurar um novo OAC no CloudFront.

i Note

Se você usar os métodos PUT ou POST com o URL de função do Lambda, os usuários deverão incluir o valor de hash da carga útil no cabeçalho `x-amz-content-sha256` ao enviar a solicitação ao CloudFront. O Lambda não é compatível com cargas úteis não conectadas.

Tópicos

- [Pré-requisitos](#)
- [Permitir acesso do OAC ao URL de função do Lambda](#)
- [Criar o OAC](#)

Pré-requisitos

Antes de criar e configurar o OAC, você deve ter uma distribuição do CloudFront com um URL da função do Lambda como origem. Para ter mais informações, consulte [Usar um URL da função do Lambda](#).

Permitir acesso do OAC ao URL de função do Lambda

Antes de criar um OAC ou configurá-lo em uma distribuição do CloudFront, ele deve ter permissão para acessar o URL da função do Lambda. Faça isso depois de criar uma distribuição do CloudFront, mas antes de adicionar o OAC ao URL da função do Lambda na configuração de distribuição.

Note

Para atualizar a política do IAM para o URL da função do Lambda, você deve usar o AWS Command Line Interface (AWS CLI). Não há suporte para a edição da política do IAM no console do Lambda, no momento.

O comando AWS CLI a seguir concede ao serviço de entidade principal (`cloudfront.amazonaws.com`) do CloudFront acesso ao URL da função do Lambda. O elemento `Condition` na política permite que o CloudFront acesse o Lambda somente quando a solicitação for em nome da distribuição do CloudFront que contém o URL da função do Lambda.

Example : o comando AWS CLI para atualizar uma política para permitir acesso somente leitura a um OAC do CloudFront

O comando AWS CLI a seguir permite que a distribuição do CloudFront (`E1PDK09ESKHJWT`) acesse seu `FUNCTION_URL_NAME` do Lambda.

```
aws lambda add-permission \  
--statement-id "AllowCloudFrontServicePrincipal" \  
--action "lambda:InvokeFunctionUrl" \  
--principal "cloudfront.amazonaws.com" \  
--source-arn "arn:aws:cloudfront::123456789012:distribution/E1PDK09ESKHJWT" \  
--function-name FUNCTION_URL_NAME
```

Note

Se você criar uma distribuição e ela não tiver permissão para o URL da função do Lambda, é possível selecionar Copy CLI command (Copiar comando CLI) no console do CloudFront

e, em seguida, inserir esse comando no seu terminal de linha de comando. Para ter mais informações, consulte [Conceder acesso de função aos Serviços da AWS](#) no Guia do desenvolvedor do AWS Lambda.

Criar o OAC

Para criar um OAC, é possível usar o AWS Management Console, o AWS CloudFormation, a AWS CLI ou a API do CloudFront.

Console

Para criar um OAC

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Origem access (Acesso à origem).
3. Selecione Create control setting (Criar configuração de controle).
4. No formulário Create new OAC (Criar novo OAC), faça o seguinte:
 - a. Informe um Name (Nome) e (opcionalmente) uma Description (Descrição) para o OAC.
 - b. Em Signing behavior (Comportamento de assinatura), recomendamos que deixe a configuração padrão (Sign requests (recommended) [Solicitações de assinatura (recomendado)]). Para ter mais informações, consulte [the section called “Configurações avançadas para controle de acesso à origem”](#).
5. Em Origin type (Tipo de origem), selecione Lambda.
6. Escolha Criar.

Tip

Depois que criar OAC, anote o Name (Nome). Você precisará dele no procedimento a seguir.

Como adicionar um controle de acesso à origem a um URL da função do Lambda em uma distribuição

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.

2. Escolha uma distribuição com um URL da função do Lambda ao qual você deseja adicionar o OAC e, depois, selecione a guia Origins (Origens).
3. Selecione o URL da função do Lambda ao qual você deseja adicionar o OAC e, depois, Edit (Editar).
4. Selecione HTTPS only (Somente HTTPS) para o Protocol (Protocolo) de sua origem.
5. No menu suspenso Origin access control (Controle de acesso à origem), selecione o nome do OAC que você deseja usar.
6. Escolha Salvar alterações.

A distribuição começa a ser implantada em todos os locais da borda do CloudFront. Quando um local da borda recebe a nova configuração, ele assina todas as solicitações enviadas ao URL da função do Lambda.

CloudFormation

Para criar um OAC com o AWS CloudFormation, use o tipo de recurso `AWS::CloudFront::OriginAccessControl`. O exemplo a seguir mostra a sintaxe do modelo AWS CloudFormation no formato YAML, para criar um OAC.

```
Type: AWS::CloudFront::OriginAccessControl
Properties:
  OriginAccessControlConfig:
    Description: An optional description for the origin access control
    Name: ExampleOAC
    OriginAccessControlOriginType: lambda
    SigningBehavior: always
    SigningProtocol: sigv4
```

Para obter mais informações, consulte [AWS::CloudFront::OriginAccessControl](#) no Guia do usuário do AWS CloudFormation.

CLI

Para criar um controle de acesso à origem com a AWS Command Line Interface (AWS CLI), use o comando `aws cloudfront create-origin-access-control`. É possível usar um arquivo de entrada para fornecer os parâmetros de entrada do comando, em vez de especificar cada parâmetro individual como entrada na linha de comando.

Como criar um controle de acesso à origem (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `origin-access-control.yaml`. Esse arquivo contém todos os parâmetros de entrada para o comando `create-origin-access-control`.

```
aws cloudfront create-origin-access-control --generate-cli-skeleton yml-input >
origin-access-control.yaml
```

2. Abra o arquivo `origin-access-control.yaml` que você acabou de criar. Edite o arquivo para adicionar um nome para o OAC, uma descrição (opcional) e alterar `SigningBehavior` para `always`. Salve o arquivo.

Para obter mais informações sobre outras configurações de OAC, consulte [the section called “Configurações avançadas para controle de acesso à origem”](#).

3. Use o comando a seguir para criar o controle de acesso à origem usando parâmetros de entrada do arquivo `origin-access-control.yaml`.

```
aws cloudfront create-origin-access-control --cli-input-yml file://origin-
access-control.yaml
```

Anote o valor do `Id` na saída do comando. Você precisa dele para adicionar o OAC a um URL da função do Lambda em uma distribuição do CloudFront.

Como anexar um OAC a um URL da função do Lambda em uma distribuição existente (CLI com arquivo de entrada)

1. Use o comando a seguir para salvar a configuração da distribuição do CloudFront à qual você deseja adicionar o OAC. A distribuição deve ter um URL da função do Lambda como origem.

```
aws cloudfront get-distribution-config --id <CloudFront distribution ID> --
output yml > dist-config.yaml
```

2. Abra o arquivo chamado `dist-config.yaml` que você acabou de criar. Edite o arquivo fazendo as seguintes alterações:

- No objeto `Origins`, adicione o ID do OAC ao campo chamado `OriginAccessControlId`.
- Remova o valor do campo chamado `OriginAccessIdentity`, se houver.
- Renomeie o campo `ETag` para `IfMatch`, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

3. Use o comando a seguir para atualizar a distribuição para usar o controle de acesso à origem.

```
aws cloudfront update-distribution --id <CloudFront distribution ID> --cli-input-yaml file://dist-config.yaml
```

A distribuição começa a ser implantada em todos os locais da borda do CloudFront. Quando um local da borda recebe a nova configuração, ele assina todas as solicitações enviadas ao URL da função do Lambda.

API

Para criar um OAC com a API do CloudFront, use [CreateOriginAccessControl](#). Para obter mais informações sobre os campos especificados nessa chamada de API, consulte a documentação de referência de API do seu AWS SDK ou de outro cliente de API.

Assim que criar um controle de acesso à origem, é possível anexá-lo a um URL da função do Lambda em uma distribuição usando uma das seguintes chamadas de API:

- Para anexá-lo a uma distribuição existente, use [UpdateDistribution](#).
- Para anexá-lo a uma nova distribuição, use [CreateDistribution](#).

Para as duas chamadas de API, forneça o ID de OAC no campo `OriginAccessControlId`, dentro de uma origem. Para ter mais informações sobre os outros campos especificados nessas chamadas de API, consulte a documentação de referência da API do SDK ou de outro cliente de API da AWS.

Configurações avançadas para controle de acesso à origem

O atributo de OAC do CloudFront inclui configurações avançadas destinadas somente a casos de uso específicos. Use as configurações recomendadas, a menos que você precise usar as configurações avançadas para uma necessidade específica.

O OAC contém uma configuração chamada `Signing behavior` (Comportamento de assinatura) (no console) ou `SigningBehavior` (na API, na CLI e no AWS CloudFormation). Essa configuração fornece as seguintes opções:

`Always sign origin requests (recommended setting)` [Sempre assinar solicitações de origem (configuração recomendada)]

Recomendamos usar essa configuração, chamada `Sign requests (recommended)` [Assinar solicitações (recomendado)] no console, ou `always` na API, na CLI e no AWS CloudFormation. Com essa configuração, o CloudFront sempre assina todas as solicitações enviadas ao URL da função do Lambda.

`Never sign origin requests (Nunca assinar solicitações de origem)`

Essa configuração é chamada `Do not sign requests` (Não assinar solicitações) no console ou `never` na API, na CLI e no AWS CloudFormation. Use essa configuração para desativar o OAC para todas as origens em todas as distribuições que usam esse OAC. Isso pode economizar tempo e esforço em comparação com a remoção individual de um OAC de todas as origens e distribuições que o usam. Com essa configuração, o CloudFront não assina nenhuma solicitação enviada ao URL da função do Lambda.

Warning

Para usar essa configuração, o URL da função do Lambda deve estar acessível de forma pública. Se você usar essa configuração com um URL da função do Lambda que não esteja acessível de forma pública, o CloudFront não poderá acessar a origem. A origem do URL da função do Lambda retorna erros ao CloudFront e o CloudFront transmite esses erros aos visualizadores. Para ter mais informações, consulte [Modelo de segurança e autenticação para URLs de função do Lambda](#) no Guia do usuário do AWS Lambda.

Não substituir o cabeçalho **Authorization** do visualizador (cliente)

Essa configuração é chamada `Do not override authorization header` (Não substituir o cabeçalho de autorização) no console ou `no-override` na API, na CLI e no AWS CloudFormation. Use essa configuração quando quiser que o CloudFront assine solicitações de origem somente quando a solicitação do visualizador correspondente não incluir um cabeçalho `Authorization`. Com essa configuração, o CloudFront transmite o cabeçalho `Authorization` da solicitação do visualizador quando houver, mas assina a solicitação de origem (adicionando seu próprio cabeçalho `Authorization`) quando a solicitação do visualizador não inclui um cabeçalho `Authorization`.

Warning

- Se usar essa configuração, você deverá especificar a assinatura do Signature versão 4 para o URL da função do Lambda em vez do nome ou CNAME da distribuição do CloudFront. Quando o CloudFront encaminha o cabeçalho `Authorization` da solicitação do visualizador para o URL da função do Lambda, o Lambda valida a assinatura em relação ao host do domínio de URL do Lambda. Se a assinatura não for baseada no domínio de URL do Lambda, o host na assinatura não corresponderá ao host usado pela origem do URL do Lambda. Isso significa que a solicitação falhará, ocasionando um erro de validação da assinatura.
- Para transmitir o cabeçalho de `Authorization` da solicitação do visualizador, você deve adicionar o cabeçalho de `Authorization` a uma [política de cache](#) para todos os comportamentos de cache que usam URLs da função do Lambda associados a esse controle de acesso à origem.

Restringir o acesso a uma origem do Amazon Simple Storage Service

O CloudFront fornece duas maneiras para enviar solicitações autenticadas a uma origem do Amazon S3: controle de acesso à origem (OAC) e identidade do acesso de origem (OAI). O OAC ajuda a proteger suas origens, assim como no Amazon S3. Recomendamos o uso do OAC porque ele é compatível com:

- Todos os buckets do Amazon S3 em todas as Regiões da AWS; por exemplo, regiões opcionais lançadas após dezembro de 2022.
- [Criptografia do lado do servidor com o AWS KMS](#) (SSE-KMS) do Amazon S3

- Solicitações dinâmicas (PUT e DELETE) para o Amazon S3

A identidade do acesso de origem (OAI) não funciona para cenários na lista anterior, ou exige outras soluções alternativas nesses casos. Os tópicos a seguir descrevem como usar o controle de acesso de origem (OAC) com uma origem do Amazon S3. Para obter informações sobre como migrar da identidade do acesso de origem (OAI) para o controle de acesso de origem (OAC), consulte [the section called “Migrar da identidade do acesso de origem \(OAI\) para o controle de acesso à origem \(OAC\)”](#).

Observações

- Ao usar o CloudFront OAC com origens de bucket do Amazon S3, você deve definir a Propriedade de objeto do Amazon S3 como Proprietário do bucket aplicado, o padrão para novos buckets do Amazon S3. Se você precisar de ACLs, use a configuração Proprietário do bucket preferido para manter o controle sobre os objetos enviados via CloudFront.
- Se usar um bucket do Amazon S3 configurado como um [endpoint de site](#), você deverá configurá-lo com o CloudFront como uma origem personalizada. Isso significa que você não pode usar o OAC (nem a OAI). O OAC não é compatível com o redirecionamento de origem usando o Lambda@Edge.

Tópicos

- [the section called “Criar um controle de acesso à origem”](#)
- [the section called “Excluir uma distribuição com um OAC anexado a um bucket do S3”](#)
- [the section called “Migrar da identidade do acesso de origem \(OAI\) para o controle de acesso à origem \(OAC\)”](#)
- [the section called “Configurações avançadas para controle de acesso à origem”](#)

Criar um controle de acesso à origem

Conclua as etapas descritas nos tópicos a seguir para configurar um novo controle de acesso à origem no CloudFront.

Tópicos

- [Pré-requisitos](#)

- [Conceder permissão ao controle de acesso à origem para acessar o bucket do S3](#)
- [Criar o controle de acesso à origem](#)

Pré-requisitos

Antes de criar e configurar o controle de acesso à origem (OAC), você deve ter uma distribuição do CloudFront com uma origem de bucket do Amazon S3. Essa origem deve ser um bucket normal do S3, não um bucket configurado como [endpoint de site](#). Para obter mais informações sobre como configurar uma distribuição do CloudFront com uma origem de bucket do S3, consulte [the section called “Conceitos básicos de uma distribuição básica”](#).

Note

Ao usar o OAC para proteger sua origem de bucket do S3, a comunicação entre o CloudFront e o Amazon S3 é sempre por meio de HTTPS, independentemente de suas configurações específicas.

Conceder permissão ao controle de acesso à origem para acessar o bucket do S3

Antes de criar um controle de acesso à origem (OAC) ou configurá-lo em uma distribuição do CloudFront, verifique se o OAC tem permissão para acessar a origem do bucket do S3. Faça isso depois de criar uma distribuição do CloudFront, mas antes de adicionar o OAC à origem do S3 na configuração de distribuição.

Para conceder permissão ao OAC para acessar o bucket do S3, use uma [política de bucket](#) do S3 para possibilitar que a entidade principal de serviço do CloudFront (`cloudfront.amazonaws.com`) acesse o bucket. Use um elemento `Condition` na política para permitir que o CloudFront acesse o bucket somente quando a solicitação for em nome da distribuição do CloudFront que contém a origem do S3.

Para obter informações sobre como adicionar ou modificar uma política de bucket, consulte [Adicionar uma política de bucket usando o console do Amazon S3](#) no Guia do usuário do Amazon S3.

Veja a seguir exemplos de políticas de bucket do S3 que permitem que um OAC do CloudFront acesse uma origem do S3.

Example Política de bucket do S3 que permite acesso somente leitura a um OAC do CloudFront

```
{
```

```

"Version": "2012-10-17",
"Statement": {
  "Sid": "AllowCloudFrontServicePrincipalReadOnly",
  "Effect": "Allow",
  "Principal": {
    "Service": "cloudfront.amazonaws.com"
  },
  "Action": "s3:GetObject",
  "Resource": "arn:aws:s3:::<S3 bucket name>/*",
  "Condition": {
    "StringEquals": {
      "AWS:SourceArn":
"arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
    }
  }
}
}

```

Exemplo Política de bucket do S3 que permite acesso de leitura e gravação a um OAC do CloudFront

```

{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "AllowCloudFrontServicePrincipalReadWrite",
    "Effect": "Allow",
    "Principal": {
      "Service": "cloudfront.amazonaws.com"
    },
    "Action": [
      "s3:GetObject",
      "s3:PutObject"
    ],
    "Resource": "arn:aws:s3:::<S3 bucket name>/*",
    "Condition": {
      "StringEquals": {
        "AWS:SourceArn":
"arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
      }
    }
  }
}

```

SSE-KMS

Se os objetos na origem do bucket do S3 forem criptografados usando [criptografia do lado do servidor com o AWS Key Management Service \(SSE-KMS\)](#), o OAC deverá ter permissão para usar a chave do AWS KMS. Para conceder permissão ao OAC para usar a chave do KMS, adicione uma instrução à [política de chaves do KMS](#). Para obter informações sobre como modificar uma política de chaves, consulte [Alterar uma política de chaves](#) no Guia do desenvolvedor do AWS Key Management Service.

O exemplo a seguir mostra uma declaração de política de chaves do KMS que permite ao OAC usar a chave do KMS.

Exemplo Declaração de política de chaves do KMS que permite a um OAC do CloudFront acessar uma chave KMS para o SSE-KMS

```
{
  "Sid": "AllowCloudFrontServicePrincipalSSE-KMS",
  "Effect": "Allow",
  "Principal": {
    "Service": [
      "cloudfront.amazonaws.com"
    ]
  },
  "Action": [
    "kms:Decrypt",
    "kms:Encrypt",
    "kms:GenerateDataKey*"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "AWS:SourceArn":
        "arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
    }
  }
}
```

Criar o controle de acesso à origem

Para criar um controle de acesso à origem (OAC), você pode usar o AWS Management Console, o AWS CloudFormation, a AWS CLI ou a API do CloudFront.

Console

Como criar um controle de acesso à origem

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Origem access (Acesso à origem).
3. Selecione Create control setting (Criar configuração de controle).
4. No formulário Create control setting (Criar configuração de controle), faça o seguinte:
 - a. No painel Details (Detalhes), insira um Name (Nome) e (opcionalmente) uma Description (Descrição) para o controle de acesso à origem.
 - b. No painel Settings (Configurações), recomendamos que você deixe a configuração padrão (Sign requests (recommended) [Solicitações de assinatura (recomendado)]). Para ter mais informações, consulte [the section called “Configurações avançadas para controle de acesso à origem”](#).
5. Selecione S3 no menu suspenso Origin type (Tipo de origem).
6. Escolha Criar.

Depois que o OAC for criado, anote o Name (Nome). Você precisará dele no procedimento a seguir.

Como adicionar um controle de acesso à origem a uma origem do S3 em uma distribuição

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha uma distribuição com uma origem do S3 à qual você deseja adicionar o OAC e, depois, selecione a guia Origins (Origens).
3. Selecione a origem do S3 à qual você deseja adicionar o OAC e, depois, Edit (Editar).
4. Em Origin access (Acesso à origem), selecione Origin access control settings (recommended) (Configurações de controle de acesso à origem [recomendado]).
5. No menu suspenso Origin access control (Controle de acesso à origem), selecione o OAC que você deseja usar.
6. Escolha Salvar alterações.

A distribuição começa a ser implantada em todos os locais da borda do CloudFront. Quando um local da borda recebe a nova configuração, ele assina todas as solicitações enviadas à origem do bucket do S3.

CloudFormation

Para criar um controle de acesso à origem (OAC) com o AWS CloudFormation, use o tipo de recurso `AWS::CloudFront::OriginAccessControl`. O exemplo a seguir mostra a sintaxe do modelo AWS CloudFormation no formato YAML, para criar um controle de acesso à origem.

```
Type: AWS::CloudFront::OriginAccessControl
Properties:
  OriginAccessControlConfig:
    Description: An optional description for the origin access control
    Name: ExampleOAC
    OriginAccessControlOriginType: s3
    SigningBehavior: always
    SigningProtocol: sigv4
```

Para obter mais informações, consulte [AWS::CloudFront::OriginAccessControl](#) no Guia do usuário do AWS CloudFormation.

CLI

Para criar um controle de acesso à origem com a AWS Command Line Interface (AWS CLI), use o comando `aws cloudfront create-origin-access-control`. É possível usar um arquivo de entrada para fornecer os parâmetros de entrada do comando, em vez de especificar cada parâmetro individual como entrada na linha de comando.

Como criar um controle de acesso à origem (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `origin-access-control.yaml`. Esse arquivo contém todos os parâmetros de entrada para o comando `create-origin-access-control`.

```
aws cloudfront create-origin-access-control --generate-cli-skeleton yaml-input >
origin-access-control.yaml
```

2. Abra o arquivo `origin-access-control.yaml` que você acabou de criar. Edite o arquivo para adicionar um nome para o OAC, uma descrição (opcional) e alterar `SigningBehavior` para `always`. Salve o arquivo.

Para obter mais informações sobre outras configurações de OAC, consulte [the section called “Configurações avançadas para controle de acesso à origem”](#).

- Use o comando a seguir para criar o controle de acesso à origem usando parâmetros de entrada do arquivo `origin-access-control.yaml`.

```
aws cloudfront create-origin-access-control --cli-input-yaml file://origin-access-control.yaml
```

Anote o valor do Id na saída do comando. Você precisa dele para adicionar o OAC a uma origem de bucket do S3 em uma distribuição do CloudFront.

Como anexar um OAC a uma origem de bucket do S3 em uma distribuição existente (CLI com arquivo de entrada)

- Use o comando a seguir para salvar a configuração da distribuição do CloudFront à qual você deseja adicionar o OAC. A distribuição deve ter uma origem de bucket do S3.

```
aws cloudfront get-distribution-config --id <CloudFront distribution ID> --output yaml > dist-config.yaml
```

- Abra o arquivo chamado `dist-config.yaml` que você acabou de criar. Edite o arquivo fazendo as seguintes alterações:
 - No objeto `Origins`, adicione o ID do OAC ao campo chamado `OriginAccessControlId`.
 - Remova o valor do campo chamado `OriginAccessIdentity`, se houver.
 - Renomeie o campo `ETag` para `IfMatch`, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

- Use o comando a seguir para atualizar a distribuição para usar o controle de acesso à origem.

```
aws cloudfront update-distribution --id <CloudFront distribution ID> --cli-  
input-yaml file://dist-config.yaml
```

A distribuição começa a ser implantada em todos os locais da borda do CloudFront. Quando um local da borda recebe a nova configuração, ele assina todas as solicitações enviadas à origem do bucket do S3.

API

Para criar um controle de acesso à origem com a API do CloudFront, use [CreateOriginAccessControl](#). Para obter mais informações sobre os campos especificados nessa chamada de API, consulte a documentação de referência de API do seu AWS SDK ou de outro cliente de API.

Assim que criar um controle de acesso à origem, você pode anexá-lo a uma origem de bucket do S3 em uma distribuição usando uma das seguintes chamadas de API:

- Para anexá-lo a uma distribuição existente, use [UpdateDistribution](#).
- Para anexá-lo a uma nova distribuição, use [CreateDistribution](#).

Para as duas chamadas de API, forneça o ID de controle de acesso à origem no campo `OriginAccessControlId`, dentro de uma origem. Para mais informações sobre os outros campos especificados nessas chamadas de API, consulte [Referência de configurações da distribuição](#) e a documentação de referência da API do AWS SDK ou de outro cliente de API.

Excluir uma distribuição com um OAC anexado a um bucket do S3

Se precisar excluir uma distribuição com um OAC anexado a um bucket do S3, exclua a distribuição antes de excluir a origem do bucket do S3. Como alternativa, inclua a Região no nome de domínio de origem. Se isso não for possível, é possível remover o OAC da distribuição mudando para público antes da exclusão. Para ter mais informações, consulte [Excluir uma distribuição do](#) .

Migrar da identidade do acesso de origem (OAI) para o controle de acesso à origem (OAC)

Para migrar de uma identidade do acesso de origem (OAI) herdada para um controle de acesso à origem (OAC), primeiro atualize a origem do bucket do S3 para permitir que a OAI e o OAC acessem o conteúdo do bucket. Isso garante que o CloudFront nunca perca o acesso ao bucket durante a

transição. Para permitir que a OAI e o OAC acessem um bucket do S3, atualize a [política de bucket](#) para incluir duas declarações, uma para cada tipo de entidade principal.

O exemplo de política de bucket do S3 a seguir permite que uma OAI e um OAC acessem uma origem do S3.

Exemplo Política de bucket do S3 que permite acesso somente leitura a uma OAI e um OAC

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCloudFrontServicePrincipalReadOnly",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::<S3 bucket name>/*",
      "Condition": {
        "StringEquals": {
          "AWS:SourceArn":
            "arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
        }
      }
    },
    {
      "Sid": "AllowLegacyOAIReadOnly",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access
Identity <origin access identity ID>"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::<S3 bucket name>/*"
    }
  ]
}
```

Depois de atualizar a política de bucket da origem do S3 para permitir o acesso à OAI e ao OAC, você pode atualizar a configuração de distribuição para usar o OAC em vez da OAI. Para ter mais informações, consulte [the section called “Criar um controle de acesso à origem”](#).

Depois que a distribuição estiver totalmente implantada, você poderá remover a declaração na política de bucket que permite o acesso à OAI. Para ter mais informações, consulte [the section called “Conceder permissão ao controle de acesso à origem para acessar o bucket do S3”](#).

Configurações avançadas para controle de acesso à origem

O recurso de controle de acesso à origem do CloudFront inclui configurações avançadas destinadas somente a casos de uso específicos. Use as configurações recomendadas, a menos que você precise usar as configurações avançadas para uma necessidade específica.

O controle de acesso à origem contém uma configuração chamada Signing behavior (Comportamento de assinatura) (no console) ou SigningBehavior (na API, na CLI e no AWS CloudFormation). Essa configuração fornece as seguintes opções:

Always sign origin requests (recommended setting) [Sempre assinar solicitações de origem (configuração recomendada)]

Recomendamos usar essa configuração, chamada Sign requests (recommended) [Assinar solicitações (recomendado)] no console, ou `always` na API, na CLI e no AWS CloudFormation. Com essa configuração, o CloudFront sempre assina todas as solicitações enviadas à origem do bucket do S3.

Never sign origin requests (Nunca assinar solicitações de origem)

Essa configuração é chamada Do not sign requests (Não assinar solicitações) no console ou `never` na API, na CLI e no AWS CloudFormation. Use essa configuração para desativar o controle de acesso à origem para todas as origens em todas as distribuições que usam esse controle. Isso pode economizar tempo e esforço em comparação com a remoção individual de um controle de acesso à origem de todas as origens e distribuições que o usam. Com essa configuração, o CloudFront não assina nenhuma solicitação enviada à origem do bucket do S3.

Warning

Para usar essa configuração, a origem do bucket do S3 deve estar acessível ao público. Se você usar essa configuração com uma origem de bucket do S3 que não esteja acessível ao público, o CloudFront não poderá acessar a origem. A origem do bucket do S3 retorna erros ao CloudFront e o CloudFront transmite esses erros aos visualizadores.

Não substituir o cabeçalho **Authorization** do visualizador (cliente)

Essa configuração é chamada Do not override authorization header (Não substituir o cabeçalho de autorização) no console ou no-override na API, na CLI e no AWS CloudFormation. Use essa configuração quando quiser que o CloudFront assine solicitações de origem somente quando a solicitação do visualizador correspondente não incluir um cabeçalho Authorization. Com essa configuração, o CloudFront transmite o cabeçalho Authorization da solicitação do visualizador quando houver, mas assina a solicitação de origem (adicionando seu próprio cabeçalho Authorization) quando a solicitação do visualizador não inclui um cabeçalho Authorization.

Warning

Para transmitir o cabeçalho Authorization da solicitação do visualizador, você deve adicionar o cabeçalho Authorization a uma [política de cache](#) para todos os comportamentos de cache que usam origens de bucket do S3 associadas a esse controle de acesso à origem.

Usar uma identidade de acesso à origem (herdada, não recomendado)

Visão geral da identidade do acesso de origem

A identidade do acesso de origem (OAI) do CloudFront fornece funcionalidade semelhante ao controle de acesso à origem (OAC), mas não funciona em todos os cenários. É por isso que recomendamos usar o OAC em vez da OAI. O OAI não é compatível especificamente com:

- Buckets do Amazon S3 em todas as Regiões da AWS, inclusive regiões opcionais.
- [Criptografia do lado do servidor com o AWS KMS](#) (SSE-KMS) do Amazon S3
- Solicitações dinâmicas (PUT, POST ou DELETE) para o Amazon S3
- Novas Regiões da AWS lançadas após dezembro de 2022

Para obter informações sobre como migrar de OAI para OAC, consulte [the section called “Migrar da identidade do acesso de origem \(OAI\) para o controle de acesso à origem \(OAC\)”](#).

Conceder permissão a uma identidade de acesso à origem para ler arquivos no bucket do Amazon S3

Ao criar uma OAI ou adicionar uma a uma distribuição com o console do CloudFront, é possível atualizar automaticamente a política de bucket do Amazon S3 para conceder à OAI permissão para acessar seu bucket. Você também pode optar por criar ou atualizar manualmente a política do bucket. Seja qual for o método usado, você ainda deverá revisar as permissões para se certificar de que:

- Sua OAI do CloudFront pode acessar arquivos no bucket em nome dos visualizadores que os solicitam por meio do CloudFront.
- Os visualizadores não podem usar URLs do Amazon S3 para acessar seus arquivos fora do CloudFront.

Important

Se você configurar o CloudFront para aceitar e encaminhar todos os métodos HTTP compatíveis com o CloudFront, certifique-se de conceder à OAI do CloudFront as permissões desejadas. Por exemplo, se você configurar o CloudFront para aceitar e encaminhar solicitações que usem o método DELETE, configure sua política de bucket para lidar com as solicitações DELETE de maneira adequada, para que os visualizadores possam excluir somente os arquivos desejados.

Usar políticas de bucket do Amazon S3

É possível conceder a uma OAI do CloudFront acesso a arquivos em um bucket do Amazon S3 criando ou atualizando a política de bucket das seguintes maneiras:

- Usando a guia Permissions (Permissões) do bucket do Amazon S3 no [console do Amazon S3](#).
- Usando o [PutBucketPolicy](#) na API do Amazon S3.
- Usando o [console do CloudFront](#). Ao adicionar uma OAI às suas configurações de origem no console do CloudFront, é possível escolher Yes, update the bucket policy (Sim, atualizar a política de bucket) para informar o CloudFront para atualizar a política de bucket em seu nome.

Se você atualizar a política de bucket manualmente, certifique-se de:

- Especificar a OAI correta como `Principal` na política.
- Conceder à OAI as permissões necessárias para acessar objetos em nome dos visualizadores.

Para obter mais informações, consulte as seções a seguir.

Especificar uma OAI como **Principal** em uma política de bucket

Para especificar uma OAI como `Principal` em uma política de bucket do Amazon S3, use o nome do recurso da Amazon (ARN) da OAI, que inclui o respectivo ID. Por exemplo:

```
"Principal": {
  "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access Identity <origin
access identity ID>"
}
```

Encontre o ID do OAI no console do CloudFront Security (Segurança), Origin access (Acesso à origem), Identities (legacy) (Identidades [legado]). Como alternativa, use [ListCloudFrontOriginAccessIdentities](#) na API do CloudFront.

Conceder permissões a uma OAI

Para conceder permissões à OAI para acessar objetos em seu bucket do Amazon S3, use ações na política que se relacionem a operações de API específicas do Amazon S3. Por exemplo, a ação `s3:GetObject` possibilita que a OAI leia objetos no bucket. Para obter mais informações, consulte os exemplos na seção a seguir, ou consulte [Ações do Amazon S3](#) no Guia do usuário do Amazon Simple Storage Service.

Exemplos de política de bucket do Amazon S3

Os exemplos a seguir mostram políticas de bucket do Amazon S3 que permitem que a OAI do CloudFront acesse um bucket do S3.

Encontre o ID do OAI no console do CloudFront Security (Segurança), Origin access (Acesso à origem), Identities (legacy) (Identidades [legado]). Como alternativa, use [ListCloudFrontOriginAccessIdentities](#) na API do CloudFront.

Example Política de bucket do Amazon S3 que concede à OAI acesso de leitura

O exemplo a seguir permite que a OAI leia objetos no bucket especificado (`s3:GetObject`).

```
{
  "Version": "2012-10-17",
  "Id": "PolicyForCloudFrontPrivateContent",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access
Identity <origin access identity ID>"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::<S3 bucket name>/*"
    }
  ]
}
```

Exemplo Política de bucket do Amazon S3 que concede à OAI acesso de leitura e gravação

O exemplo a seguir permite que a OAI leia e grave objetos no bucket especificado (`s3:GetObject` e `s3:PutObject`). Isso permite que os visualizadores façam upload de arquivos no bucket do Amazon S3 por meio do CloudFront.

```
{
  "Version": "2012-10-17",
  "Id": "PolicyForCloudFrontPrivateContent",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access
Identity <origin access identity ID>"
      },
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Resource": "arn:aws:s3:::<S3 bucket name>/*"
    }
  ]
}
```

Usar ACLs de objeto do Amazon S3 (não recomendado)

Important

Recomendamos o [uso de políticas de bucket do Amazon S3](#) para conceder a uma OAI acesso a um bucket do S3. Você pode usar listas de controle de acesso (ACLs) conforme descrito nesta seção, mas não recomendamos fazê-lo.

O Amazon S3 recomenda a configuração de [S3 Object Ownership](#) (Propriedade do objeto do S3) como bucket owner enforced (aplicado pelo proprietário do bucket), o que significa que as ACLs estão desabilitadas para o bucket e os objetos nele contidos. Ao aplicar essa configuração para Object Ownership (Propriedade de objeto), você deve usar políticas de bucket para conceder acesso à OAI (consulte a seção anterior).

Esta seção a seguir é apenas para casos de uso herdados que exigem ACLs.

É possível conceder a uma OAI do CloudFront acesso a arquivos em um bucket do Amazon S3 criando ou atualizando a ACL do arquivo das seguintes maneiras:

- Usando a guia Permissions (Permissões) do objeto do Amazon S3 no [console do Amazon S3](#).
- Usando [PutObjectAcl](#) na API do Amazon S3.

Ao conceder acesso a uma OAI usando uma ACL, é necessário especificar a OAI por meio de seu ID de usuário canônico do Amazon S3. No console do CloudFront, é possível encontrar este ID em Security (Segurança), Origin access (Acesso à origem), Identities (legacy) (Identidades [legado]). Se estiver usando a API do CloudFront, use o valor do elemento `S3CanonicalUserId` retornado quando você criou a OAI ou chame [ListCloudFrontOriginAccessIdentities](#) na API do CloudFront.

Usar uma identidade de acesso à origem nas regiões do Amazon S3 compatíveis apenas com a autenticação Signature versão 4

As regiões mais recentes do Amazon S3 exigem o uso do Signature Version 4 para solicitações autenticadas. (Para ver as versões de assinatura com suporte em cada região do Amazon S3, consulte [Endpoints e cotas do Amazon Simple Storage Service](#) na Referência geral da AWS). Se estiver usando uma identidade de acesso de origem e seu bucket estiver em uma das regiões que exijam o Signature versão 4, observe o seguinte:

- As solicitações DELETE, GET, HEAD, OPTIONS e PATCH são compatíveis sem qualificações.
- As solicitações POST não são compatíveis.

Restringir o acesso aos Application Load Balancers

Para uma aplicação web ou outro conteúdo fornecido por um Application Load Balancer voltado à internet no Elastic Load Balancing, o CloudFront pode armazenar objetos em cache e disponibilizá-los diretamente aos usuários (visualizadores), reduzindo a carga no Application Load Balancer. Um balanceador de carga voltado para a internet tem um nome de DNS que pode ser resolvido publicamente e roteia solicitações de clientes até destinos na internet.

O CloudFront também pode ajudar a reduzir a latência e até mesmo absorver alguns ataques de negação distribuída de serviço (DDoS).

No entanto, se os usuários puderem ignorar o CloudFront e acessar seu Application Load Balancer diretamente, você não obterá esses benefícios. Mas você pode configurar o Amazon CloudFront e seu Application Load Balancer para impedir que os usuários acessem diretamente o Application Load Balancer. Assim os usuários podem acessar o Application Load Balancer somente por meio do CloudFront, assegurando que você obtenha os benefícios de usá-lo.

Para impedir que os usuários acessem diretamente um Application Load Balancer e permitir acesso somente por meio do CloudFront, execute estas etapas de alto nível:

1. Configure o CloudFront para adicionar um cabeçalho HTTP personalizado às solicitações enviadas ao Application Load Balancer.
2. Configure o Application Load Balancer para encaminhar apenas solicitações que contenham o cabeçalho HTTP personalizado.
3. (Opcional) Exija HTTPS para melhorar a segurança dessa solução.

Para obter mais informações, consulte os tópicos a seguir. Depois de concluir essas etapas, os usuários só podem acessar seu Application Load Balancer por meio do CloudFront.

Tópicos

- [Configurar o CloudFront para adicionar um cabeçalho HTTP personalizado às solicitações](#)
- [Configurar um Application Load Balancer para encaminhar apenas solicitações que contenham um cabeçalho específico](#)
- [\(Opcional\) Melhorar a segurança dessa solução](#)
- [\(Opcional\) Limite o acesso à origem usando a lista de prefixos gerenciada pela AWS para o CloudFront](#)

Configurar o CloudFront para adicionar um cabeçalho HTTP personalizado às solicitações

Você pode configurar o CloudFront para adicionar um cabeçalho HTTP personalizado às solicitações enviadas para sua origem (neste caso, um Application Load Balancer).

Important

Esse caso de uso depende de manter em segredo o nome e o valor do cabeçalho personalizado. Se o nome e o valor do cabeçalho não forem secretos, outros clientes HTTP poderão incluí-los em solicitações enviadas diretamente para o Application Load Balancer. Isso pode fazer com que o Application Load Balancer se comporte como se as solicitações viessem do CloudFront quando não o fizessem. Para evitar isso, mantenha o nome e o valor do cabeçalho personalizado em segredo.

Você pode configurar o CloudFront para adicionar um cabeçalho HTTP personalizado às solicitações de origem com o console do CloudFront, o AWS CloudFormation ou a API do CloudFront.

Para adicionar um cabeçalho HTTP personalizado (console do CloudFront)

No console do CloudFront, use a configuração Origin Custom Headers (Cabeçalhos personalizados de origem) em Origin Settings (Configurações de origem). Insira o nome do cabeçalho e o respectivo valor.

Note

Na produção, use valores e nomes de cabeçalho gerados aleatoriamente. Trate os nomes e os valores do cabeçalho como credenciais seguras; por exemplo, nomes de usuário e senhas.

Você pode editar a configuração Origin Custom Headers (Cabeçalhos personalizados de origem) ao criar ou editar uma origem para uma distribuição existente do CloudFront e ao criar uma nova distribuição. Para ter mais informações, consulte [Atualizar uma distribuição](#) e [Criar uma distribuição](#).

Para adicionar um cabeçalho HTTP personalizado (AWS CloudFormation)

Em um modelo do AWS CloudFormation, use a propriedade `OriginCustomHeaders`, como mostrado no exemplo a seguir.

Note

O nome e o valor do cabeçalho neste exemplo são apenas para demonstração. Na produção, use valores gerados aleatoriamente. Trate o nome e o valor do cabeçalho como uma credencial segura, como um nome do usuário e senha.

```
AWSTemplateFormatVersion: '2010-09-09'
Resources:
  TestDistribution:
    Type: 'AWS::CloudFront::Distribution'
    Properties:
      DistributionConfig:
        Origins:
          - DomainName: app-load-balancer.example.com
            Id: Example-ALB
            CustomOriginConfig:
              OriginProtocolPolicy: https-only
              OriginSSLProtocols:
                - TLSv1.2
            OriginCustomHeaders:
              - HeaderName: X-Custom-Header
                HeaderValue: random-value-1234567890
        Enabled: 'true'
      DefaultCacheBehavior:
        TargetOriginId: Example-ALB
        ViewerProtocolPolicy: allow-all
        CachePolicyId: 658327ea-f89d-4fab-a63d-7e88639e58f6
      PriceClass: PriceClass_All
      ViewerCertificate:
        CloudFrontDefaultCertificate: 'true'
```

Para obter mais informações, consulte as propriedades [Origin](#) e [OriginCustomHeader](#) no Guia do usuário do AWS CloudFormation.

Para adicionar um cabeçalho HTTP personalizado (API do CloudFront)

Na API do CloudFront, use o objeto `CustomHeaders` dentro de `Origin`. Para obter mais informações, consulte [CreateDistribution](#) e [UpdateDistribution](#) na Referência da API do Amazon CloudFront e a documentação do seu SDK ou outro cliente de API.

Há alguns nomes de cabeçalho que não é possível especificar como cabeçalhos personalizados de origem. Para ter mais informações, consulte [Cabeçalhos personalizados que o CloudFront não pode adicionar às solicitações da origem](#).

Configurar um Application Load Balancer para encaminhar apenas solicitações que contenham um cabeçalho específico

Depois de configurar o CloudFront para adicionar um cabeçalho HTTP personalizado às solicitações enviadas ao Application Load Balancer (consulte a [seção anterior](#)), é possível configurar o balanceador de carga para encaminhar apenas solicitações que contenham esse cabeçalho personalizado. Para isso, adicione uma nova regra e modifique a regra padrão no listener de seu balanceador de carga.

Pré-requisitos

Para usar os procedimentos a seguir, você precisa de um Application Load Balancer com pelo menos um listener. Se você ainda não criou um, consulte [Criar um Application Load Balancer](#) no Guia do usuário de Application Load Balancers.

Os procedimentos a seguir modificam um listener HTTPS. Você pode usar o mesmo processo para modificar um listener HTTP.

Para atualizar as regras em um listener do Application Load Balancer

1. Adicione uma nova regra. Use as instruções de [Adicionar uma regra](#) com as seguintes modificações:
 - Adicione a regra ao balanceador de carga que é a origem da distribuição do CloudFront.
 - Em Adicionar condição, escolha o Cabeçalho HTTP. Especifique o nome e o valor do cabeçalho HTTP que você adicionou como um cabeçalho personalizado de origem no CloudFront.
 - Em Adicionar ação, escolha Encaminhar para. Escolha o grupo de destino para o qual deseja encaminhar as solicitações.

2. Edite a regra padrão no receptor do balanceador de carga. Use as instruções de [Editar uma regra](#) com as seguintes modificações:
 - Edite a regra padrão do balanceador de carga que é a origem da distribuição do CloudFront.
 - Exclua a ação padrão e, em Adicionar ação, escolha Retornar resposta fixa.
 - Para Response code (Código de resposta), insira **403**.
 - Para Response body (Corpo de resposta), insira **Access denied**.

Ao concluir essas etapas, o receptor do balanceador de carga terá duas regras. Uma regra encaminha solicitações que contêm o cabeçalho HTTP (solicitações que vêm do CloudFront). A outra regra envia uma resposta fixa a todas as outras solicitações (solicitações que não vêm do CloudFront).

Você pode verificar se a solução funciona enviando uma solicitação para a distribuição do CloudFront e outra para o Application Load Balancer. A solicitação ao CloudFront retorna sua aplicação ou conteúdo da Web e a enviada diretamente ao Application Load Balancer retorna uma resposta 403 com a mensagem de texto simples `Access denied`.

(Opcional) Melhorar a segurança dessa solução

Para melhorar a segurança dessa solução, configure sua distribuição do CloudFront para usar sempre HTTPS ao enviar solicitações ao Application Load Balancer. Lembre-se, essa solução só funciona se você mantiver o nome e o valor do cabeçalho personalizado em segredo. Usar HTTPS pode ajudar a impedir que um bisbilhoteiro descubra o nome e o valor do cabeçalho. Também recomendamos alternar o nome e o valor do cabeçalho periodicamente.

Usar HTTPS para solicitações de origem

Para configurar o CloudFront para usar HTTPS em solicitações de origem, defina a configuração de Origin Protocol Policy (Política de protocolo de origem) como HTTPS Only (Somente HTTPS). Essa configuração está disponível no console do CloudFront, no AWS CloudFormation e na API do CloudFront. Para ter mais informações, consulte [Protocolo \(somente origens personalizadas\)](#).

O seguinte também se aplica ao configurar o CloudFront para usar HTTPS em solicitações de origem:

- Você deve configurar o CloudFront para encaminhar o cabeçalho Host para a origem com a política de solicitação de origem. É possível usar a [política de solicitação de origem gerenciada do AllViewer](#).

- Certifique-se de que o Application Load Balancer tem um receptor HTTPS (conforme mostrado [na seção anterior](#)). Para obter mais informações, consulte [Criar um listener HTTPS](#) no Guia do usuário de Application Load Balancers. Usar um receptor HTTPS requer que você tenha um certificado SSL/TLS que corresponda ao nome de domínio que é roteado para o Application Load Balancer.
- Os certificados SSL/TLS para o CloudFront só podem ser solicitados (ou importados) na Região da AWS us-east-1 no AWS Certificate Manager (ACM). Como o CloudFront é um serviço global, ele distribui automaticamente o certificado da região us-east-1 para todas as regiões associadas à sua distribuição do CloudFront.
 - Por exemplo, se você tiver um Application Load Balancer (ALB) na região ap-southeast-2, deverá configurar certificados SSL/TLS na região ap-southeast-2 (para usar HTTPS entre o CloudFront e a origem do ALB) e na região us-east-1 (para usar HTTPS entre os visualizadores e o CloudFront). Ambos os certificados devem corresponder ao nome de domínio que é encaminhado para o Application Load Balancer. Para ter mais informações, consulte [Região da AWS para AWS Certificate Manager](#).
- Se os usuários finais (também conhecidos como visualizadores ou clientes) da sua aplicação Web puderem usar HTTPS, você também poderá configurar o CloudFront para preferir (ou até mesmo exigir) conexões HTTPS dos usuários finais. Para fazer isso, use a configuração Viewer Protocol Policy (Política de protocolo do visualizador). Você pode ajustá-la para redirecionar usuários finais de HTTP para HTTPS, ou para rejeitar solicitações que usam HTTP. Essa configuração está disponível no console do CloudFront, no AWS CloudFormation e na API do CloudFront. Para ter mais informações, consulte [Política de protocolo do visualizador](#).

Alternar o nome e o valor do cabeçalho

Além de usar HTTPS, também recomendamos alternar o nome e o valor do cabeçalho periodicamente. As etapas de alto nível para fazer isso são as seguintes:

1. Configure o CloudFront para adicionar um cabeçalho HTTP personalizado adicional às solicitações enviadas ao Application Load Balancer.
2. Atualize a regra do listener do Application Load Balancer para encaminhar solicitações que contenham esse cabeçalho HTTP personalizado adicional.
3. Configure o CloudFront para parar de adicionar o cabeçalho HTTP personalizado original às solicitações enviadas ao Application Load Balancer.
4. Atualize a regra do listener do Application Load Balancer para interromper o encaminhamento de solicitações que contenham o cabeçalho HTTP personalizado original.

Para obter mais informações sobre como realizar essas etapas, consulte as seções anteriores.

(Opcional) Limite o acesso à origem usando a lista de prefixos gerenciada pela AWS para o CloudFront

Para restringir ainda mais o acesso ao seu Application Load Balancer, é possível configurar o grupo de segurança associado ao Application Load Balancer para que ele só aceite tráfego do CloudFront quando o serviço estiver usando uma lista de prefixos gerenciada pela AWS. Isso evita que o tráfego que não se origina do CloudFront chegue ao seu Application Load Balancer na camada de rede (camada 3) ou na camada de transporte (camada 4).

Para ter mais informações, consulte a postagem do blog [Limitar o acesso às suas origens usando a lista de prefixos gerenciada pela AWS do Amazon CloudFront](#).

Restringir a distribuição geográfica do conteúdo

É possível usar restrições geográficas, também conhecidas como bloqueios geográficos, para impedir que usuários em localizações geográficas específicas acessem o conteúdo que você está distribuindo por meio de uma distribuição do Amazon CloudFront. Para usar restrições geográficas, você tem duas opções:

- Use o recurso de restrições geográficas do CloudFront. Use essa opção para restringir o acesso a todos os arquivos associados a uma distribuição e restringir o acesso no nível do país.
- Usar um serviço de geolocalização de terceiros. Use essa opção para restringir o acesso a um subconjunto de arquivos associados a uma distribuição ou restringir o acesso a uma granularidade mais específica no nível do país.

Tópicos

- [Usar restrições geográficas do CloudFront](#)
- [Usar um serviço de geolocalização de terceiros](#)

Usar restrições geográficas do CloudFront

Quando um usuário solicita seu conteúdo, o CloudFront normalmente o fornece, independentemente de onde o usuário está localizado. Se precisar impedir que usuários de alguns países acessem seu

conteúdo, você poderá usar o recurso de restrições geográficas do CloudFront para executar uma das seguintes ações:

- Conceder aos usuários acesso ao conteúdo somente se estiverem em um dos países aprovados na lista de permissões.
- Impedir que os usuários acessem o conteúdo se estiverem em um dos países proibidos na lista de bloqueio.

Por exemplo, se a solicitação for proveniente de um país em que você não estiver autorizado a distribuir o conteúdo, poderá usar as restrições geográficas do CloudFront para bloquear a solicitação.

Note

O CloudFront determina a localização de seus usuários usando um banco de dados de terceiros. A precisão do mapeamento entre os endereços IP e os países varia de acordo com a região. Com base em testes recentes, a precisão geral é de 99,8%. Se o CloudFront não conseguir determinar a localização de um usuário, ele fornecerá o conteúdo solicitado pelo usuário.

Veja como as restrições geográficas funcionam:

1. Imagine que você tenha direitos para distribuir o conteúdo apenas em Liechtenstein. Você deve atualizar a distribuição do CloudFront e adicionar uma lista de permissões que contenha somente Liechtenstein. (Ou é possível adicionar uma lista de bloqueio com todos os países, exceto Liechtenstein.)
2. Um usuário em Munique solicita seu conteúdo, e o DNS encaminha a solicitação ao local de borda do CloudFront em Milão, na Itália.
3. O local da borda em Milão procura sua distribuição e determina que o usuário em Munique não pode baixar seu conteúdo.
4. O CloudFront retorna um código de status HTTP 403 (Forbidden) ao usuário.

Opcionalmente, você pode configurar o CloudFront para retornar uma mensagem de erro personalizada para o usuário e especificar por quanto tempo o CloudFront deve armazenar a resposta de erro em cache para o arquivo solicitado. O valor de padrão é de 10 segundos. Para obter

mais informações, consulte [Criar uma página de erro personalizada para códigos de status HTTP específicos](#).

As restrições geográficas se aplicam a toda uma distribuição. Se precisar aplicar uma restrição a parte de seu conteúdo e uma restrição diferente (ou nenhuma) a outra parte, você deverá criar distribuições distintas do CloudFront ou [usar um serviço de geolocalização de terceiros](#).

Se você habilitar [logs padrão](#) (logs de acesso) do CloudFront, poderá identificar as solicitações recusadas pelo CloudFront procurando as entradas de log nas quais o valor de `sc-status` (o código de status HTTP) for 403. No entanto, se usar somente os logs padrão, não será possível distinguir uma solicitação recusada pelo CloudFront com base na localização do usuário de uma solicitação recusada pelo CloudFront porque o usuário não tem permissão para acessar o arquivo por outro motivo. Se você tiver um serviço de geolocalização de terceiros, como Digital Element ou MaxMind, poderá identificar a localização das solicitações com base no endereço IP da coluna `c-ip` (IP do cliente) nos logs de acesso. Para obter mais informações sobre os logs padrão do CloudFront, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

O procedimento a seguir explica como usar o console do CloudFront para adicionar restrições geográficas a uma distribuição existente. Para obter informações sobre como usar o console para criar uma distribuição, consulte [Criar uma distribuição](#).

Como adicionar restrições geográficas à distribuição na Web do CloudFront (console)

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Distribuições e selecione a distribuição que deseja atualizar.
3. Escolha a guia Segurança e selecione Restrições geográficas.
4. Escolha Edit (Editar).
5. Selecione Allow list (Lista de permissões), para criar uma lista de países permitidos, ou Block list (Lista de bloqueio), para criar uma lista de países bloqueados.
6. Adicione os países desejados à lista e escolha Save changes (Salvar as alterações).

Usar um serviço de geolocalização de terceiros

Com o recurso de restrições geográficas do CloudFront, é possível controlar a distribuição do conteúdo em nível de país para todos os arquivos que estão sendo distribuídos com uma distribuição da web específica. Se tiver um caso de uso para restrições geográficas em que as restrições não

sigam limites de país, ou se quiser restringir o acesso a apenas alguns arquivos distribuídos por determinada distribuição, você poderá combinar o CloudFront com um serviço de geolocalização de terceiros. Isso confere a você controle do conteúdo com base não apenas no país, mas também na cidade, no CEP ou no código postal, ou até mesmo na latitude e na longitude.

Se você usar um serviço de geolocalização de terceiros, recomendamos usar URLs assinados do CloudFront, que permitem especificar uma data e hora de expiração do URL. Além disso, recomendamos que você use um bucket do Amazon S3 como origem. Isso possibilita usar um [controle de acesso à origem](#) do CloudFront para impedir que os usuários acessem seu conteúdo diretamente da origem. Para obter mais informações sobre URLs assinados e controle de acesso à origem, consulte [Veicular conteúdo privado com URLs e cookies assinados](#).

As etapas a seguir explicam como controlar o acesso aos seus arquivos usando um serviço de geolocalização de terceiros.

Como usar um serviço de geolocalização de terceiros para restringir o acesso a arquivos em uma distribuição do CloudFront

1. Obtenha uma conta com um serviço de geolocalização.
2. Faça upload do conteúdo em um bucket do Amazon S3.
3. Configure o Amazon CloudFront e o Amazon S3 para fornecer conteúdo privado. Para obter mais informações, consulte [Veicular conteúdo privado com URLs e cookies assinados](#).
4. Configure seu aplicativo web para:
 - Envie o endereço IP de cada solicitação de usuário para o serviço de geolocalização.
 - Avalie o valor de retorno do serviço de geolocalização para determinar se o usuário está em um local no qual você deseja que o CloudFront distribua seu conteúdo.
 - Se você quiser distribuir o conteúdo para a localização do usuário, gere um URL assinado para o conteúdo do CloudFront. Se não quiser distribuir o conteúdo para essa localização, retorne o código de status HTTP 403 (Forbidden) para o usuário. É também possível configurar o CloudFront para retornar uma mensagem de erro personalizada. Para obter mais informações, consulte [the section called “Criar uma página de erro personalizada para códigos de status HTTP específicos”](#).

Para obter mais informações, consulte a documentação do serviço de geolocalização que você estiver usando.

Você pode usar uma variável do servidor da Web para obter os endereços IP dos usuários que estiverem acessando seu site. Observe as seguintes advertências:

- Se o servidor da Web não estiver conectado à Internet por um balanceador de carga, você poderá usar uma variável de servidor da Web para obter o endereço IP remoto. No entanto, esse endereço IP nem sempre é o endereço IP do usuário. Ele pode ser o endereço IP de um servidor de proxy, dependendo de como o usuário está conectado à Internet.
- Se o servidor da web estiver conectado à Internet por um load balancer, uma variável dele poderá conter o endereço IP do load balancer, não do usuário. Nessa configuração, recomendamos que você use o último endereço IP do cabeçalho HTTP `X-Forwarded-For`. Esse cabeçalho normalmente contém mais de um endereço IP, e a maioria deles são para proxies ou balanceadores de carga. O último endereço IP da lista é o que tem maior probabilidade de estar associado à localização geográfica do usuário.

Se o servidor da Web não estiver conectado a um balanceador de carga, recomendamos que você use variáveis de servidor da Web, em vez do cabeçalho `X-Forwarded-For`, para evitar a falsificação do endereço IP.

Use criptografia de nível de campo para ajudar a proteger dados confidenciais

Com o Amazon CloudFront, é possível impor conexões seguras de ponta a ponta a servidores de origem usando HTTPS. A criptografia no nível de campo acrescenta uma camada adicional de segurança que permite proteger dados específicos em todo o processamento do sistema, de modo que apenas alguns aplicativos possam vê-los.

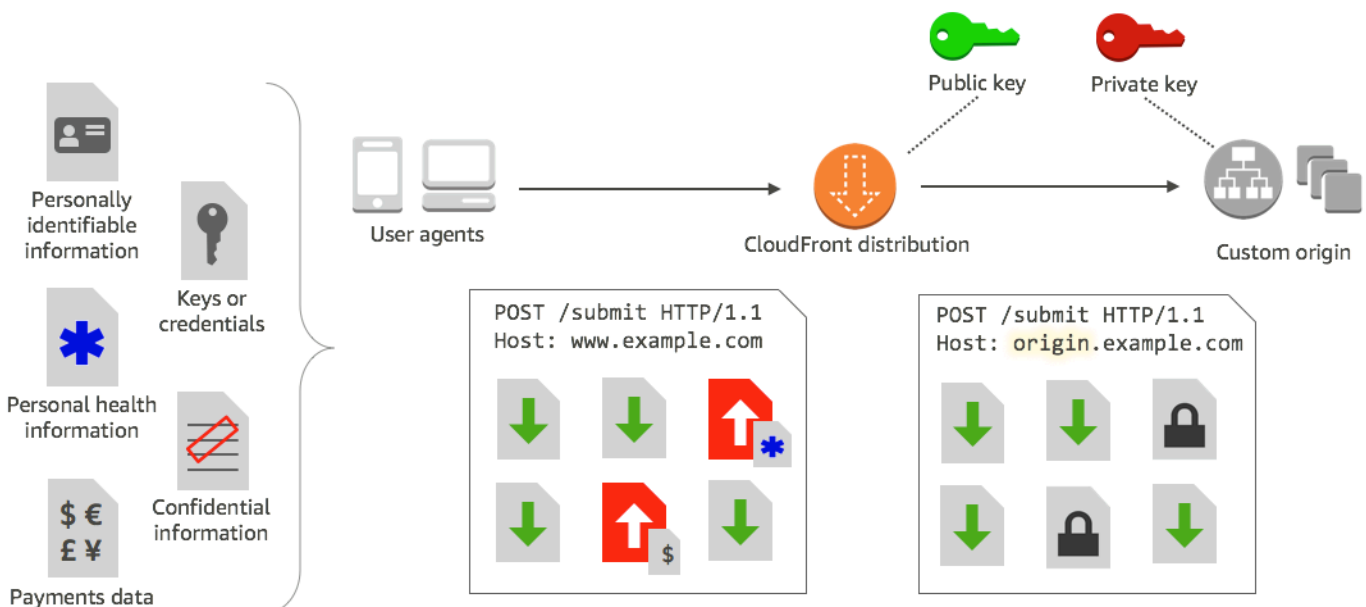
A criptografia no nível de campo habilita permitir que usuários faça upload de informações confidenciais com segurança nos servidores web. As informações confidenciais fornecidas pelos usuários são criptografadas na borda, próximo ao usuário, e permanecem criptografadas em toda a pilha de aplicativos. Essa criptografia garante que somente os aplicativos que precisam dos dados e que têm as credenciais para descriptografá-los possam fazê-lo.

Para usar a criptografia no nível de campo, ao configurar a distribuição do CloudFront, especifique o conjunto de campos nas solicitações POST que você quer que sejam criptografados e a chave pública a ser usada para criptografá-los. Você pode criptografar até 10 campos de dados em uma solicitação. (Não é possível criptografar todos os dados em uma solicitação com criptografia no nível de campo; é preciso especificar campos individuais para criptografar.)

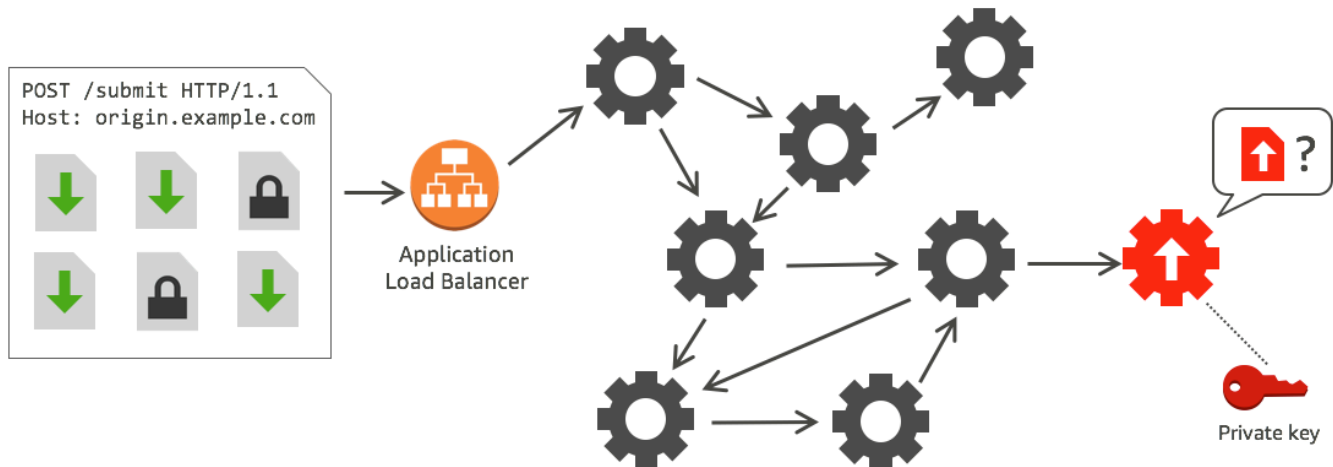
Quando a solicitação HTTPS com criptografia no nível de campo é encaminhada para a origem, e a solicitação é roteada em todo o subsistema ou aplicativo de origem, os dados confidenciais ainda são criptografados, reduzindo o risco de uma violação dos dados ou da perda acidental de dados confidenciais. Os componentes que precisam acessar os dados confidenciais por motivos comerciais, como um sistema de processamento de pagamento que precisa de acesso a um número de crédito, podem usar a chave privada apropriada para descriptografar e acessar os dados.

Note

Para usar a criptografia no nível de campo, a origem deve oferecer suporte à codificação em partes.



A criptografia no nível de campo do CloudFront usa criptografia assimétrica, também conhecida como criptografia de chave pública. Basta você fornecer uma chave pública para o CloudFront e todos os dados confidenciais especificados são criptografados automaticamente. A chave que você fornece ao CloudFront não pode ser usada para descriptografar os valores criptografados; somente sua chave privada poderá fazer isso.



Tópicos

- [Visão geral da criptografia no nível de campo](#)
- [Configurar a criptografia em nível de campo](#)
- [Descriptografar campos de dados na origem](#)

Visão geral da criptografia no nível de campo

As etapas a seguir fornecem uma visão geral de como configurar a criptografia no nível de campo. Para obter as etapas específicas, consulte [Configurar a criptografia em nível de campo](#).

1. Obtenha um par de chaves pública e privada. É necessário obter e adicionar a chave pública antes de iniciar a configuração da criptografia no nível de campo do CloudFront.
2. Crie um perfil de criptografia no nível de campo. Os perfis de criptografia no nível de campo, que são criados no CloudFront, definem os campos que você deseja criptografar.
3. Crie uma configuração de criptografia no nível de campo. A configuração especifica os perfis a serem usados, com base no tipo de conteúdo da solicitação ou em um argumento de consulta, para criptografar campos de dados específicos. Também é possível escolher as opções de comportamento de encaminhamento de solicitações desejadas para diferentes situações. Por exemplo, você pode definir o comportamento para quando o nome do perfil especificado pelo argumento da consulta de um URL de solicitação não existir no CloudFront.
4. Vincule um comportamento de cache. Vincule a configuração a um comportamento de cache de uma distribuição para especificar quando o CloudFront deverá criptografar os dados.

Configurar a criptografia em nível de campo

Siga essas etapas para começar a usar a criptografia no nível de campo. Para saber mais sobre cotas (anteriormente conhecidas como limites) na criptografia em nível de campo, consulte [Cotas](#).

- [Etapa 1: Criar um par de chaves do RSA](#)
- [Etapa 2: Adicionar uma chave pública ao CloudFront](#)
- [Etapa 3: Criar um perfil de criptografia no nível de campo](#)
- [Etapa 4: Criar uma configuração](#)
- [Etapa 5: Adicionar uma configuração ao comportamento de cache](#)

Etapa 1: Criar um par de chaves do RSA

Para começar a usar, é necessário criar um par de chaves RSA que inclua uma chave pública e uma chave privada. A chave pública permite que o CloudFront criptografe dados e a chave privada permite que os componentes na origem descriptografem os campos que foram criptografados. Você pode usar o OpenSSL ou outra ferramenta para criar um par de chaves. O tamanho da chave deve ser de 2048 bits.

Por exemplo, se estiver usando OpenSSL, poderá usar o seguinte comando para gerar um par de chaves de 2.048 bits e salvá-lo no arquivo `private_key.pem`:

```
openssl genrsa -out private_key.pem 2048
```

O arquivo resultante contém a chave pública e a privada. Para extrair a chave pública do arquivo, execute o seguinte comando:

```
openssl rsa -pubout -in private_key.pem -out public_key.pem
```

O arquivo de chave pública (`public_key.pem`) contém o valor de chave codificada que você colar na etapa seguinte.

Etapa 2: Adicionar uma chave pública ao CloudFront

Após obter o par de chaves RSA, adicione a sua chave pública ao CloudFront.

Como adicionar sua chave pública ao CloudFront (console)

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Public key.
3. Escolha Add public key (Adicionar chave pública).
4. Em Key name, digite um nome exclusivo para a chave. O nome não pode conter espaços e só pode incluir caracteres alfanuméricos, sublinhados (_) e hífen (-). O número máximo de caracteres é 128.
5. Em Key value (Valor de chave), cole o valor da chave pública codificada, incluindo as linhas -----BEGIN PUBLIC KEY----- e -----END PUBLIC KEY-----.
6. Em Comment, adicione um comentário opcional. Por exemplo, você pode incluir a data de expiração para a chave pública.
7. Escolha Adicionar.

É possível adicionar mais chaves para uso com o CloudFront repetindo as etapas desse procedimento.

Etapa 3: Criar um perfil de criptografia no nível de campo

Depois de adicionar pelo menos uma chave pública ao CloudFront, crie um perfil que informe ao CloudFront quais campos serão criptografados.

Para criar um perfil de criptografia no nível de campo (console)

1. No painel de navegação, escolha Field-level encryption.
2. Escolha Create profile (Criar perfil).
3. Preencha os seguintes campos:

Profile name

Digite um nome exclusivo para o perfil. O nome não pode conter espaços e só pode incluir caracteres alfanuméricos, sublinhados (_) e hífen (-). O número máximo de caracteres é 128.

Nome da chave pública

Na lista suspensa, escolha o nome de uma chave pública que você adicionou ao CloudFront na etapa 2. O CloudFront usa a chave para criptografar os campos especificados neste perfil.

Nome do provedor

Digite uma frase que ajude a identificar a chave, como o provedor que forneceu o par de chaves. Essas informações, juntamente com a chave privada, são necessárias quando os aplicativos descriptografam os campos de dados. O nome do provedor não pode conter espaços e só pode incluir caracteres alfanuméricos, dois pontos (:), sublinhados (_) e hífen (-). O número máximo de caracteres é 128.

Padrão do nome do campo para correspondência

Digite os nomes dos campos de dados, ou padrões que identifiquem nomes de campos de dados na solicitação, a serem criptografados pelo CloudFront. Escolha a opção + para adicionar todos os campos que você deseja criptografar com essa chave.

Para o padrão do nome de campo, você pode digitar o nome inteiro do campo de dados, como DateOfBirth ou apenas a primeira parte do nome seguido por um caractere curinga (*), como CreditCard*. O padrão do nome de campo deve incluir apenas caracteres alfanuméricos, colchetes ([e]), pontos (.), sublinhados (_) e hífen (-), além do caractere opcional curinga (*).

Certifique-se de que você não está usando caracteres sobrepostos para padrões diferentes de nomes de campos. Por exemplo, se você tiver o padrão de nome de campo ABC*, não poderá adicionar um outro padrão de nome de campo que seja AB*. Além disso, os nomes dos campos diferenciam maiúsculas de minúsculas, e o número máximo de caracteres que podem ser usados é 128.

Comentário

(Opcional) Digite um comentário sobre este perfil. O número máximo de caracteres que você pode usar é 128.

4. Após preencher os campos, escolha Create profile (Criar perfil).
5. Se você deseja adicionar mais perfis, escolha Add profile.

Etapa 4: Criar uma configuração

Depois de criar um ou mais perfis de criptografia no nível de campo, crie uma configuração para especificar o tipo de conteúdo da solicitação que inclua os dados a serem criptografados, o perfil a ser usado na criptografia e outras opções especificando como você deseja que o CloudFront processe a criptografia.

Por exemplo, quando o CloudFront não puder criptografar os dados, você poderá especificar se ele deve bloquear ou encaminhar uma solicitação para a origem nos seguintes cenários:

- Quando o tipo de conteúdo de uma solicitação não estiver em uma configuração: se não tiver adicionado um tipo de conteúdo a uma configuração, você poderá especificar se o CloudFront deve encaminhar a solicitação com esse tipo de conteúdo para a origem sem criptografar os campos de dados ou bloquear a solicitação e retornar um erro.

Note

Se você adicionar um tipo de conteúdo a uma configuração, mas não tiver especificado um perfil para usar com esse tipo, o CloudFront sempre encaminhará as solicitações com esse tipo de conteúdo para a origem.

- Quando o nome de perfil fornecido em um argumento de consulta for desconhecido: ao especificar o argumento de consulta `fle-profile` com um nome de perfil que não existe para a distribuição, você poderá especificar se o CloudFront deve enviar a solicitação para a origem sem criptografar os campos de dados ou bloquear a solicitação e retornar um erro.

Em uma configuração, você também pode especificar se um perfil fornecido como um argumento de consulta em uma URL poderá substituir o perfil que foi mapeado para o tipo de conteúdo dessa consulta. Por padrão, o CloudFront usará o perfil que você mapeou para um tipo de conteúdo, se houver um perfil especificado. Isso permite que você tenha um perfil que é usado por padrão, mas que possa decidir, em determinadas solicitações, que deseja impor um perfil diferente.

Assim, por exemplo, é possível especificar (em sua configuração) **SampleProfile** como o perfil do argumento de consulta a ser usado. Depois, você poderá usar o URL `https://d1234.cloudfront.net?fle-profile=SampleProfile` em vez de `https://d1234.cloudfront.net`, para que o CloudFront use **SampleProfile** para essa solicitação, em vez do perfil configurado para o tipo de conteúdo da solicitação.

Você pode criar até 10 configurações para uma única conta e, em seguida, associar uma das configurações ao comportamento de cache de qualquer distribuição da conta.

Para criar uma configuração de criptografia no nível de campo (console)

1. Na página Field-level encryption (Criptografia de nível de campo), escolha Create configuration (Criar configuração).

Observação: se você não tiver criado pelo menos um perfil, não verá a opção para criar uma configuração.

2. Preencha os seguintes campos para especificar o perfil a ser usado. (Alguns campos não podem ser alterados.)

Tipo de conteúdo (não pode ser alterado)

O tipo do conteúdo é definido como `application/x-www-form-urlencoded` e não pode ser alterado.

ID do perfil padrão (opcional)

Na lista suspensa, escolha o perfil que você deseja mapear para o tipo de conteúdo no campo Content type (Tipo de conteúdo).

Formato do conteúdo (não pode ser alterado)

O formato do conteúdo é definido como `URLencoded` e não pode ser alterado.

3. Se quiser alterar o comportamento padrão do CloudFront para as opções a seguir, marque a caixa de seleção apropriada.

Encaminhar solicitação para a origem quando o tipo de conteúdo da solicitação não está configurado

Marque esta caixa de seleção se você quiser permitir que a solicitação vá para a origem caso você não tenha especificado um perfil a ser usado para o tipo de conteúdo da solicitação.

Substituir perfil de um tipo de conteúdo por um argumento de consulta fornecido

Marque esta caixa de seleção se você quiser permitir que um perfil fornecido em um argumento de consulta substitua o perfil que você especificou para um tipo de conteúdo.

4. Se você marcar a caixa de seleção permitindo que um argumento de consulta substitua o perfil padrão, deverá preencher os seguintes campos adicionais para a configuração. É possível criar até cinco desses mapeamentos de argumento de consulta para usar com as consultas.

Argumento de consulta

Digite o valor que deseja incluir nas URLs para o argumento de consulta `fle-profile`. Esse valor informa o CloudFront para usar o ID do perfil (que você especificará no próximo campo) associado a esse argumento de consulta para a criptografia no nível do campo para essa consulta.

O número máximo de caracteres que você pode usar é 128. O valor não pode incluir espaços e deve usar apenas caracteres alfanuméricos, traço (-), ponto (.), sublinhado (_), asterisco (*), sinal de mais (+) e porcentagem (%).

ID do perfil

Na lista suspensa, escolha o perfil que deseja associar ao valor que você digitou em Query argument (Argumento de consulta).

Encaminhar solicitação para a origem quando o perfil especificado em um argumento de consulta não existe

Marque esta caixa de seleção se você quiser permitir que a solicitação vá para a origem caso o perfil especificado em um argumento de consulta não esteja definido no CloudFront.

Etapa 5: Adicionar uma configuração ao comportamento de cache

Para usar a criptografia no nível de campo, vincule uma configuração ao comportamento de cache de uma distribuição adicionando o ID da configuração como um valor para a sua distribuição.

Important

Para vincular uma configuração de criptografia em nível de campo a um comportamento de cache, a distribuição deve ser configurada para sempre usar HTTPS e aceitar solicitações HTTP POST e PUT de visualizadores. Isso significa que:

- A opção Viewer Protocol Policy (Política de protocolo do visualizador) do comportamento de cache deve ser definida como Redirect HTTP to HTTPS (Redirecionar HTTP para HTTPS) ou HTTPS Only (Somente HTTPS). (No AWS CloudFormation ou na API do CloudFront, `ViewerProtocolPolicy` deve ser definida como `redirect-to-https` ou `https-only`.)

- A opção Allowed HTTP Methods (Métodos HTTP permitidos) do comportamento de cache deve ser definida como GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE. (No AWS CloudFormation ou na API do CloudFront, AllowedMethods deve ser definido como GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE. É possível defini-los em qualquer ordem.)
- A opção Origin Protocol Policy (Política de protocolo de origem) deve ser definida como Match Viewer (Corresponder ao visualizador) ou HTTPS Only (Somente HTTPS). (No AWS CloudFormation ou na API do CloudFront, OriginProtocolPolicy deve ser definida como match-viewer ou https-only.)

Para ter mais informações, consulte [Referência de configurações da distribuição](#).

Descriptografar campos de dados na origem

O CloudFront criptografa os campos de dados usando o [AWS Encryption SDK](#). Os dados permanecem criptografados em toda a pilha de aplicativos e podem ser acessados somente por aplicativos que tenham as credenciais necessárias para descriptografá-los.

Após a criptografia, o texto cifrado é codificado em base64. Quando as suas aplicações descriptografam o texto na origem, primeiro devem decodificar o texto cifrado e, depois, usar o SDK de criptografia da AWS para descriptografar os dados.

O exemplo de código a seguir ilustra como os aplicativos podem descriptografar dados na origem. Observe o seguinte:

- Para simplificar, este exemplo carrega chaves públicas e privadas (em formato DER) a partir de arquivos do diretório de trabalho. Na prática, você pode armazenar a chave privada em um local offline seguro, como um módulo de segurança de hardware offline, e distribuir a chave pública para a equipe de desenvolvimento.
- O CloudFront usa informações específicas ao criptografar os dados, e o mesmo conjunto de parâmetros deve ser usado na origem para descriptografá-los. Os parâmetros que o CloudFront usa ao inicializar a MasterKey incluem:
 - PROVIDER_NAME: você especificou esse valor quando criou um perfil de criptografia no nível de campo. Use o mesmo valor aqui.
 - KEY_NAME: você criou um nome para sua chave pública quando fez upload para o CloudFront e especificou o nome da chave no perfil. Use o mesmo valor aqui.

- ALGORITMO: o CloudFront usa RSA/ECB/OAEPWithSHA-256AndMGF1Padding como o algoritmo para criptografia e, portanto, o mesmo algoritmo deve ser usado para descriptografar os dados.
- Se você executar o seguinte programa de exemplo com um texto cifrado como entrada, a saída dos dados descriptografados será exibida em seu console. Para obter mais informações, consulte o [Código de exemplo em Java](#) no SDK de criptografia da AWS.

Código de exemplo

```
import java.nio.file.Files;
import java.nio.file.Paths;
import java.security.KeyFactory;
import java.security.PrivateKey;
import java.security.PublicKey;
import java.security.spec.PKCS8EncodedKeySpec;
import java.security.spec.X509EncodedKeySpec;

import org.apache.commons.codec.binary.Base64;

import com.amazonaws.encryptionsdk.AwsCrypto;
import com.amazonaws.encryptionsdk.CryptoResult;
import com.amazonaws.encryptionsdk.jce.JceMasterKey;

/**
 * Sample example of decrypting data that has been encrypted by CloudFront field-level
 * encryption.
 */
public class DecryptExample {

    private static final String PRIVATE_KEY_FILENAME = "private_key.der";
    private static final String PUBLIC_KEY_FILENAME = "public_key.der";
    private static PublicKey publicKey;
    private static PrivateKey privateKey;

    // CloudFront uses the following values to encrypt data, and your origin must use
    // same values to decrypt it.
    // In your own code, for PROVIDER_NAME, use the provider name that you specified
    // when you created your field-level
    // encryption profile. This sample uses 'DEMO' for the value.
    private static final String PROVIDER_NAME = "DEMO";
```



```
// In your own code, use the key name that you specified when you added your public
key to CloudFront. This sample
// uses 'DEMOKEY' for the key name.
private static final String KEY_NAME = "DEMOKEY";
// CloudFront uses this algorithm when encrypting data.
private static final String ALGORITHM = "RSA/ECB/OAEPWithSHA-256AndMGF1Padding";

public static void main(final String[] args) throws Exception {

    final String dataToDecrypt = args[0];

    // This sample uses files to get public and private keys.
    // In practice, you should distribute the public key and save the private key
in secure storage.
    populateKeyPair();

    System.out.println(decrypt(debase64(dataToDecrypt)));
}

private static String decrypt(final byte[] bytesToDecrypt) throws Exception {
    // You can decrypt the stream only by using the private key.

    // 1. Instantiate the SDK
    final AwsCrypto crypto = new AwsCrypto();

    // 2. Instantiate a JCE master key
    final JceMasterKey masterKey = JceMasterKey.getInstance(
        publicKey,
        privateKey,
        PROVIDER_NAME,
        KEY_NAME,
        ALGORITHM);

    // 3. Decrypt the data
    final CryptoResult <byte[], ? > result = crypto.decryptData(masterKey,
bytesToDecrypt);
    return new String(result.getResult());
}

// Function to decode base64 cipher text.
private static byte[] debase64(final String value) {
    return Base64.decodeBase64(value.getBytes());
}
```

```
private static void populateKeyPair() throws Exception {
    final byte[] PublicKeyBytes =
Files.readAllBytes(Paths.get(PUBLIC_KEY_FILENAME));
    final byte[] privateKeyBytes =
Files.readAllBytes(Paths.get(PRIVATE_KEY_FILENAME));
    publicKey = KeyFactory.getInstance("RSA").generatePublic(new
X509EncodedKeySpec(PublicKeyBytes));
    privateKey = KeyFactory.getInstance("RSA").generatePrivate(new
PKCS8EncodedKeySpec(privateKeyBytes));
}
}
```

Vídeo sob demanda e vídeo de transmissão ao vivo com o CloudFront

É possível usar o CloudFront para fornecer vídeo sob demanda (VOD) ou streaming de vídeo ao vivo usando qualquer origem HTTP. Uma forma de configurar fluxos de trabalho de vídeo na nuvem é usar o CloudFront em conjunto com os [AWS Media Services](#).

Tópicos

- [Sobre o streaming de vídeo](#)
- [Fornecer vídeo sob demanda com o CloudFront](#)
- [Fornecer streaming de vídeo ao vivo com o CloudFront e o AWS Media Services](#)

Sobre o streaming de vídeo

É necessário usar um codificador para empacotar conteúdo de vídeo para que o CloudFront possa distribuir o conteúdo. O processo de empacotamento cria segmentos que apresentam o conteúdo de áudio, vídeo e legendas. Ele também gera arquivos de manifesto, que descrevem em uma ordem específica quais segmentos reproduzir e quando. Os formatos comuns para pacotes são MPEG DASH, Apple HLS, Microsoft Smooth Streaming e CMAF.

Transmissão de VOD

Para streaming de VOD, o conteúdo do vídeo é armazenado em um servidor e os visualizadores podem assisti-lo a qualquer momento. Para criar um ativo que os visualizadores possam transmitir, use um codificador, como [AWS Elemental MediaConvert](#), para formatar e empacotar seus arquivos de mídia.

Quando o vídeo estiver empacotado nos formatos corretos, você poderá armazená-lo em um servidor ou em um bucket do Amazon S3 e, depois, enviá-lo com o CloudFront conforme solicitado pelos visualizadores.

Streaming de vídeo ao vivo

Para streaming de vídeo ao vivo, seu conteúdo de vídeo é transmitido em tempo real à medida que os eventos ao vivo acontecem ou é configurado como um canal ao vivo transmitido 24 horas por dia, 7 dias por semana. Para criar saídas ao vivo para transmissão e entrega de streaming,

use um codificador como o AWS Elemental MediaLive para compactar o vídeo e formatá-lo para dispositivos de visualização.

Depois que o vídeo é codificado, você pode enviá-lo ao AWS Elemental MediaStore ou convertê-lo em diferentes formatos de entrega usando o AWS Elemental MediaPackage. Use uma dessas origens para configurar uma distribuição do CloudFront para fornecer o conteúdo. Para etapas específicas e orientações para a criação de distribuições que funcionam em conjunto com esses serviços, consulte [Distribuir vídeo usando o AWS Elemental MediaStore como origem](#) e [Veicular vídeo ao vivo formatado com o AWS Elemental MediaPackage](#).

O Wowza e o Unified Streaming também fornecem ferramentas que podem ser usadas para streaming de vídeo com o CloudFront. Para obter mais informações sobre como usar o Wowza com o CloudFront, consulte [Bring your Wowza Streaming Engine license to CloudFront live HTTP streaming](#), no site da documentação do Wowza. Para obter informações sobre como usar o Unified Streaming com o CloudFront para streaming de VOD, consulte [CloudFront](#) no site de documentação do Unified Streaming.

Fornecer vídeo sob demanda com o CloudFront

Para entregar transmissão de vídeo sob demanda (VOD) com o CloudFront, use os seguintes serviços:

- O Amazon S3 para armazenar o conteúdo em seu formato original e para armazenar o vídeo transcodificado.
- Um codificador (como AWS Elemental MediaConvert) para transcodificar o vídeo em formatos de streaming.
- O CloudFront para entregar o vídeo transcodificado aos visualizadores. Para o Microsoft Smooth Streaming, consulte [Configurar vídeo sob demanda para o Microsoft Smooth Streaming](#).

Para criar uma solução de VOD com o CloudFront

1. Faça upload do conteúdo em um bucket do Amazon S3. Para saber mais sobre como trabalhar com o Amazon S3, consulte [o Manual do usuário do Amazon Simple Storage Service](#).
2. Transcodifique seu conteúdo usando um trabalho do MediaConvert. O trabalho converte seu vídeo nos formatos exigidos pelos aparelhos de vídeo que seus visualizadores usam. Você também pode usar o trabalho para criar ativos que variam em resolução e taxa de bits. Esses

ativos são usados para streaming de taxa de bits adaptável (ABR), que ajusta a qualidade da visualização dependendo da largura de banda disponível para o visualizador. O MediaConvert armazena o vídeo transcodificado em um bucket do S3.

3. Entregue o conteúdo convertido usando uma distribuição do CloudFront. Os visualizadores podem ver o conteúdo em qualquer dispositivo, a qualquer momento.

Tip

É possível explorar como usar um modelo do AWS CloudFormation para implantar uma solução de VOD da AWS em conjunto com todos os componentes associados. Para ver as etapas de uso do modelo, consulte [Implantação automatizada](#) no guia Vídeo sob demanda na AWS.

Configurar vídeo sob demanda para o Microsoft Smooth Streaming

Você tem as seguintes opções para usar o CloudFront para distribuir conteúdo de vídeo sob demanda (VOD) que transcodificou para o formato Microsoft Smooth Streaming:

- Especifique um servidor Web que execute o Microsoft IIS e dê suporte ao Smooth Streaming como origem para sua distribuição.
- Habilite o Smooth Streaming nos comportamentos de cache de uma distribuição do CloudFront. Como você pode usar vários comportamentos de cache em uma distribuição, é possível usar uma distribuição para arquivos de mídia Smooth Streaming, bem como outro conteúdo.

Important

Se você especificar um servidor Web executando o Microsoft IIS como sua origem, não habilite o Smooth Streaming nos comportamentos de cache da distribuição do CloudFront. O CloudFront não poderá usar um servidor Microsoft IIS como origem se você habilitar o Smooth Streaming como um comportamento de cache.

Se você ativar o Smooth Streaming em um comportamento de cache (ou seja, se você não tiver um servidor que execute o Microsoft IIS), observe o seguinte:

- Você poderá continuar distribuindo outros conteúdos usando o mesmo comportamento de cache se o conteúdo for correspondente ao valor de Path Pattern desse comportamento de cache.
- O CloudFront pode usar um bucket do Amazon S3 ou uma origem personalizada para arquivos de mídia do Smooth Streaming. O CloudFront não poderá usar um servidor Microsoft IIS como origem se você habilitar o Smooth Streaming para o comportamento de cache.
- Arquivos de mídia no formato Smooth Streaming não podem ser invalidados. Se você quiser atualizar os arquivos antes de eles expirarem, renomeie-os. Para ter mais informações, consulte [Adicionar, remover ou substituir conteúdo distribuído pelo CloudFront](#).

Para obter informações sobre clientes Smooth Streaming, consulte [Smooth Streaming](#) no site de documentação da Microsoft.

Como usar o CloudFront para distribuir arquivos do Smooth Streaming quando um servidor Web do Microsoft IIS não é a origem

1. Transcodifique seus arquivos de mídia para o formato MP4 fragmentado do Smooth Streaming.
2. Execute um destes procedimentos:
 - Se estiver usando o console do CloudFront: ao criar ou atualizar uma distribuição, habilite o Smooth Streaming em um ou mais comportamentos de cache da distribuição.
 - Se estiver usando a API do CloudFront: adicione o elemento `SmoothStreaming` ao tipo complexo `DistributionConfig` para um ou mais comportamentos de cache da distribuição.
3. Carregue os arquivos do Smooth Streaming para sua origem.
4. Crie um arquivo `clientaccesspolicy.xml` ou `crossdomainpolicy.xml` e adicione-o a um local acessível na raiz de sua distribuição, por exemplo, `https://d111111abcdef8.cloudfront.net/clientaccesspolicy.xml`. Veja abaixo um exemplo de política:

```
<?xml version="1.0" encoding="utf-8"?>
<access-policy>
<cross-domain-access>
<policy>
<allow-from http-request-headers="*">
<domain uri="*" />
</allow-from>
<grant-to>
```

```
<resource path="/" include-subpaths="true"/>
</grant-to>
</policy>
</cross-domain-access>
</access-policy>
```

Para obter mais informações, consulte [Disponibilizar um serviço para além dos limites do domínio](#) no site Microsoft Developer Network.

5. Para links em seu aplicativo (por exemplo, um media player), especifique o URL do arquivo de mídia no seguinte formato padrão:

```
https://d111111abcdef8.cloudfront.net/video/presentation.ism/Manifest
```

Fornecer streaming de vídeo ao vivo com o CloudFront e o AWS Media Services

Para usar o AWS Media Services com o CloudFront para entregar conteúdo ao vivo para um público global, consulte as orientações a seguir.

Use [AWS Elemental MediaLive](#) para codificar fluxos de vídeo ao vivo em tempo real. Para codificar um streaming de vídeo grande, o MediaLive o compacta em versões menores (codifica) que podem ser distribuídas aos visualizadores.

Depois de comprimir um fluxo de vídeo ao vivo, você pode usar uma das duas opções principais a seguir para preparar e veicular o conteúdo:

- Converter conteúdo nos formatos necessários e, depois, veiculá-lo: se você precisar de conteúdo em vários formatos, use o [AWS Elemental MediaPackage](#) para empacotar o conteúdo para diferentes tipos de dispositivo. Ao empacotar o conteúdo, você também pode implementar recursos extras e adicionar gerenciamento de direitos digitais (DRM) para evitar o uso não autorizado de seu conteúdo. Para obter instruções passo a passo sobre como usar o CloudFront para fornecer conteúdo formatado pelo MediaPackage, consulte [Veicular vídeo ao vivo formatado com o AWS Elemental MediaPackage](#).
- Armazenar e veicular conteúdo usando origem escalável: se o MediaLive tiver codificado o conteúdo nos formatos exigidos por todos os dispositivos que os visualizadores usam, utilize uma origem altamente escalável como o [AWS Elemental MediaStore](#) para veicular o conteúdo. Para obter instruções passo a passo sobre o uso do CloudFront para fornecer conteúdo armazenado

em um contêiner do MediaStore, consulte [Distribuir vídeo usando o AWS Elemental MediaStore como origem](#).

Depois de configurar a origem usando uma dessas opções, você pode distribuir o streaming de vídeo ao vivo para os visualizadores usando o CloudFront.

 Tip

Você pode saber mais sobre uma solução da AWS que implanta automaticamente serviços para a criação de uma experiência de visualização em tempo real altamente disponível. Para ver as etapas para implantar essa solução automaticamente, consulte [Implantação automatizada de streaming ao vivo](#).

Tópicos

- [Distribuir vídeo usando o AWS Elemental MediaStore como origem](#)
- [Veicular vídeo ao vivo formatado com o AWS Elemental MediaPackage](#)

Distribuir vídeo usando o AWS Elemental MediaStore como origem

Se você tiver vídeo armazenado em um contêiner do [AWS Elemental MediaStore](#), poderá criar uma distribuição do CloudFront para fornecer o conteúdo.

Para começar, conceda ao CloudFront acesso ao contêiner do MediaStore. Crie uma distribuição do CloudFront e configure-a para trabalhar com o MediaStore.

Como veicular conteúdo de um contêiner do AWS Elemental MediaStore

1. Siga o procedimento em [Permitir que o Amazon CloudFront acesse o contêiner do AWS Elemental MediaStore](#) e, depois, retorne a estas etapas para criar a distribuição.
2. Crie uma distribuição com as seguintes configurações:
 - a. Domínio de origem: o endpoint de dados atribuído ao contêiner do MediaStore. Na lista suspensa, escolha o contêiner do MediaStore para o vídeo ao vivo.
 - b. Caminho de origem: a estrutura de pastas no contêiner do MediaStore onde os objetos são armazenados. Para ter mais informações, consulte [the section called “Caminho de origem”](#).

- c. Adicionar cabeçalho personalizado: adicione nomes e valores de cabeçalhos se desejar que o CloudFront inclua cabeçalhos personalizados ao encaminhar solicitações para a origem.
- d. Política de protocolo do visualizador: escolha Redirecionar HTTP para HTTPS. Para ter mais informações, consulte [the section called “Política de protocolo do visualizador”](#).
- e. Política de cache e política de solicitação de origem
 - Em Cache policy (Política de cache), escolha Create policy (Criar política) e, em seguida, crie uma política de cache apropriada para suas necessidades de armazenamento em cache e duração de segmento. Depois de criar a política, atualize a lista de políticas de cache e escolha a que você acabou de criar.
 - Em Origin request policy (Política de solicitação de origem), escolha CORS-CustomOrigin na lista suspensa.

Para as outras configurações, você pode definir valores específicos com base em outros requisitos técnicos ou nas necessidades de sua empresa. Para obter uma lista de todas as opções para distribuições e informações sobre como configurá-las, consulte [the section called “Configurações de distribuição”](#).

3. Para links em sua aplicação (por exemplo, um leitor multimídia), especifique o nome do arquivo de mídia no mesmo formato usado para outros objetos distribuídos usando o CloudFront.

Veicular vídeo ao vivo formatado com o AWS Elemental MediaPackage

Se tiver formatado um streaming ao vivo usando o AWS Elemental MediaPackage, você poderá criar uma distribuição do CloudFront e configurar comportamentos de cache para veicular o streaming ao vivo. O processo a seguir pressupõe que você já tenha [criado um canal](#) e [adicionado endpoints](#) para o vídeo ao vivo usando o MediaPackage.

Para criar uma distribuição do CloudFront para o MediaPackage manualmente, siga estas etapas:

Etapas

- [Etapa 1: Criar e configurar uma distribuição do CloudFront](#)
- [Etapa 2: adicionar Origens aos domínios dos endpoints do MediaPackage](#)
- [Etapa 3: Configurar comportamentos de cache para todos os endpoints](#)
- [Etapa 4: habilitar a autorização CDN do MediaPackage baseada em cabeçalho](#)
- [Etapa 5: usar o CloudFront para atender ao canal de transmissão ao vivo](#)

Etapa 1: Criar e configurar uma distribuição do CloudFront

Execute o procedimento a seguir para configurar uma distribuição do CloudFront para o canal de vídeo ao vivo que você criou com o MediaPackage.

Como criar uma distribuição para seu canal de vídeo ao vivo

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha Create distribution (Criar distribuição).
3. Escolha as configurações para a distribuição, incluindo o seguinte:

Domínio de origem

A origem em que o canal de vídeo ao vivo e os endpoints do MediaPackage estão. Escolha o campo de texto e, na lista suspensa, escolha o domínio de origem do MediaPackage para o vídeo ao vivo. Você pode mapear um domínio para vários endpoints de origem.

Se você criou o domínio de origem usando outra conta da AWS, digite o valor do URL de origem no campo. A origem deve ser um URL de HTTPS.

Por exemplo, para um endpoint HLS como `https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.m3u8`, o domínio de origem é `3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com`.

Para ter mais informações, consulte [the section called “Domínio de origem”](#).

Caminho de origem

O caminho para o endpoint do MediaPackage de onde o conteúdo é fornecido.

O campo Caminho de origem não está preenchido para você. Você deve inserir manualmente o caminho de origem correto.

Para obter mais informações sobre como um caminho de origem funciona, consulte [the section called “Caminho de origem”](#).

⚠ Important

O caminho curinga * é necessário para rotear em algum lugar na distribuição do CloudFront. Para evitar que solicitações que não correspondam a um caminho explícito sejam roteadas para a origem real, crie uma origem “fictícia” para esse caminho curinga.

Example : criar uma origem “fictícia”

No exemplo a seguir, os endpoints abc123 e def456 roteiam para a origem “real”, mas as solicitações de conteúdo de vídeo de qualquer outro endpoint são roteadas para mediapackage.us-west-2.amazonaws.com sem o subdomínio adequado, o que resulta em um erro HTTP 404.

Endpoints do MediaPackage:

```
https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.m3u8
https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/def456/index.m3u8
```

Origem A do CloudFront:

```
Domain: 3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com
Path: None
```

Origem B do CloudFront:

```
Domain: mediapackage.us-west-2.amazonaws.com
Path: None
```

Comportamento do cache do CloudFront:

1. Path: /out/v1/abc123/* forward to Origin A
2. Path: /out/v1/def456/* forward to Origin A
3. Path: * forward to Origin B

Para as outras configurações de distribuição, defina valores específicos com base em outros requisitos técnicos ou nas necessidades de sua empresa. Para obter uma lista de todas as opções para distribuições e informações sobre como configurá-las, consulte [the section called “Configurações de distribuição”](#).

Quando terminar de escolher as outras configurações de distribuição, escolha Create distribution (Criar distribuição).

4. Escolha a distribuição que acabou de criar e, em seguida, Behaviors (Comportamentos).
5. Selecione o comportamento de cache padrão e escolha Edit (Editar). Especifique as configurações do comportamento de cache correto para o canal que você escolheu para a origem. Posteriormente, você incluirá uma ou mais origens adicionais e editará as configurações do comportamento de cache delas.
6. Acesse a [página Distribuições do CloudFront](#).
7. Aguarde até que o valor da coluna Last modified (Última modificação) da distribuição seja alterada de Deploying (Em implantação) para uma data e hora, indicando que o CloudFront criou a distribuição.

Etapa 2: adicionar Origens aos domínios dos endpoints do MediaPackage

Repita as etapas aqui para adicionar cada um dos endpoints do canal do MediaPackage à distribuição, lembrando-se de que é necessário criar uma origem “fictícia”.

Para adicionar outros endpoints como origens

1. No console do CloudFront, escolha a distribuição que você criou para o canal.
2. Escolha Origins (Origens) e Create origin (Criar origem).
3. Em Origin domain (Domínio de origem), na lista suspensa, escolha um endpoint do MediaPackage para o canal.
4. Para as outras configurações, defina valores com base em outros requisitos técnicos ou nas necessidades de sua empresa. Para ter mais informações, consulte [the section called “Configurações de origem”](#).
5. Escolha Create origin (Criar origem).

Etapa 3: Configurar comportamentos de cache para todos os endpoints

Para cada parâmetro, você deve configurar comportamentos de cache para adicionar padrões de caminho que roteiam solicitações corretamente. Os padrões de caminho especificados dependem do formato de vídeo que você está veiculando. O procedimento a seguir inclui as informações de padrão de caminho a serem usadas para os formatos Apple HLS, CMAF, DASH e Microsoft Smooth Streaming.

Geralmente, você configura dois comportamentos de cache para cada endpoint:

- O manifesto pai, que é o índice para seus arquivos.
- Os segmentos, que são os arquivos do conteúdo de vídeo.

Para criar um comportamento de cachê de um endpoint

1. No console do CloudFront, escolha a distribuição que você criou para o canal.
2. Escolha a guia Behaviors (Comportamentos) e Create behavior (Criar comportamento).
3. Em Padrão de caminho, use um GUID `OriginEndpoint` do `MediaPackage` específico como prefixo do caminho.

Padrões de caminho

Para um endpoint HLS como `https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.m3u8`, crie estes dois comportamentos de cache:

- Para manifestos pai e filho, use `/out/v1/abc123/*.m3u8`.
- Para os segmentos de conteúdo, use `/out/v1/abc123/*.ts`.

Para um endpoint CMAF como `https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.m3u8`, crie estes dois comportamentos de cache:

- Para manifestos pai e filho, use `/out/v1/abc123/*.m3u8`.
- Para os segmentos de conteúdo, use `/out/v1/abc123/*.mp4`.

Para um endpoint DASH como `https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.mpd`, crie estes dois comportamentos de cache:

- Para o manifesto pai, use `/out/v1/abc123/*.mpd`.
- Para os segmentos de conteúdo, use `/out/v1/abc123/*.mp4`.

Para um endpoint do Microsoft Smooth Streaming

como `https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.ism`, apenas um manifesto é veiculado e, portanto, você cria apenas um comportamento de cache: `out/v1/abc123/index.ism/*`.

4. Para cada comportamento de cache, especifique os valores das seguintes configurações:

Política de protocolo do visualizador

Escolha Redirect HTTP to HTTPS (Redirecionar HTTP para HTTPS).

Política de cache e política de solicitação de origem

Em Cache policy (Política de cache), escolha Create policy (Criar política). Para sua nova política de cache, especifique as seguintes configurações:

Minimum TTL (TTL mínimo)

Defina como cinco segundos ou menos, para ajudar a evitar o fornecimento de conteúdo obsoleto.

Strings de consulta

Em Query strings (Strings de consulta) (em Cache key settings (Configurações da chave de cache)), escolha Include specified query strings (Incluir strings de consulta especificadas). Em Allow (Permitir), digite os valores a seguir e, em seguida, escolha Add item (Adicionar item):

- Adicione `m` como o parâmetro da string de consulta que você deseja que o CloudFront use como base para o armazenamento em cache. A resposta do MediaPackage sempre inclui a etiqueta `?m=###` para capturar o tempo modificado do endpoint. Se o conteúdo já estiver armazenado em cache com um valor diferente para essa tag, o CloudFront solicitará um novo manifesto em vez de fornecer a versão em cache.
- Se você estiver usando a funcionalidade de visualização em horário deslocado no MediaPackage, especifique `start` e `end` como parâmetros de string de consulta adicionais no comportamento de cache para solicitações de manifesto (`*.m3u8`, `*.mpd` e `index.ism/*`). Dessa forma, o conteúdo é fornecido especificamente para o período solicitado no manifesto pedido. Para receber mais informações sobre a visualização com mudança de horário e a formatação dos parâmetros de solicitação de início e fim

de conteúdo, consulte [Visualização com mudança de horário](#) no Guia do usuário do AWS Elemental MediaPackage.

- Se estiver usando o recurso de filtragem de manifestos no MediaPackage, especifique `aws.manifestfilter` como um parâmetro adicional de string de consulta para a política de consulta que você usa com o comportamento de cache para solicitações de manifesto (`*.m3u8`, `*.mpd` e `index.ism/*`). Isso configura a distribuição para encaminhar a string de consulta `aws.manifestfilter` para a origem do MediaPackage, que é necessário para que o recurso de filtragem de manifestos funcione. Para obter mais informações, consulte [Filtragem de manifestos](#) no Guia do Usuário do AWS Elemental MediaPackage.
- Se você estiver usando HLS de baixa latência (LL-HLS), especifique `_HLS_msn` e `_HLS_part` como parâmetros adicionais de string de consulta para a política de cache que você usa com o comportamento de cache para solicitações de manifesto (`*.m3u8`). Isso configura a distribuição para encaminhar as strings de consulta `_HLS_msn` e `_HLS_part` para a origem do MediaPackage, o que é necessário para que o atributo de solicitação de bloqueio de playlist LL-HLS funcione.

5. Escolha Criar.
6. Depois de criar a política de cache, volte para o fluxo de trabalho de criação do comportamento do cache. Atualize a lista de políticas de cache e escolha a política que você acabou de criar.
7. Escolha Create behavior (Criar comportamento).
8. Se o endpoint não for um endpoint do Microsoft Smooth Streaming, repita essas etapas para criar um segundo comportamento de cache.

Etapa 4: habilitar a autorização CDN do MediaPackage baseada em cabeçalho

Recomendamos habilitar a autorização CDN do MediaPackage baseada em cabeçalho entre os endpoints do MediaPackage e a distribuição do CloudFront. Para ter mais informações, consulte [Habilitar a autorização de CDN em MediaPackage](#) no Guia do usuário do AWS Elemental MediaPackage.

Etapa 5: usar o CloudFront para atender ao canal de transmissão ao vivo

Depois de criar a distribuição, adicionar as origens, criar os comportamentos de cache e habilitar a autorização CDN baseada em cabeçalho, você pode fornecer o canal de streaming ao vivo usando o CloudFront. O CloudFront roteia solicitações dos visualizadores para os endpoints MediaPackage corretos com base nas configurações que você definiu para os comportamentos de cache.

Para links em sua aplicação (por exemplo, um media player), especifique o URL do arquivo de mídia no formato padrão para URLs do CloudFront. Para ter mais informações, consulte [the section called “Personalizar URLs de arquivos”](#).

Personalizar na borda com funções

Com o Amazon CloudFront, você pode escrever seu próprio código para personalizar como suas distribuições do CloudFront processam solicitações e respostas HTTP. O código é executado perto dos visualizadores (usuários) para minimizar a latência e você não precisa gerenciar servidores ou outra infraestrutura. Você pode escrever código para manipular as solicitações e respostas que fluem pelo CloudFront, executar autenticação e autorização básicas, gerar respostas HTTP na borda e muito mais.

O código que você escreve e anexa à sua distribuição do CloudFront é chamado de função da borda. O CloudFront oferece duas maneiras de escrever e gerenciar funções da borda:

CloudFront Functions

É possível escrever funções leves em JavaScript para personalizações de CDN de alta escala e sensíveis à latência. O ambiente de tempo de execução do CloudFront Functions oferece tempos de startup de submilissegundos, é dimensionado imediatamente para lidar com milhões de solicitações por segundo e é altamente seguro. O CloudFront Functions é um recurso nativo do CloudFront, o que significa que você pode criar, testar e implantar seu código inteiramente no CloudFront.

Lambda@Edge

O Lambda@Edge é uma extensão do [AWS Lambda](#) que oferece computação avançada e flexível para funções complexas e lógica completa de aplicações mais perto dos visualizadores, além de ser altamente seguro. As funções do Lambda@Edge são executadas em um ambiente de tempo de execução Node.js ou Python. Você as publica em uma única Região da AWS e ao associar a função a uma distribuição do CloudFront, o Lambda@Edge replica automaticamente o código no mundo todo.

Se você executa AWS WAF no CloudFront, é possível usar cabeçalhos AWS WAF inseridos para o CloudFront Functions e para o Lambda@Edge. Isso funciona para solicitações e respostas do visualizador e da origem.

Tópicos

- [Diferenças entre o CloudFront Functions e o Lambda@Edge](#)
- [Personalizar na borda com o CloudFront Functions](#)
- [Personalização na borda com o Lambda@Edge](#)

- [Restrições das funções de borda](#)

Diferenças entre o CloudFront Functions e o Lambda@Edge

O CloudFront Functions e o Lambda@Edge fornecem uma maneira de executar código em resposta aos eventos do CloudFront.

O CloudFront Functions é ideal para funções leves e curtas para os seguintes casos de uso:

- Normalização da chave de cache: transforme atributos de solicitação HTTP (cabeçalhos, strings de consulta, cookies e até mesmo o caminho do URL) para criar uma [chave de cache](#) ideal, que pode melhorar a taxa de acertos do cache.
- Manipulação de cabeçalhos: insira, modifique ou exclua cabeçalhos HTTP na solicitação ou na resposta. Por exemplo, você pode adicionar um cabeçalho `True-Client-IP` a cada solicitação.
- Redirecionamento ou regravações de URL: redirecione os visualizadores para outras páginas com base nas informações da solicitação ou reescreva todas as solicitações de um caminho para outro.
- Solicitar autorização: valide tokens de autorização com hash, como tokens Web JSON (JWT), por meio da inspeção dos cabeçalhos de autorização ou outros metadados de solicitação.

Para começar a usar o CloudFront Functions, consulte [Personalizar na borda com o CloudFront Functions](#).

O Lambda@Edge é ideal para os seguintes casos de uso:

- Funções que levam alguns milissegundos ou mais para serem concluídas.
- Funções que exigem CPU ou memória ajustável.
- Funções que dependem de bibliotecas de terceiros (incluindo o SDK da AWS para integração com outros Serviços da AWS).
- Funções que exigem acesso à rede para usar serviços externos para processamento.
- Funções que exigem acesso ao sistema de arquivos ou acesso ao corpo de solicitações HTTP.

Para começar a usar o Lambda@Edge, consulte [Personalização na borda com o Lambda@Edge](#).

Para ajudar você a escolher a opção para o caso de uso, use a tabela a seguir para entender as diferenças entre o CloudFront Functions e o Lambda@Edge.

	CloudFront Functions	Lambda@Edge
Linguagens de programação	JavaScript (compatível com ECMAScript 5.1)	Node.js e Python
Origens de eventos	<ul style="list-style-type: none"> • Solicitação do visualizador • Resposta do visualizador 	<ul style="list-style-type: none"> • Solicitação do visualizador • Resposta do visualizador • Solicitação da origem • Resposta da origem
Suporte a KeyValueStore do Amazon CloudFront	<p>Sim</p> <p>O KeyValueStore do CloudFront é compatível somente com o runtime 2.0 do JavaScript</p>	Não
Dimensionar	10.000.000 de solicitações por segundo ou mais	Até 10.000 solicitações por segundo por região
Duração da função	Submilissegundo	<p>Até 5 segundos (solicitação do visualizador e resposta do visualizador)</p> <p>Até 30 segundos (solicitação da origem e resposta da origem)</p>
Tamanho máximo de memória da função	2 MB	<p>128 MB (solicitação e resposta do visualizador)</p> <p>10.240 MB (10 GB) (solicitação e resposta da origem)</p>

	CloudFront Functions	Lambda@Edge
		Para ter mais informações, consulte Cotas do Lambda@Edge .
Tamanho máximo do código de função e bibliotecas incluídas	10 KB	1 MB (solicitação do visualizador e resposta do visualizador) 50 MB (solicitação da origem e resposta da origem)
Acesso à rede	Não	Sim
Acesso ao sistema de arquivos	Não	Sim
Acesso ao órgão de solicitação	Não	Sim
Acesso a dados de geolocalização e dispositivos	Sim	Não (solicitação do visualizador e resposta do visualizador) Sim (solicitação da origem e resposta da origem)
Pode criar e testar inteiramente no CloudFront	Sim	Não
Registro em log de funções e métricas	Sim	Sim
Definição de preços	Nível gratuito disponível; cobrado por solicitação	Sem nível gratuito; cobrado por solicitação e duração da função

Personalizar na borda com o CloudFront Functions

Com o CloudFront Functions, você pode escrever funções leves em JavaScript para personalizações de CDN de alta escala e sensíveis à latência. Suas funções podem manipular as solicitações e respostas que fluem pelo CloudFront, executar autenticação e autorização básicas, gerar respostas HTTP na borda e muito mais. O ambiente de tempo de execução do CloudFront Functions oferece tempos de startup de submilissegundos, é dimensionado imediatamente para lidar com milhões de solicitações por segundo e é altamente seguro. O CloudFront Functions é um recurso nativo do CloudFront, o que significa que você pode criar, testar e implantar seu código inteiramente no CloudFront.

Quando você associa uma função do CloudFront a uma distribuição do Lambda, o CloudFront intercepta solicitações e respostas nos locais da borda do CloudFront e os passa à sua função. Você pode invocar CloudFront Functions quando ocorrerem os seguintes eventos:

- Quando o CloudFront receber uma solicitação de um visualizador (solicitação do visualizador)
- Antes do CloudFront retornar a resposta para o visualizador (resposta do visualizador)

Para ter mais informações sobre o CloudFront Functions, consulte os seguintes tópicos:

Tópicos

- [Tutorial: Criar uma função simples com o CloudFront Functions](#)
- [Tutorial: Criar uma função do CloudFront que inclua valores de chave](#)
- [Escrever código de função](#)
- [Criar funções](#)
- [Testar funções](#)
- [Atualizar funções](#)
- [Publicar as funções](#)
- [Associar funções a distribuições](#)
- [KeyValueStore do Amazon CloudFront](#)

Tutorial: Criar uma função simples com o CloudFront Functions

Este tutorial mostra como começar a usar o CloudFront Functions. É possível criar uma função simples que redirecione o visualizador para um URL diferente e também exiba um cabeçalho de resposta personalizado.

Sumário

- [Pré-requisitos](#)
- [Criar a função](#)
- [Verificar a função](#)

Pré-requisitos

Para usar o CloudFront Functions, você precisa de uma distribuição do CloudFront. Se você não tiver uma, consulte [Conceitos básicos de uma distribuição básica do CloudFront](#).

Criar a função

É possível usar o console do CloudFront para criar uma função simples que redirecione o visualizador para um URL diferente e também exiba um cabeçalho de resposta personalizado.


Como criar uma função do CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Funções e selecione Criar função.
3. Na página Create function (Criar função), em Name (Nome), insira um nome de função, como *MyFunctionName*.
4. (Opcional) Em Description (Descrição), insira uma descrição para a função, como **Simple test function**.
5. Em Runtime, mantenha a versão padrão selecionada do JavaScript.
6. Escolha a opção Criar função.
7. Copie o código da função a seguir. Esse código de função redireciona o visualizador para uma URL diferente e também retorna um cabeçalho de resposta personalizado.

```
function handler(event) {
```

```
// NOTE: This example function is for a viewer request event trigger.
// Choose viewer request for event trigger when you associate this function
with a distribution.
var response = {
  statusCode: 302,
  statusDescription: 'Found',
  headers: {
    'cloudfront-functions': { value: 'generated-by-CloudFront-Functions' },
    'location': { value: 'https://aws.amazon.com/cloudfront/' }
  }
};
return response;
}
```

8. Em Function code (Código da função), cole o código no editor para substituir o código padrão.
9. Escolha Salvar alterações.
10. (Opcional) É possível testar a função antes de publicá-la. Este tutorial não descreve como testar uma função. Para ter mais informações, consulte [Testar funções](#).
11. Escolha a guia Publish (Publicar) e, em seguida, escolha Publish function (Publicar função). Você deve publicar a função antes de associá-la à sua distribuição do CloudFront.
12. Em seguida, é possível associar a função a uma distribuição ou comportamento de cache. Na página *MyFunctionName*, selecione a guia Publish (Publicar).

 Warning

Nas etapas a seguir, selecione uma distribuição ou um comportamento de cache que seja usado para testes. Não associe essa função de demonstração a um comportamento de distribuição ou cache usado na produção.

13. Escolha Add association.
14. Na caixa de diálogo Associate (Associar), escolha uma distribuição e/ou um comportamento de cache. Em Event type (Tipo de evento), mantenha o valor padrão.
15. Escolha Add association.

A tabela Distribuições associadas mostra a distribuição associada.

16. Aguarde alguns minutos para que a distribuição associada termine a implantação. Para conferir o status da distribuição, selecione a distribuição na tabela Associated distributions (Distribuições associadas) e escolha View distribution (Visualizar distribuição).

Quando o status da distribuição for Deployed (Implantado), você estará pronto para verificar se a função funciona.

Verificar a função

Depois de implantar a função, você pode verificar se ela está funcionando para a distribuição.

Como verificar a função

1. No navegador da Web, acesse o nome de domínio da distribuição (por exemplo, `https://d111111abcdef8.cloudfront.net`).

A função retorna um redirecionamento para o navegador, de modo que o navegador vai automaticamente para `https://aws.amazon.com/cloudfront/`.

2. Em uma janela de linha de comando, é possível usar uma ferramenta, como curl, para enviar uma solicitação ao nome de domínio da distribuição.

```
curl -v https://d111111abcdef8.cloudfront.net/
```

Na resposta, você verá a resposta de redirecionamento (302 Found) e os cabeçalhos de resposta personalizados que a função adicionou. O resultado deve ser semelhante ao seguinte.

Example

```
curl -v https://d111111abcdef8.cloudfront.net/
> GET / HTTP/1.1
> Host: d111111abcdef8.cloudfront.net
> User-Agent: curl/7.64.1
> Accept: */*
>
< HTTP/1.1 302 Found
< Server: CloudFront
< Date: Tue, 16 Mar 2021 18:50:48 GMT
< Content-Length: 0
< Connection: keep-alive
< Location: https://aws.amazon.com/cloudfront/
< Cloudfront-Functions: generated-by-CloudFront-Functions
< X-Cache: FunctionGeneratedResponse from cloudfront
< Via: 1.1 3035b31bddaf14eded329f8d22cf188c.cloudfront.net (CloudFront)
< X-Amz-Cf-Pop: PHX50-C2
```



```
< X-Amz-Cf-Id: ULZdIz6j43uGB1Xyob_JctF9x7CCbwpNniiM1mNbmwzH1YWP9FsEHg==
```

Tutorial: Criar uma função do CloudFront que inclua valores de chave

Este tutorial mostra como incluir chave-valor com a função do CloudFront. Os valores de chave fazem parte de um par de chave-valor. Você deve incluir o nome (do par de chave-valor) no código de função. Quando a função for executada, o CloudFront substituirá o nome pelo valor.

Pares de chave-valor são variáveis contidas em um armazenamento de valores de chave. Ao usar uma chave na função (em vez de valores codificados), a função é mais flexível. É possível alterar o valor da chave sem precisar implantar alterações no código. Os pares de chave-valor também podem reduzir o tamanho da função. Para ter mais informações, consulte [???](#).

Sumário

- [Pré-requisitos](#)
- [Criar o armazenamento de valores de chave](#)
- [Adicionar pares de chave-valor ao armazenamento de valores de chave](#)
- [Associar o armazenamento de valores de chave à função](#)
- [Testar e publicar o código de função](#)

Pré-requisitos

Se você não conhece as funções do CloudFront Functions e o armazenamento de valores de chave, recomendamos seguir o tutorial em [the section called “Tutorial: Criar uma função simples do CloudFront”](#).

Depois de concluir esse tutorial, você pode seguir este tutorial para estender a função que você criou. Para este tutorial, recomendamos criar primeiro o armazenamento de valores de chave.

Criar o armazenamento de valores de chave

Primeiro, crie o armazenamento de valores de chave para usar em sua função.

Como criar o armazenamentos de valores de chave

1. Planeje os pares de chave-valor que você deseja incluir na função. Anote os nomes das chaves. Os pares de chave-valor a serem usados em uma função devem estar em um único armazenamento de valores de chave.

2. Decida-se sobre a ordem do trabalho. Há duas maneiras de proceder:
 - Crie um armazenamento de valores de chave e adicione pares de chave-valor ao armazenamento. Depois, crie (ou modifique) a função e incorpore os nomes das chaves.
 - Ou crie (ou modifique) a função e incorpore os nomes das chaves que deseja usar. Depois, crie um armazenamento de valores de chave e adicione pares de chave-valor.
3. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
4. No painel de navegação, selecione Funções e, depois, escolha a guia KeyValueStores.
5. Selecione Criar KeyValueStore e preencha os seguintes campos:
 - Informe um nome e uma descrição (opcional) para o armazenamento.
 - Deixe o URI do S3 em branco. Neste tutorial, você inserirá os pares de chave-valor manualmente.
6. Escolha Criar. A página de detalhes do novo armazenamento de chave-valor é exibida. Essa página inclui uma seção de Pares de chave-valor que está em branco no momento.

Adicionar pares de chave-valor ao armazenamento de valores de chave

Depois, adicione manualmente uma lista de pares de chave-valor ao armazenamento de valores de chave que você criou anteriormente.

Como adicionar pares de chave-valor ao armazenamento de valores de chave

1. Na seção Pares de chave-valor, selecione Adicionar pares de chave-valor.
2. Selecione Adicionar par e, depois, insira uma chave e um valor. Escolha a marca de seleção para confirmar suas alterações e repita essa etapa para adicionar mais.
3. Ao terminar, selecione Salvar alterações para salvar os pares de chave-valor no armazenamento de valores de chave. Na caixa de diálogo de confirmação, escolha Concluído.

Agora você tem um armazenamento de valores de chave que contém um grupo de pares de chave-valor.

Associar o armazenamento de valores de chave à função

Agora você criou o armazenamento de chave-valor. E criou ou modificou uma função que inclui os nomes das chaves do armazenamento de chave-valor. Agora é possível associar o armazenamento de chave-valor e a função. Você vai criar essa associação de dentro da função.

Como associar o armazenamento de valores de chave à função

1. Selecione Funções no painel de navegação. A guia Funções é exibida na parte superior, por padrão.
2. Selecione o nome da função e na seção KeyValueStore associado, selecione Associar KeyValueStore existente.
3. Selecione o armazenamento de valores de chave e escolha Associar KeyValueStore.

Note

É possível associar somente um armazenamento de valores de chave a cada função.

Testar e publicar o código de função

Depois de associar o armazenamento de valores de chave à sua função, você pode testar e publicar o código da função. Sempre é necessário testar o código da função toda vez que o modificar, inclusive ao fazer o seguinte:

- Associar o armazenamento de chave-valor à função.
- Modifique a função e o armazenamento de valores de chave para incluir um novo par de chave-valor.
- Alterar o valor de um par de chave-valor.

Como testar e publicar o código da função

1. Para obter informações sobre como testar uma função, consulte [the section called “Testar funções”](#). Verificar se escolheu testar a função no estágio DEVELOPMENT.
2. Publique a função quando estiver tudo pronto para usá-la (com os pares de chave-valor novos ou revisados) em um ambiente LIVE.

Na publicação, o CloudFront copia a versão da função do estágio DEVELOPMENT para o estágio ativo. A função tem o novo código e está associada ao armazenamento de chave-valor. (Não há necessidade de realizar a associação novamente, no estágio ativo.)

Para obter informações sobre como publicar a função, consulte [the section called “Publicar as funções”](#).

Escrever código de função

É possível usar o CloudFront Functions para escrever funções leves em JavaScript para personalizações de CDN de alta escala e sensíveis à latência. Seu código de função pode manipular as solicitações e respostas que fluem pelo CloudFront, executar autenticação e autorização básicas, gerar respostas HTTP na borda e muito mais.

Para ajudar a escrever código de função para o CloudFront Functions, consulte os tópicos a seguir.

Tópicos

- [Determinar o propósito da função](#)
- [Estrutura de eventos do CloudFront Functions](#)
- [Recursos de tempo de execução JavaScript para funções do CloudFront](#)
- [Métodos auxiliares para armazenamentos de chave-valor](#)
- [Código de exemplo para CloudFront Functions](#)

Determinar o propósito da função

Antes de escrever seu código de função, determine o propósito da sua função. A maioria das funções no CloudFront Functions tem uma das seguintes finalidades.

Tópicos

- [Modificar a solicitação HTTP em um tipo de evento de solicitação do visualizador](#)
- [Gerar uma resposta HTTP em um tipo de evento de solicitação do visualizador](#)
- [Modificar a resposta HTTP em um tipo de evento de resposta do visualizador](#)
- [Informações relacionadas](#)

Independentemente do propósito da sua função, o `handler` é o ponto de entrada para qualquer função. É preciso um único argumento chamado `event`, que é passado para a função pelo CloudFront. O `event` é um objeto JSON que contém uma representação da solicitação HTTP (e a resposta, se sua função modificar a resposta HTTP).

Modificar a solicitação HTTP em um tipo de evento de solicitação do visualizador

Sua função pode modificar a solicitação HTTP que o CloudFront recebe do visualizador (cliente) e retornar a solicitação modificada ao CloudFront para processamento contínuo. Por exemplo, seu código de função pode normalizar a [chave de cache](#) ou modificar cabeçalhos de solicitação.

Quando você criar uma função que modifica a solicitação HTTP, certifique-se de selecionar o tipo de evento `viewer request` (solicitação do visualizador). Isso significa que a função é executada sempre que o CloudFront recebe uma solicitação de um visualizador, antes de verificar se o objeto solicitado está no cache do CloudFront.

Example Exemplo

O pseudocódigo a seguir mostra a estrutura de uma função que modifica a solicitação HTTP.

```
function handler(event) {
    var request = event.request;

    // Modify the request object here.

    return request;
}
```

A função retorna o objeto `request` modificado para o CloudFront. O CloudFront continua processando a solicitação retornada verificando o cache do CloudFront quanto a uma ocorrência de cache e enviando a solicitação para a origem, se necessário.

Gerar uma resposta HTTP em um tipo de evento de solicitação do visualizador

Sua função pode gerar uma resposta HTTP na borda e retorná-la diretamente ao visualizador (cliente) sem verificar se há uma resposta em cache ou qualquer processamento adicional pelo CloudFront. Por exemplo, seu código de função pode redirecionar a solicitação para uma nova URL ou verificar se há autorização e retornar uma resposta 401 ou 403 a solicitações não autorizadas.

Quando você criar uma função que gera uma resposta HTTP, certifique-se de selecionar o tipo de evento `viewer request` (solicitação do visualizador). Isso significa que a função é executada sempre

que o CloudFront recebe uma solicitação de um visualizador, antes que o CloudFront faça qualquer processamento adicional da solicitação.

Example Exemplo

O pseudocódigo a seguir mostra a estrutura de uma função que gera uma resposta HTTP.

```
function handler(event) {
    var request = event.request;

    var response = ...; // Create the response object here,
                        // using the request properties if needed.

    return response;
}
```

A função retorna um objeto `response` ao CloudFront, que o CloudFront retorna imediatamente ao visualizador sem verificar o cache do CloudFront ou enviar uma solicitação para a origem.

Modificar a resposta HTTP em um tipo de evento de resposta do visualizador

Sua função pode modificar a resposta HTTP antes que o CloudFront a envie para o visualizador (cliente), independentemente de a resposta vir do cache do CloudFront ou da origem. Por exemplo, o código da função pode adicionar ou modificar cabeçalhos de resposta, códigos de status e o conteúdo do corpo.

Ao criar uma função que modifica a resposta HTTP, certifique-se de selecionar o tipo de evento de `viewer response` (resposta do visualizador). Isso significa que a função é executada antes que o CloudFront retorne uma resposta ao visualizador, independentemente de a resposta vir do cache do CloudFront ou da origem.

Example Exemplo

O pseudocódigo a seguir mostra a estrutura de uma função que modifica a resposta HTTP.

```
function handler(event) {
    var request = event.request;
    var response = event.response;

    // Modify the response object here,
    // using the request properties if needed.

    return response;
}
```

```
}
```

A função retorna o objeto `response` modificado ao CloudFront, que o CloudFront retorna imediatamente ao visualizador.

Informações relacionadas

Para ter mais informações sobre como trabalhar com o CloudFront Functions, consulte os seguintes tópicos:

- [Estrutura de eventos](#)
- [Recursos de tempo de execução JavaScript](#)
- [Código de exemplo](#)
- [Restrições das funções de borda](#)

Estrutura de eventos do CloudFront Functions

O CloudFront Functions passa um objeto `event` para o seu código de função como entrada quando executa a função. Quando você [testa uma função](#), você cria o objeto `event` e o passa para sua função. Quando você cria um objeto `event` para testar uma função, você pode omitir os campos `distributionDomainName`, `distributionId` e `requestId` no objeto `context`. Certifique-se de que os nomes de cabeçalhos estejam em letras minúsculas, o que sempre é o caso no objeto `event` que o CloudFront Functions passa para sua função na produção.

Segue uma visão geral da estrutura desse objeto de evento.

```
{
  "version": "1.0",
  "context": {
    <context object>
  },
  "viewer": {
    <viewer object>
  },
  "request": {
    <request object>
  },
  "response": {
    <response object>
  }
}
```

```
}
```

Para obter mais informações, consulte os tópicos a seguir.

Tópicos

- [Campo de versão](#)
- [Objeto de contexto](#)
- [Objeto do visualizador](#)
- [Objeto de solicitação](#)
- [Objeto da resposta](#)
- [Código de status e corpo](#)
- [Estrutura de uma string de consulta, um cabeçalho ou um cookie](#)
- [Exemplo de objeto de resposta](#)
- [Exemplo de objeto de evento](#)

Campo de versão

O campo `version` contém uma cadeia de caracteres que especifica a versão do objeto de evento do CloudFront Functions. A versão atual é `1.0`.

Objeto de contexto

O objeto `context` contém informações contextuais sobre o evento. Isso inclui os seguintes campos:

distributionDomainName

O nome de domínio do CloudFront (por exemplo, `d111111abcdef8.cloudfront.net`) da distribuição associada ao evento.

distributionId

O ID da distribuição (por exemplo, `EDFDVBD6EXAMPLE`) que está associado ao evento.

eventType

O tipo de evento, `viewer-request` ou `viewer-response`.

requestId

Uma cadeia de caracteres que identifica exclusivamente uma solicitação do CloudFront (e sua resposta associada).

Objeto do visualizador

O objeto `viewer` contém um campo `ip` cujo valor é o endereço IP do visualizador (cliente) que enviou a solicitação. Se o visualizador usar um proxy HTTP ou um balanceador de carga para enviar a solicitação, o valor será o endereço IP do proxy ou do balanceador de carga.

Objeto de solicitação

O objeto `request` contém uma representação de uma solicitação HTTP Viewer-to-CloudFront. No objeto `event` passado para a função, o objeto `request` representa a solicitação real que o CloudFront recebeu do visualizador.

Se o código de função retornar um objeto `request` ao CloudFront, ele deverá usar essa mesma estrutura.

O objeto `request` contém os campos a seguir.

method

O método HTTP da solicitação. Se o código de função exibir uma `request`, ele não poderá modificar esse campo. Este é o único campo somente leitura no objeto `request`.

uri

O caminho relativo do objeto solicitado.

Note

Se a sua função modificar o valor `uri`, o seguinte será aplicável:

- O novo valor `uri` deve começar com uma barra (/).
- Se uma função alterar o valor `uri`, isso alterará o objeto solicitado pelo visualizador.
- Se uma função alterar o valor do `uri`, isso não mudará o comportamento do cache da solicitação ou da origem para a qual a solicitação de origem é enviada.

querystring

Um objeto que representa a cadeia de consulta na solicitação. Se a solicitação não inclui uma string de consulta, o objeto `request` ainda incluirá um objeto `querystring` vazio.

O objeto `queryString` contém um campo para cada parâmetro de cadeia de consulta na solicitação.

headers

Um objeto que representa os cabeçalhos HTTP na solicitação. Se a solicitação contiver quaisquer cabeçalhos `Cookie`, esses cabeçalhos não farão parte do objeto `headers`. Os cookies são representados separadamente no objeto `cookies`.

O objeto `headers` contém um campo para cada cabeçalho na solicitação. Os nomes de cabeçalho são convertidos em letras minúsculas no objeto do evento, e os nomes de cabeçalho devem estar em letras minúsculas quando forem adicionados pelo código da função. Quando o CloudFront Functions converte o objeto de evento novamente em uma solicitação HTTP, a primeira letra de toda palavra em nomes de cabeçalho é maiúscula. As palavras são separadas por hífen (-). Por exemplo, se o código da função adicionar um cabeçalho chamado `example-header-name`, o CloudFront o converterá em `Example-Header-Name` na solicitação HTTP.

cookies

Um objeto que representa os cookies na solicitação (cabeçalhos `Cookie`).

O objeto `cookies` contém um campo para cada cookie na solicitação.

Para obter mais informações sobre a estrutura de cadeias de consulta, cabeçalhos e cookies, consulte [Estrutura de uma string de consulta, um cabeçalho ou um cookie](#).

Para obter um objeto `event` de exemplo, consulte [Exemplo de objeto de evento](#).

Objeto da resposta

O objeto `response` contém uma representação de uma resposta HTTP do CloudFront-to-viewer. No objeto `event` passado para a função, o objeto `response` representa a resposta real do CloudFront a uma solicitação de visualizador.

Se seu código de função retornar um objeto `response`, ele deverá usar essa mesma estrutura.

O objeto `response` contém os campos a seguir.

statusCode

O código de status HTTP da resposta. Esse valor é um inteiro, não uma cadeia de caracteres.

Sua função pode gerar ou modificar o `statusCode`.

statusDescription

A descrição do status HTTP da resposta. Se o seu código de função gerar uma resposta, esse campo será opcional.

headers

Um objeto que representa os cabeçalhos HTTP na resposta. Se a resposta contiver quaisquer cabeçalhos `Set-Cookie`, esses cabeçalhos não farão parte do objeto `headers`. Os cookies são representados separadamente no objeto `cookies`.

O objeto `headers` contém um campo para cada cabeçalho na resposta. Os nomes de cabeçalho são convertidos em letras minúsculas no objeto do evento, e os nomes de cabeçalho devem estar em letras minúsculas quando forem adicionados pelo código da função. Quando o CloudFront Functions converte o objeto de evento novamente em uma resposta HTTP, a primeira letra de toda palavra em nomes de cabeçalho é maiúscula. As palavras são separadas por hífen (-). Por exemplo, se o código da função adicionar um cabeçalho chamado `example-header-name`, o CloudFront o converterá em `Example-Header-Name` na resposta HTTP.

cookies

Um objeto que representa os cookies na resposta (cabeçalhos `Set-Cookie`).

O objeto `cookies` contém um campo para cada cookie na resposta.

body

Adicionar o campo `body` é opcional e ele não estará presente no objeto `response`, a menos que você o especifique na função. A função não tem acesso ao corpo original retornado pelo cache ou pela origem do CloudFront. Se você não especificar o campo `body` na função de resposta do visualizador, o corpo original retornado pelo cache do CloudFront ou pela origem será retornado ao visualizador.

Se quiser que o CloudFront retorne um corpo personalizado ao visualizador, especifique o conteúdo do corpo no campo `data` e a codificação do corpo no campo `encoding`. É possível especificar a codificação como texto simples (`"encoding": "text"`) ou como conteúdo codificado em Base64 (`"encoding": "base64"`).

Como atalho, também é possível especificar o conteúdo do corpo diretamente no campo `body` (`"body": "<specify the body content here>"`). Ao fazer isso, omite os campos `data` e `encoding`. Nesse caso, o CloudFront tratará o corpo como texto simples.

encoding

A codificação do conteúdo do body (campo data). As únicas codificações válidas são text e base64.

Se você especificar encoding como base64, mas o corpo não for um base64 válido, o CloudFront retornará um erro.

data

O conteúdo do body.

Para obter mais informações sobre códigos de status modificados e conteúdo do corpo, consulte [Código de status e corpo](#).

Para obter mais informações sobre a estrutura de cabeçalhos e cookies, consulte [Estrutura de uma string de consulta, um cabeçalho ou um cookie](#).

Para obter um objeto response de exemplo, consulte [Exemplo de objeto de resposta](#).

Código de status e corpo

Com o CloudFront Functions, é possível atualizar o código de status da resposta do visualizador, substituir todo o corpo da resposta por um novo ou remover o corpo da resposta. Alguns cenários comuns para atualizar a resposta do visualizador após avaliar os aspectos da resposta do cache ou da origem do CloudFront incluem o seguinte:

- Alterar o status para definir um código de status HTTP 200 e a criar conteúdo estático do corpo para retornar ao visualizador.
- Alterar o status para definir um código de status HTTP 301 ou 302 para redirecionar o usuário para outro site.
- Decidir se deseja enviar ou descartar o corpo da resposta do visualizador.

Note

Se a origem retornar um erro HTTP igual ou superior a 400, a função do CloudFront não será executada. Para obter mais informações, consulte [Restrições de todas as funções de borda](#).

Ao trabalhar com a resposta HTTP, o CloudFront Functions não tem acesso ao corpo da resposta. É possível substituir o conteúdo estático do corpo definindo-o como o valor desejado ou remover o corpo definindo o valor como vazio. Se você não atualizar o campo do corpo na função, o corpo original retornado pelo cache do CloudFront será retornado ao visualizador.

Tip

Ao usar o CloudFront Functions para substituir um corpo, certifique-se de alinhar os cabeçalhos correspondentes, como `content-encoding`, `content-type` ou `content-length`, ao novo conteúdo do corpo.

Por exemplo, se a origem ou o cache do CloudFront retornar `content-encoding: gzip`, mas a função de resposta do visualizador definir um corpo que seja texto simples, a função também precisará alterar os cabeçalhos `content-encoding` e `content-type` adequadamente.

Se a sua função do CloudFront estiver configurada para retornar um erro HTTP de 400 ou superior, seu visualizador não verá uma [página de erro personalizada](#) que você especificou para o mesmo código de status.

Estrutura de uma string de consulta, um cabeçalho ou um cookie

Strings de consulta, cabeçalhos e cookies compartilham a mesma estrutura. As strings de consulta podem aparecer em solicitações. Os cabeçalhos aparecem em solicitações e respostas. Os cookies aparecem em solicitações e respostas.

Cada cadeia de consulta, cabeçalho ou cookie é um campo exclusivo dentro do objeto pai `querystring`, `headers` ou `cookies`. O nome do campo é o nome da cadeia de consulta, cabeçalho ou cookies. Cada campo contém uma propriedade `value` com o valor da cadeia de consulta, cabeçalho ou cookie.

Sumário

- [Valores de string de consulta ou objetos de string de consulta](#)
- [Considerações especiais sobre cabeçalhos](#)
- [Duplicar cadeias de consulta, cabeçalhos e cookies \(matriz multiValue\)](#)
- [Atributos de cookies](#)

Valores de string de consulta ou objetos de string de consulta

Uma função pode gerar um valor além de um objeto de string de consulta. O valor de string de consulta pode ser usado para organizar os parâmetros da string de consulta em qualquer ordem personalizada.

Example Exemplo

Para modificar uma string de consulta no código de função, use o código da seguinte maneira:

```
var request = event.request;
request.querystring =
  'ID=42&Exp=1619740800&TTL=1440&NoValue=&querymv=val1&querymv=val2,val3';
```

Considerações especiais sobre cabeçalhos

Somente para cabeçalhos, os nomes de cabeçalho são convertidos em letras minúsculas no objeto do evento, e os nomes de cabeçalho devem estar em letras minúsculas quando forem adicionados pelo código da função. Quando o CloudFront Functions converte o objeto de evento novamente em uma solicitação ou resposta HTTP, a primeira letra de toda palavra em nomes de cabeçalho é maiúscula. As palavras são separadas por hífen (-). Por exemplo, se o código da função adicionar um cabeçalho chamado `example-header-name`, o CloudFront o converterá em `Example-Header-Name` na solicitação ou resposta HTTP.

Example Exemplo

Pense no cabeçalho `Host` a seguir em uma solicitação HTTP.

```
Host: video.example.com
```

Esse cabeçalho é representado da seguinte forma no objeto `request`:

```
"headers": {
  "host": {
    "value": "video.example.com"
  }
}
```

Para acessar o cabeçalho `Host` em seu código de função, use o código da seguinte maneira:

```
var request = event.request;
```

```
var host = request.headers.host.value;
```

Para adicionar ou modificar um cabeçalho em seu código de função, use o código como a seguir (esse código adiciona um cabeçalho chamado `X-Custom-Header` com o valor `example value`):

```
var request = event.request;
request.headers['x-custom-header'] = {value: 'example value'};
```

Duplicar cadeias de consulta, cabeçalhos e cookies (matriz `multiValue`)

Uma solicitação ou resposta HTTP pode conter mais de uma cadeia de consulta, cabeçalho ou cookie com o mesmo nome. Nesse caso, as cadeias de consulta duplicadas, cabeçalhos ou cookies são recolhidos em um campo no objeto `request` ou `response`, mas esse campo contém uma propriedade extra chamada `multiValue`. A propriedade `multiValue` contém uma matriz com os valores de cada uma das cadeias de consulta duplicadas, cabeçalhos ou cookies.

Example Exemplo

Pense em uma solicitação HTTP com os cabeçalhos `Accept` a seguir.

```
Accept: application/json
Accept: application/xml
Accept: text/html
```

Esses cabeçalhos são representados da forma a seguir no objeto `request`.

```
"headers": {
  "accept": {
    "value": "application/json",
    "multiValue": [
      {
        "value": "application/json"
      },
      {
        "value": "application/xml"
      },
      {
        "value": "text/html"
      }
    ]
  }
}
```

```
}
```

Note

O primeiro valor de cabeçalho (nesse caso, `application/json`) é repetido nas duas propriedades `value` e `multiValue`. Isso permite que você acesse todos os valores por loop por meio da matriz `multiValue`.

Se o código de função modificar uma string de consulta, um cabeçalho ou um cookie com uma matriz `multiValue`, o CloudFront Functions usará as seguintes regras para aplicar as alterações:

1. Se a matriz `multiValue` existir e tiver qualquer modificação, então essa modificação é aplicada. O primeiro elemento na propriedade `value` é ignorado.
2. Caso contrário, qualquer modificação na propriedade `value` será aplicada e os valores subsequentes (se existirem) permanecerão inalterados.

A propriedade `multiValue` é usada somente quando a solicitação ou resposta HTTP contém cadeias de consulta duplicadas, cabeçalhos ou cookies com o mesmo nome, como mostrado no exemplo anterior. No entanto, se houver vários valores em uma única cadeia de consulta, cabeçalho ou cookie, a propriedade `multiValue` não será usada.

Example Exemplo

Pense em uma solicitação com um cabeçalho `Accept` que contém três valores.

```
Accept: application/json, application/xml, text/html
```

Esse cabeçalho é representado da forma a seguir no objeto `request`.

```
"headers": {  
  "accept": {  
    "value": "application/json, application/xml, text/html"  
  }  
}
```


Atributos de cookies

Em um cabeçalho Set-Cookie em uma resposta HTTP, o cabeçalho contém o par nome-valor para o cookie e, opcionalmente, um conjunto de atributos separados por ponto-e-vírgula.

Example Exemplo

```
Set-Cookie: cookie1=val1; Secure; Path=/; Domain=example.com; Expires=Wed, 05 Apr 2021
07:28:00 GMT
```

No objeto `response`, esses atributos são representados na propriedade `attributes` do campo `cookie`. Por exemplo, o cabeçalho Set-Cookie anterior é representado da seguinte forma:

```
"cookie1": {
  "value": "val1",
  "attributes": "Secure; Path=/; Domain=example.com; Expires=Wed, 05 Apr 2021
07:28:00 GMT"
}
```

Exemplo de objeto de resposta

O exemplo a seguir mostra um objeto `response`, a saída de uma função de resposta do visualizador, no qual o corpo foi substituído por uma função de resposta do visualizador.

```
{
  "response": {
    "statusCode": 200,
    "statusDescription": "OK",
    "headers": {
      "date": {
        "value": "Mon, 04 Apr 2021 18:57:56 GMT"
      },
      "server": {
        "value": "unicorn/19.9.0"
      },
      "access-control-allow-origin": {
        "value": "*"
      },
      "access-control-allow-credentials": {
        "value": "true"
      },
    },
  },
}
```

```
    "content-type": {
      "value": "text/html"
    },
    "content-length": {
      "value": "86"
    }
  },
  "cookies": {
    "ID": {
      "value": "id1234",
      "attributes": "Expires=Wed, 05 Apr 2021 07:28:00 GMT"
    },
    "Cookie1": {
      "value": "val1",
      "attributes": "Secure; Path=/; Domain=example.com; Expires=Wed, 05 Apr 2021
07:28:00 GMT",
      "multiValue": [
        {
          "value": "val1",
          "attributes": "Secure; Path=/; Domain=example.com; Expires=Wed, 05 Apr 2021
07:28:00 GMT"
        },
        {
          "value": "val2",
          "attributes": "Path=/cat; Domain=example.com; Expires=Wed, 10 Jan 2021
07:28:00 GMT"
        }
      ]
    }
  },
  // Adding the body field is optional and it will not be present in the response
  // object
  // unless you specify it in your function.
  // Your function does not have access to the original body returned by the
  // CloudFront
  // cache or origin.
  // If you don't specify the body field in your viewer response function, the
  // original
  // body returned by the CloudFront cache or origin is returned to viewer.

  "body": {
    "encoding": "text",
```

```
    "data": "<!DOCTYPE html><html><body><p>Here is your custom content.</p></body></html>"
  }
}
```

Exemplo de objeto de evento

O exemplo a seguir mostra um objeto event completo.

Note

O objeto event é a entrada para sua função. Sua função retorna apenas o objeto request ou response, não o objeto event completo.

```
{
  "version": "1.0",
  "context": {
    "distributionDomainName": "d1111111abcdef8.cloudfront.net",
    "distributionId": "EDFDVBD6EXAMPLE",
    "eventType": "viewer-response",
    "requestId": "EXAMPLEentjQpEXAMPLE_SG5Z-EXAMPLEPmPfEXAMPLEu3EqEXAMPLE=="
  },
  "viewer": {"ip": "198.51.100.11"},
  "request": {
    "method": "GET",
    "uri": "/media/index.mpd",
    "querystring": {
      "ID": {"value": "42"},
      "Exp": {"value": "1619740800"},
      "TTL": {"value": "1440"},
      "NoValue": {"value": ""},
      "querymv": {
        "value": "val1",
        "multiValue": [
          {"value": "val1"},
          {"value": "val2,val3"}
        ]
      }
    }
  },
  "headers": {
```

```
    "host": {"value": "video.example.com"},
    "user-agent": {"value": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:83.0)
Gecko/20100101 Firefox/83.0"},
    "accept": {
      "value": "application/json",
      "multiValue": [
        {"value": "application/json"},
        {"value": "application/xml"},
        {"value": "text/html"}
      ]
    },
    "accept-language": {"value": "en-GB,en;q=0.5"},
    "accept-encoding": {"value": "gzip, deflate, br"},
    "origin": {"value": "https://website.example.com"},
    "referer": {"value": "https://website.example.com/videos/12345678?
action=play"},
    "cloudfront-viewer-country": {"value": "GB"}
  },
  "cookies": {
    "Cookie1": {"value": "value1"},
    "Cookie2": {"value": "value2"},
    "cookie_consent": {"value": "true"},
    "cookiemv": {
      "value": "value3",
      "multiValue": [
        {"value": "value3"},
        {"value": "value4"}
      ]
    }
  }
},
"response": {
  "statusCode": 200,
  "statusDescription": "OK",
  "headers": {
    "date": {"value": "Mon, 04 Apr 2021 18:57:56 GMT"},
    "server": {"value": "unicorn/19.9.0"},
    "access-control-allow-origin": {"value": "*"},
    "access-control-allow-credentials": {"value": "true"},
    "content-type": {"value": "application/json"},
    "content-length": {"value": "701"}
  },
  "cookies": {
    "ID": {
```


- O módulo criptográfico tem as seguintes alterações:
 - `hash.digest()`: o tipo de retorno será alterado para `Buffer` se nenhuma codificação for fornecida.
 - `hmac.digest()`: o tipo de retorno será alterado para `Buffer` se nenhuma codificação for fornecida.
- Para ter mais informações sobre novos recursos adicionais, consulte [Recursos de runtime 2.0 do JavaScript para CloudFront Functions](#).

Tópicos

- [Recursos de runtime 1.0 do JavaScript para CloudFront Functions](#)
- [Recursos de runtime 2.0 do JavaScript para CloudFront Functions](#)

Recursos de runtime 1.0 do JavaScript para CloudFront Functions

O ambiente de runtime do JavaScript do CloudFront Functions é compatível com o [ECMAScript \(ES\) versão 5.1](#) e também é compatível com alguns recursos das versões ES 6 a 9. Ele também fornece alguns métodos não padronizados que não fazem parte das especificações ES.

Os tópicos a seguir listam todos os recursos de idioma compatíveis.

Tópicos

- [Recursos principais](#)
- [Objetos primitivos](#)
- [Objetos integrados](#)
- [Tipos de erro](#)
- [Variáveis globais](#)
- [Módulos integrados](#)
- [Recursos restritos](#)

Recursos principais

Os seguintes recursos principais do ES são compatíveis.

Tipos

Todos os tipos ES 5.1 são compatíveis. Isso inclui valores booleanos, números, cadeias de caracteres, objetos, matrizes, funções, construtores de funções e expressões regulares.

Operadores

Todos os operadores ES 5.1 são compatíveis.

O operador de exponenciação ES 7 (**) é compatível.

Declarações

Note

As instruções `const` e `let` não são compatíveis.

As seguintes instruções ES 5.1 são compatíveis:

- `break`
- `catch`
- `continue`
- `do-while`
- `else`
- `finally`
- `for`
- `for-in`
- `if`
- `return`
- `switch`
- `throw`
- `try`
- `var`
- `while`
- Instruções rotuladas

Literais

Os literais de modelo ES 6 são compatíveis: cadeias de várias linhas, interpolação de expressão e modelos de aninhamento.

Funções

Todos os recursos de função ES 5.1 são compatíveis.

As funções de seta ES 6 são compatíveis, assim como a sintaxe de parâmetro de descanso ES 6.

Unicode

Texto de origem e literais de cadeias de caracteres podem conter caracteres codificados em Unicode. Sequências de escape de ponto de código Unicode de seis caracteres (por exemplo, `\uXXXX`) também são compatíveis.

Modo estrito

As funções operam no modo estrito por padrão, então você não precisa adicionar uma instrução `use strict` ao seu código de função. Elas não podem ser alteradas.

Objetos primitivos

Os seguintes objetos primitivos de ES são compatíveis.

Objeto

Os seguintes métodos ES 5.1 em objetos são compatíveis:

- `create` (sem lista de propriedades)
- `defineProperties`
- `defineProperty`
- `freeze`
- `getOwnPropertyDescriptor`
- `getOwnPropertyNames`
- `getPrototypeOf`
- `hasOwnProperty`
- `isExtensible`
- `isFrozen`

- `prototype.isPrototypeOf`
- `isSealed`
- `keys`
- `preventExtensions`
- `prototype.propertyIsEnumerable`
- `seal`
- `prototype.toString`
- `prototype.valueOf`

Os seguintes métodos ES 6 em objetos são compatíveis:

- `assign`
- `is`
- `prototype.setPrototypeOf`

Os seguintes métodos ES 8 em objetos são compatíveis:

- `entries`
- `values`

String

Os seguintes métodos ES 5.1 em cadeias de caracteres são compatíveis:

- `fromCharCode`
- `prototype.charAt`
- `prototype.concat`
- `prototype.indexOf`
- `prototype.lastIndexOf`
- `prototype.match`
- `prototype.replace`
- `prototype.search`
- `prototype.slice`
- `prototype.split`
- `prototype.substr`

- `prototype.substring`
- `prototype.toLowerCase`
- `prototype.trim`
- `prototype.toUpperCase`

Os seguintes métodos ES 6 em cadeias de caracteres são compatíveis:

- `fromCodePoint`
- `prototype.codePointAt`
- `prototype.endsWith`
- `prototype.includes`
- `prototype.repeat`
- `prototype.startsWith`

Os seguintes métodos ES 8 em cadeias de caracteres são compatíveis:

- `prototype.padStart`
- `prototype.padEnd`

Os seguintes métodos ES 9 em cadeias de caracteres são compatíveis:

- `prototype.trimStart`
- `prototype.trimEnd`

Os seguintes métodos não padronizados em cadeias de caracteres são compatíveis:

- `prototype.bytesFrom(array | string, encoding)`

Cria uma cadeia de caracteres de bytes de uma matriz de octetos ou uma cadeia de caracteres codificada. As opções de codificação de cadeia de caracteres são `hex`, `base64` e `base64url`.

- `prototype.fromBytes(start[, end])`

Cria uma cadeia de caracteres Unicode de uma cadeia de caracteres de bytes em que cada byte é substituído pelo ponto de código Unicode correspondente.

- `prototype.fromUTF8(start[, end])`

Cria uma cadeia de caracteres Unicode de uma cadeia de caracteres de bytes codificada UTF-8. Se a codificação estiver incorreta, ela retorna `null`.

- `prototype.toBytes(start[, end])`

Cria uma cadeia de caracteres de bytes de uma cadeia de caracteres Unicode. Todos os caracteres devem estar no intervalo de [0.255]. Se não, ele retorna `null`.

- `prototype.toUTF8(start[, end])`

Cria uma cadeia de caracteres de bytes codificada UTF-8 de uma cadeia de caracteres Unicode.

telefone

Todos os métodos ES 5.1 em números são compatíveis.

Os seguintes métodos ES 6 em números são compatíveis:

- `isFinite`
- `isInteger`
- `isNaN`
- `isSafeInteger`
- `parseFloat`
- `parseInt`
- `prototype.toExponential`
- `prototype.toFixed`
- `prototype.toPrecision`
- `EPSILON`
- `MAX_SAFE_INTEGER`
- `MAX_VALUE`
- `MIN_SAFE_INTEGER`
- `MIN_VALUE`
- `NEGATIVE_INFINITY`
- `NaN`
- `POSITIVE_INFINITY`

Objetos integrados

Os seguintes objetos integrados do ES são compatíveis.

Matemática

Todos os métodos matemáticos ES 5.1 são compatíveis.

Note

No ambiente de tempo de execução do CloudFront Functions, a implementação `Math.random()` usa o OpenBSD `arc4random` propagado com o carimbo de data/hora de quando a função é executada.

Os seguintes métodos matemáticos ES 6 são compatíveis:

- `acosh`
- `asinh`
- `atanh`
- `cbrt`
- `clz32`
- `cosh`
- `expm1`
- `fround`
- `hypot`
- `imul`
- `log10`
- `log1p`
- `log2`
- `sign`
- `sinh`
- `tanh`
- `trunc`
- `E`
- `LN10`
- `LN2`
- `LOG10E`

- LOG2E
- PI
- SQRT1_2
- SQRT2

Data

Todos os recursos Date do ES 5.1 são compatíveis.

Note

Por razões de segurança, Date sempre retorna o mesmo valor, que seria o horário de início da função, durante o tempo de vida de uma única execução de função. Para ter mais informações, consulte [Recursos restritos](#).

Função

Os métodos `apply`, `bind` e `call` são compatíveis.

Os construtores de função não são compatíveis.

Expressões regulares

Todos os recursos de expressão regular ES 5.1 são compatíveis. A linguagem de expressão regular é compatível com Perl. Os grupos de captura nomeados ES 9 são compatíveis.

JSON

Todos os recursos JSON ES 5.1 são compatíveis, incluindo `parse` e `stringify`.

Array

Os seguintes métodos ES 5.1 em matrizes são compatíveis:

- `isArray`
- `prototype.concat`
- `prototype.every`
- `prototype.filter`
- `prototype.forEach`
- `prototype.indexOf`
- `prototype.join`

- `prototype.lastIndexOf`
- `prototype.map`
- `prototype.pop`
- `prototype.push`
- `prototype.reduce`
- `prototype.reduceRight`
- `prototype.reverse`
- `prototype.shift`
- `prototype.slice`
- `prototype.some`
- `prototype.sort`
- `prototype.splice`
- `prototype.unshift`

Os seguintes métodos ES 6 em matrizes são compatíveis:

- `of`
- `prototype.copyWithIn`
- `prototype.fill`
- `prototype.find`
- `prototype.findIndex`

Os seguintes métodos ES 7 em matrizes são compatíveis:

- `prototype.includes`

Matrizes digitadas

As seguintes matrizes ES 6 digitadas são compatíveis:

- `Int8Array`
- `Uint8Array`
- `Uint8ClampedArray`
- `Int16Array`
- `Uint16Array`
- `Int32Array`

- `Uint32Array`
- `Float32Array`
- `Float64Array`
- `prototype.copyWithIn`
- `prototype.fill`
- `prototype.join`
- `prototype.set`
- `prototype.slice`
- `prototype.subarray`
- `prototype.toString`

ArrayBuffer

Os seguintes métodos em `ArrayBuffer` são compatíveis:

- `prototype.isView`
- `prototype.slice`

Promessa

Os seguintes métodos em `promises` são compatíveis:

- `reject`
- `resolve`
- `prototype.catch`
- `prototype.finally`
- `prototype.then`

Criptografia

O módulo criptográfico fornece auxiliares de código de autenticação de mensagens (HMAC) padrão e hash. Você pode carregar o módulo usando `require('crypto')`. O módulo expõe os seguintes métodos que se comportam exatamente como suas contrapartes Node.js:

- `createHash(algorithm)`
- `hash.update(data)`
- `hash.digest([encoding])`
- `createHmac(algorithm, secret key)`
- `hmac.update(data)`

- `hmac.digest([encoding])`

Para mais informações, consulte [Criptográficos \(hash e HMAC\)](#) na seção de módulos integrados.

Console

Este é um objeto auxiliar para depuração. Ele só é compatível com o método `log()` para gravar mensagens de log.

Note

O CloudFront Functions não é compatível com sintaxe de vírgula, como `console.log('a', 'b')`. Em vez disso, use o formato `console.log('a' + ' ' + 'b')`.

Tipos de erro

Os seguintes objetos de erro são compatíveis:

- `Error`
- `EvalError`
- `InternalError`
- `MemoryError`
- `RangeError`
- `ReferenceError`
- `SyntaxError`
- `TypeError`
- `URIError`

Variáveis globais

O objeto `globalThis` é compatível.

As seguintes funções globais do ES 5.1 são compatíveis:

- `decodeURI`
- `decodeURIComponent`

- `encodeURIComponent`
- `encodeURIComponent`
- `isFinite`
- `isNaN`
- `parseFloat`
- `parseInt`

As seguintes restrições globais são válidas:

- `NaN`
- `Infinity`
- `undefined`

Módulos integrados

Os seguintes módulos integrados são compatíveis:

Modules

- [Criptográficos \(hash e HMAC\)](#)
- [String de consulta](#)

Criptográficos (hash e HMAC)

O módulo criptográfico (`crypto`) fornece auxiliares de código de autenticação de mensagens (HMAC) padrão e hash. Você pode carregar o módulo usando `require('crypto')`. O módulo fornece os seguintes métodos que se comportam exatamente como suas contrapartes Node.js.

Métodos de hash

`crypto.createHash(algorithm)`

Cria e retorna um objeto hash que você pode usar para gerar resumos de hash usando o algoritmo fornecido: `md5`, `sha1` ou `sha256`.

`hash.update(data)`

Atualiza o conteúdo de hash com os `data` fornecidos.

```
hash.digest([encoding])
```

Calcula o resumo de todos os dados passados usando `hash.update()`. A codificação pode ser `hex`, `base64` ou `base64url`.

Métodos de HMAC

```
crypto.createHmac(algorithm, secret key)
```

Cria e retorna um objeto HMAC que usa o `algorithm` e a `secret key` fornecidos. O algoritmo pode ser `md5`, `sha1` ou `sha256`.

```
hmac.update(data)
```

Atualiza o conteúdo HMAC com os `data` fornecidos.

```
hmac.digest([encoding])
```

Calcula o resumo de todos os dados passados usando `hmac.update()`. A codificação pode ser `hex`, `base64` ou `base64url`.

String de consulta

Note

O [objeto de evento CloudFront Functions](#) analisa automaticamente as cadeias de consulta de URL para você. Isso significa que na maioria dos casos você não precisa usar este módulo.

O módulo cadeia de consulta (`querystring`) fornece métodos para analisar e formatar cadeias de consulta de URL. Você pode carregar o módulo usando `require('querystring')`. O módulo fornece os seguintes métodos:

```
querystring.escape(string)
```

O URL codifica a `string` fornecida, retornando uma cadeia de consulta escapada. O método é usado por `querystring.stringify()` e não deve ser usado diretamente.

```
querystring.parse(string[, separator[, equal[, options]])
```

Analisa uma cadeia de consulta (`string`) e retorna um objeto.

O parâmetro `separator` é uma substring para delimitar pares de chaves e valores na cadeia de consulta. O padrão é `&`.

O parâmetro `equal` é uma substring para delimitar chaves e valores na cadeia de consulta. O padrão é `=`.

O parâmetro `options` é um objeto com as seguintes chaves:

`decodeURIComponent` *function*

Uma função para decodificar caracteres codificados por percentual na cadeia de consulta. O padrão é `querystring.unescape()`.

`maxKeys` *number*

O número máximo de chaves a serem analisadas. O padrão é `1000`. Use um valor de `0` para remover as limitações para as chaves de contagem.

Por padrão, os caracteres codificados por percentual dentro da cadeia de consulta são assumidos para usar a codificação UTF-8. Sequências UTF-8 inválidas são substituídas pelo caractere de substituição U+FFFD.

Por exemplo, para a seguinte cadeia de consulta:

```
'name=value&abc=xyz&abc=123'
```

O valor de retorno de `querystring.parse()` é:

```
{
  name: 'value',
  abc: ['xyz', '123']
}
```

`querystring.decode()` é um alias para `querystring.parse()`.

`querystring.stringify(object[, separator[, equal[, options]])`

Serializa um `object` e retorna uma cadeia de consulta.

O parâmetro `separator` é uma substring para delimitar pares de chaves e valores na cadeia de consulta. O padrão é `&`.

O parâmetro `equal` é uma substring para delimitar chaves e valores na cadeia de consulta. O padrão é `=`.

O parâmetro `options` é um objeto com as seguintes chaves:

`encodeURIComponent` *function*

A função a ser usada para converter caracteres URL-inseguros para codificação percentual na cadeia de consulta. O padrão é `querystring.escape()`.

Por padrão, os caracteres que exigem percentual de codificação dentro da cadeia de consulta são codificados como UTF-8. Para usar uma codificação diferente, especifique a opção `encodeURIComponent`.

Por exemplo, para o seguinte código:

```
querystring.stringify({ name: 'value', abc: ['xyz', '123'], anotherName: '' });
```

O valor de retorno é:

```
'name=value&abc=xyz&abc=123&anotherName='
```

`querystring.encode()` é um alias para `querystring.stringify()`.

`querystring.unescape(string)`

Decodifica caracteres codificados por porcentagem de URL na `string` fornecida, retornando uma cadeia de consulta sem escapamento. Esse método é usado por `querystring.parse()` e não deve ser usado diretamente.

Recursos restritos

Os seguintes recursos de linguagem JavaScript não são compatíveis ou são restritos devido a preocupações de segurança.

Avaliação dinâmica do código

A avaliação dinâmica do código não é compatível. Ambos os construtores `eval()` e `Function` lançam um erro se forem tentados. Por exemplo, `const sum = new Function('a', 'b', 'return a + b')` lança um erro.

Temporizadores

As funções `setTimeout()`, `setImmediate()` e `clearTimeout()` não são compatíveis. Não há provisão para adiar ou renderizar dentro de uma execução de função. Sua função deve ser executada de forma síncrona até a conclusão.

Data e carimbos de data/hora

Por razões de segurança, não há acesso a temporizadores de alta resolução. Todos os métodos `Date` para consultar a hora atual sempre retornam o mesmo valor durante o tempo de vida de uma única execução de função. O carimbo de data/hora retornado é o momento em que a função começou a ser executada. Consequentemente, você não pode medir o tempo decorrido em sua função.

Acesso ao sistema de arquivos

Não há acesso ao sistema de arquivos. Por exemplo, não há módulo `fs` para acesso ao sistema de arquivos como no `Node.js`.

Acesso a processos

Não há acesso a processos. Por exemplo, não há um objeto global `process` para processar o acesso às informações como no `Node.js`.

Variáveis de ambiente

Não há acesso às variáveis de ambiente.

Em vez disso, é possível usar o `KeyValueStore` do `CloudFront` para criar um `datastore` centralizado de pares de chave-valor para o `CloudFront Functions`. O `KeyValueStore` do `CloudFront` permite atualizações dinâmicas nos dados de configuração sem a necessidade de implantar alterações no código. É necessário usar o [JavaScript runtime 2.0](#) para usar o `KeyValueStore` do `CloudFront`. Para ter mais informações, consulte [KeyValueStore do Amazon CloudFront](#).

Acesso à rede

Não há suporte para chamadas de rede. Por exemplo, `XHR`, `HTTP (S)` e `socket` não são compatíveis.

Recursos de runtime 2.0 do JavaScript para CloudFront Functions

O ambiente de runtime do JavaScript do `CloudFront Functions` é compatível com o [ECMAScript \(ES\) versão 5.1](#) e também é compatível com alguns recursos das versões 6 a 12 do ES. Ele também

fornece alguns métodos não padronizados que não fazem parte das especificações ES. Os tópicos a seguir listam todos os recursos compatíveis nesse runtime.

Tópicos

- [Recursos principais](#)
- [Objetos primitivos](#)
- [Objetos integrados](#)
- [Tipos de erro](#)
- [Variáveis globais](#)
- [Módulos integrados](#)
- [Recursos restritos](#)

Recursos principais

Os seguintes recursos principais do ES são compatíveis.

Tipos

Todos os tipos ES 5.1 são compatíveis. Isso inclui valores booleanos, números, strings, objetos, matrizes, funções e expressões regulares.

Operadores

Todos os operadores ES 5.1 são compatíveis.

O operador de exponenciação ES 7 (**) é compatível.

Declarações

As seguintes instruções ES 5.1 são compatíveis:

- `break`
- `catch`
- `continue`
- `do-while`
- `else`
- `finally`
- `for`
- `for-in`

- `if`
- `label`
- `return`
- `switch`
- `throw`
- `try`
- `var`
- `while`

As seguintes declarações do ES 6 são compatíveis:

- `async`
- `await`
- `const`
- `let`



Note

`async`, `await`, `const` e `let` são recursos novos no runtime 2.0 do JavaScript.

Literais

Os literais de modelo ES 6 são compatíveis: cadeias de várias linhas, interpolação de expressão e modelos de aninhamento.

Funções

Todos os recursos de função ES 5.1 são compatíveis.

As funções de seta ES 6 são compatíveis, assim como a sintaxe de parâmetro de descanso ES 6.

Unicode

Texto de origem e literais de cadeias de caracteres podem conter caracteres codificados em Unicode. Sequências de escape de ponto de código Unicode de seis caracteres (por exemplo, `\uXXXX`) também são compatíveis.

Modo estrito

As funções operam no modo estrito por padrão, então você não precisa adicionar uma instrução `use strict` ao seu código de função. Elas não podem ser alteradas.

Objetos primitivos

Os seguintes objetos primitivos de ES são compatíveis.

Objeto

Os seguintes métodos ES 5.1 em objetos são compatíveis:

- `Object.create()` (sem lista de propriedades)
- `Object.defineProperties()`
- `Object.defineProperty()`
- `Object.freeze()`
- `Object.getOwnPropertyDescriptor()`
- `Object.getOwnPropertyDescriptors()`
- `Object.getOwnPropertyNames()`
- `Object.getPrototypeOf()`
- `Object.isExtensible()`
- `Object.isFrozen()`
- `Object.isSealed()`
- `Object.keys()`
- `Object.preventExtensions()`
- `Object.seal()`

Os seguintes métodos ES 6 em objetos são compatíveis:

- `Object.assign()`

Os seguintes métodos ES 8 em objetos são compatíveis:

- `Object.entries()`
- `Object.values()`

Os seguintes métodos de protótipo do ES 5.1 em objetos são compatíveis:

- `Object.prototype.hasOwnProperty()`
- `Object.prototype.isPrototypeOf()`
- `Object.prototype.propertyIsEnumerable()`
- `Object.prototype.toString()`

- `Object.prototype.valueOf()`

Os seguintes métodos de protótipo do ES 6 em objetos são compatíveis:

- `Object.prototype.is()`
- `Object.prototype.setPrototypeOf()`

String

Os seguintes métodos ES 5.1 em cadeias de caracteres são compatíveis:

- `String.fromCharCode()`

Os seguintes métodos ES 6 em cadeias de caracteres são compatíveis:

- `String.fromCodePoint()`

Os seguintes métodos de protótipo do ES 5.1 em strings são compatíveis:

- `String.prototype.charAt()`
- `String.prototype.concat()`
- `String.prototype.indexOf()`
- `String.prototype.lastIndexOf()`
- `String.prototype.match()`
- `String.prototype.replace()`
- `String.prototype.search()`
- `String.prototype.slice()`
- `String.prototype.split()`
- `String.prototype.substr()`
- `String.prototype.substring()`
- `String.prototype.toLowerCase()`
- `String.prototype.trim()`
- `String.prototype.toUpperCase()`

Os seguintes métodos de protótipo do ES 6 em strings são compatíveis:

- `String.prototype.codePointAt()`
- `String.prototype.endsWith()`
- `String.prototype.includes()`

- `String.prototype.repeat()`
- `String.prototype.startsWith()`

Os seguintes métodos de protótipo do ES 8 em strings são compatíveis:


- `String.prototype.padStart()`
- `String.prototype.padEnd()`

Os seguintes métodos de protótipo do ES 9 em strings são compatíveis:

- `String.prototype.trimStart()`
- `String.prototype.trimEnd()`

Os seguintes métodos de protótipo do ES 12 em strings são compatíveis:

- `String.prototype.replaceAll()`

 Note

`String.prototype.replaceAll()` é um recurso novo no runtime 2.0 do JavaScript.

Número

Todos os números do ES 5 são compatíveis.

As seguintes propriedades do ES 6 em números são compatíveis:

- `Number.EPSILON`
- `Number.MAX_SAFE_INTEGER`
- `Number.MIN_SAFE_INTEGER`
- `Number.MAX_VALUE`
- `Number.MIN_VALUE`
- `Number.NaN`
- `Number.NEGATIVE_INFINITY`
- `Number.POSITIVE_INFINITY`

Os seguintes métodos ES 6 em números são compatíveis:


- `Number.isFinite()`
- `Number.isInteger()`

- `Number.isNaN()`
- `Number.isSafeInteger()`
- `Number.parseInt()`
- `Number.parseFloat()`

Os seguintes métodos de protótipo do ES 5.1 em números são compatíveis:

- `Number.prototype.toExponential()`
- `Number.prototype.toFixed()`
- `Number.prototype.toPrecision()`

Separadores numéricos ES 12 são compatíveis.

 Note


Os separadores numéricos ES 12 são novos no runtime 2.0 do JavaScript.

Objetos integrados

Os seguintes objetos integrados do ES são compatíveis.

Matemática

Todos os métodos matemáticos ES 5.1 são compatíveis.

 Note

No ambiente de tempo de execução do CloudFront Functions, a implementação `Math.random()` usa o OpenBSD `arc4random` propagado com o carimbo de data/hora de quando a função é executada.

As seguintes propriedades matemáticas do ES 6 são compatíveis:

- `Math.E`
- `Math.LN10`
- `Math.LN2`
- `Math.LOG10E`

- `Math.LOG2E`
- `Math.PI`
- `Math.SQRT1_2`
- `Math.SQRT2`

Os seguintes métodos matemáticos ES 6 são compatíveis:

- `Math.abs()`
- `Math.acos()`
- `Math.acosh()`
- `Math.asin()`
- `Math.asinh()`
- `Math.atan()`
- `Math.atan2()`
- `Math.atanh()`
- `Math.cbrt()`
- `Math.ceil()`
- `Math.clz32()`
- `Math.cos()`
- `Math.cosh()`
- `Math.exp()`
- `Math.expm1()`
- `Math.floor()`
- `Math.fround()`
- `Math.hypot()`
- `Math.imul()`
- `Math.log()`
- `Math.log1p()`
- `Math.log2()`
- `Math.log10()`
- `Math.max()`
- `Math.min()`

- `Math.pow()`
- `Math.random()`
- `Math.round()`
- `Math.sign()`
- `Math.sinh()`
- `Math.sin()`
- `Math.sqrt()`
- `Math.tan()`
- `Math.tanh()`
- `Math.trunc()`

Data

Todos os recursos Date do ES 5.1 são compatíveis.

Note

Por razões de segurança, Date sempre retorna o mesmo valor, que seria o horário de início da função, durante o tempo de vida de uma única execução de função. Para ter mais informações, consulte [Recursos restritos](#).

Função

Os seguintes métodos de protótipo do ES 5.1 são compatíveis:

- `Function.prototype.apply()`
- `Function.prototype.bind()`
- `Function.prototype.call()`


Os construtores de função não são compatíveis.

Expressões regulares

Todos os recursos de expressão regular ES 5.1 são compatíveis. A linguagem de expressão regular é compatível com Perl.

As seguintes propriedades do acessador de protótipos do ES 5.1 são compatíveis:


- `RegExp.prototype.global`
- `RegExp.prototype.ignoreCase`
- `RegExp.prototype.multiline`
- `RegExp.prototype.source`
- `RegExp.prototype.sticky`
- `RegExp.prototype.flags`

 Note

`RegExp.prototype.sticky` e `RegExp.prototype.flags` são recursos novos no runtime 2.0 do JavaScript.

Os seguintes métodos de protótipo do ES 5.1 são compatíveis:

- `RegExp.prototype.exec()`
- `RegExp.prototype.test()`
- `RegExp.prototype.toString()`
- `RegExp.prototype[@@replace]()`
- `RegExp.prototype[@@split]()`

 Note

`RegExp.prototype[@@split]()` é um recurso novo no runtime 2.0 do JavaScript.

As seguintes propriedades de instância do ES 5.1 são compatíveis:

- `lastIndex`

Os grupos de captura nomeados ES 9 são compatíveis.

JSON

Os seguintes métodos do ES 5.1 são compatíveis:

- `JSON.parse()`
- `JSON.stringify()`

Array

Os seguintes métodos ES 5.1 em matrizes são compatíveis:

- `Array.isArray()`

Os seguintes métodos ES 6 em matrizes são compatíveis:

- `Array.of()`

Os seguintes métodos de protótipo do ES 5.1 são compatíveis:

- `Array.prototype.concat()`
- `Array.prototype.every()`
- `Array.prototype.filter()`
- `Array.prototype.forEach()`
- `Array.prototype.indexOf()`
- `Array.prototype.join()`
- `Array.prototype.lastIndexOf()`
- `Array.prototype.map()`
- `Array.prototype.pop()`
- `Array.prototype.push()`
- `Array.prototype.reduce()`
- `Array.prototype.reduceRight()`
- `Array.prototype.reverse()`
- `Array.prototype.shift()`
- `Array.prototype.slice()`
- `Array.prototype.some()`
- `Array.prototype.sort()`
- `Array.prototype.splice()`
- `Array.prototype.unshift()`

Os seguintes métodos de protótipo do ES 6 são compatíveis

- `Array.prototype.copyWithin()`
- `Array.prototype.fill()`
- `Array.prototype.find()`
- `Array.prototype.findIndex()`

Os seguintes métodos de protótipo do ES 7 são compatíveis:

- `Array.prototype.includes()`

Matrizes digitadas

Os seguintes construtores de matriz de tipo do ES 6 são compatíveis:

- `Float32Array`
- `Float64Array`
- `Int8Array`
- `Int16Array`
- `Int32Array`
- `Uint8Array`
- `Uint8ClampedArray`
- `Uint16Array`
- `Uint32Array`

Os seguintes métodos do ES 6 são compatíveis:

- `TypedArray.from()`
- `TypedArray.of()`


Note

`TypedArray.from()` e `TypedArray.of()` são recursos novos no runtime 2.0 do JavaScript.

Os seguintes métodos de protótipo do ES 6 são compatíveis

- `TypedArray.prototype.copyWithin()`
- `TypedArray.prototype.every()`
- `TypedArray.prototype.fill()`
- `TypedArray.prototype.filter()`
- `TypedArray.prototype.find()`
- `TypedArray.prototype.findIndex()`
- `TypedArray.prototype.forEach()`
- `TypedArray.prototype.includes()`
- `TypedArray.prototype.indexOf()`

- `TypedArray.prototype.join()`
- `TypedArray.prototype.lastIndexOf()`
- `TypedArray.prototype.map()`
- `TypedArray.prototype.reduce()`
- `TypedArray.prototype.reduceRight()`
- `TypedArray.prototype.reverse()`
- `TypedArray.prototype.some()`
- `TypedArray.prototype.set()`
- `TypedArray.prototype.slice()`
- `TypedArray.prototype.sort()`
- `TypedArray.prototype.subarray()`
- `TypedArray.prototype.toString()`

 Note

`TypedArray.prototype.every()`, `TypedArray.prototype.fill()`, `TypedArray.prototype.filter()`, `TypedArray.prototype.find()`, `TypedArray.prototype.findIndex()`, `TypedArray.prototype.forEach()`, `TypedArray.prototype.includes()`, `TypedArray.prototype.indexOf()`, `TypedArray.prototype.join()`, `TypedArray.prototype.lastIndexOf()`, `TypedArray.prototype.map()`, `TypedArray.prototype.reduce()`, `TypedArray.prototype.reduceRight()`, `TypedArray.prototype.reverse()` e `TypedArray.prototype.some()` são recursos novos no runtime 2.0 do JavaScript.

ArrayBuffer

Os seguintes métodos do ES 6 em `ArrayBuffer` são compatíveis:

- `isView()`


Os seguintes métodos de protótipo do ES 6 em `ArrayBuffer` são compatíveis:

- `ArrayBuffer.prototype.slice()`

Promessa

Os seguintes métodos do ES 6 em `promises` são compatíveis:

- `Promise.all()`
- `Promise.allSettled()`
- `Promise.any()`
- `Promise.reject()`
- `Promise.resolve()`
- `Promise.race()`

 Note

`Promise.all()`, `Promise.allSettled()`, `Promise.any()` e `Promise.race()` são recursos novos no runtime 2.0 do JavaScript.

Os seguintes métodos de protótipo do ES 6 em promises são compatíveis:


- `Promise.prototype.catch()`
- `Promise.prototype.finally()`
- `Promise.prototype.then()`

DataView

Os seguintes métodos de protótipo do ES 6 são compatíveis:

- `DataView.prototype.getFloat32()`
- `DataView.prototype.getFloat64()`
- `DataView.prototype.getInt16()`
- `DataView.prototype.getInt32()`
- `DataView.prototype.getInt8()`
- `DataView.prototype.getUint16()`
- `DataView.prototype.getUint32()`
- `DataView.prototype.getUint8()`
- `DataView.prototype.setFloat32()`
- `DataView.prototype.setFloat64()`
- `DataView.prototype.setInt16()`
- `DataView.prototype.setInt32()`
- `DataView.prototype.setInt8()`

- `DataView.prototype.setUint16()`
- `DataView.prototype.setUint32()`
- `DataView.prototype.setUint8()`


 Note

Todos os métodos de protótipo do DataView ES 6 são novos no runtime 2.0 do JavaScript.

Símbolo

Os seguintes métodos do ES 6 são compatíveis:

- `Symbol.for()`
- `Symbol.keyfor()`

 Note

Todos os métodos do Symbol ES 6 são novos no runtime 2.0 do JavaScript.

Decodificador de texto

Os seguintes métodos de protótipo são compatíveis:

- `TextDecoder.prototype.decode()`

As seguintes propriedades do acessador de protótipos são compatíveis:

- `TextDecoder.prototype.encoding`
- `TextDecoder.prototype.fatal`
- `TextDecoder.prototype.ignoreBOM`

Codificador de texto

Os seguintes métodos de protótipo são compatíveis:

- `TextEncoder.prototype.encode()`
- `TextEncoder.prototype.encodeInto()`

Tipos de erro

Os seguintes objetos de erro são compatíveis:

- `Error`
- `EvalError`
- `InternalError`
- `RangeError`
- `ReferenceError`
- `SyntaxError`
- `TypeError`
- `URIError`

Variáveis globais

O objeto `globalThis` é compatível.

As seguintes funções globais do ES 5.1 são compatíveis:

- `decodeURI()`
- `decodeURIComponent()`
- `encodeURI()`
- `encodeURIComponent()`
- `isFinite()`
- `isNaN()`
- `parseFloat()`
- `parseInt()`

As seguintes funções globais do ES 6 são compatíveis:

- `atob()`
- `btoa()`

Note

`atob()` e `btoa()` são recursos novos no runtime 2.0 do JavaScript.

As seguintes restrições globais são válidas:

- NaN
- Infinity
- undefined
- arguments

Módulos integrados

Os seguintes módulos integrados são compatíveis:

Modules

- [Buffer](#)
- [String de consulta](#)
- [Criptografia](#)

Buffer

O módulo oferece os seguintes métodos:

- `Buffer.alloc(size[, fill[, encoding]])`

Aloque um `Buffer`.

- `size`: tamanho do buffer. Insira um número inteiro.
- `fill`: opcional. Insira uma string, `Buffer`, `Uint8Array` ou um número inteiro. O padrão é `0`.
- `encoding`: opcional. Quando `fill` for uma string, insira uma das seguintes opções: `utf8`, `hex`, `base64` e `base64url`. O padrão é `utf8`.
- `Buffer.allocUnsafe(size)`

Aloque um `Buffer` não inicializado.

- `size`: insira um número inteiro.
- `Buffer.byteLength(value[, encoding])`

Exibe o tamanho de um valor, em bytes.

- `value`: uma string, `Buffer`, `TypedArray`, `DataView` ou `Arraybuffer`.

- `encoding`: opcional. Quando `value` for uma string, insira uma das seguintes opções: `utf8`, `hex`, `base64` e `base64url`. O padrão é `utf8`.
- `Buffer.compare(buffer1, buffer2)`

Compare dois `Buffers` para ajudar a classificar matrizes. Exibirá `0` se forem iguais, `-1` se `buffer1` vier primeiro ou `1` se `buffer2` vier primeiro.

- `buffer1`: insira um `Buffer`.
- `buffer2`: insira um `Buffer` diferente.
- `Buffer.concat(list[, totalLength])`

Concatene vários `Buffers`. Exibirá `0` se for nenhum. Exibe até `totalLength`.

- `list`: insira uma lista de `Buffers`. Observe que isso será truncado para `totalLength`.
- `totalLength`: opcional. Insira um número inteiro não assinado. Use a soma das instâncias de `Buffer` na lista se estiver em branco.
- `Buffer.from(array)`

Crie um `Buffer` por uma matriz.

- `array`: insira uma matriz de bytes de `0` a `255`.
- `Buffer.from(arrayBuffer, byteOffset[, length])`

Crie uma visualização de `arrayBuffer` começando pelo deslocamento `byteOffset` com o tamanho `length`.

- `arrayBuffer`: insira uma matriz `Buffer`.
- `byteOffset`: insira um número inteiro.
- `length`: opcional. Insira um número inteiro.
- `Buffer.from(buffer)`

Crie uma cópia do `Buffer`.

- `buffer`: insira um `Buffer`.
- `Buffer.from(object[, offsetOrEncoding[, length]])`

Crie um `Buffer` por meio de um objeto. Exibirá `Buffer.from(object.valueOf(), offsetOrEncoding, length)` se `valueOf()` não for igual ao objeto.

- `object`: insira um objeto.
- `offsetOrEncoding`: opcional. Insira um número inteiro ou uma string de codificação.

- `length`: opcional. Insira um número inteiro.
- `Buffer.from(string[, encoding])`

Crie um `Buffer` por meio de uma `string`.

- `string`: insira uma `string`.
- `encoding`: opcional. Insira uma das seguintes opções: `utf8`, `hex`, `base64` e `base64url`. O padrão é `utf8`.
- `Buffer.isBuffer(object)`

Confira se `object` é um `buffer`. Retorna `true` ou `false`.

- `object`: insira um objeto.
- `Buffer.isEncoding(encoding)`

Confira se `encoding` é compatível. Retorna `true` ou `false`.

- `encoding`: opcional. Insira uma das seguintes opções: `utf8`, `hex`, `base64` e `base64url`. O padrão é `utf8`.

O módulo oferece os seguintes métodos de protótipo de `buffer`:

- `Buffer.prototype.compare(target[, targetStart[, targetEnd[, sourceStart[, sourceEnd]]]])`

Compare `Buffer` com o destino. Exibirá `0` se forem iguais, `1` se `buffer` vier primeiro ou `-1` se `target` vier primeiro.

- `target`: insira um `Buffer`.
- `targetStart`: opcional. Insira um número inteiro. O padrão é `0`.
- `targetEnd`: opcional. Insira um número inteiro. O padrão é o tamanho do `target`.
- `sourceStart`: opcional. Insira um número inteiro. O padrão é `0`.
- `sourceEnd`: opcional. Insira um número inteiro. O padrão é o tamanho do `Buffer`.
- `Buffer.prototype.copy(target[, targetStart[, sourceStart[, sourceEnd]]])`

Copie o `buffer` em `target`.

- `target`: insira um `Buffer` ou uma `Uint8Array`.
- `targetStart`: opcional. Insira um número inteiro. O padrão é `0`.
- `sourceStart`: opcional. Insira um número inteiro. O padrão é `0`.

- `sourceEnd`: opcional. Insira um número inteiro. O padrão é o tamanho do `Buffer`.
- `Buffer.prototype.equals(otherBuffer)`

Compare `Buffer` com `otherBuffer`. Retorna `true` ou `false`.

- `otherBuffer`: insira uma `string`.
- `Buffer.prototype.fill(value[, offset[, end][, encoding])`

Preencha `Buffer` com `value`.

- `value`: insira uma `string`, um `Buffer` ou um número inteiro.
- `offset`: opcional. Insira um número inteiro.
- `end`: opcional. Insira um número inteiro.
- `encoding`: opcional. Insira uma das seguintes opções: `utf8`, `hex`, `base64` e `base64url`. O padrão é `utf8`.
- `Buffer.prototype.includes(value[, byteOffset][, encoding])`

Pesquise `value` no `Buffer`. Retorna `true` ou `false`.

- `value`: insira uma `string`, um `Buffer`, uma `Uint8Array` ou um número inteiro.
- `byteOffset`: opcional. Insira um número inteiro.
- `encoding`: opcional. Insira uma das seguintes opções: `utf8`, `hex`, `base64` e `base64url`. O padrão é `utf8`.
- `Buffer.prototype.indexOf(value[, byteOffset][, encoding])`

Procure o primeiro `value` em `Buffer`. Exibirá `index` se for encontrado; exibirá `-1` se não for encontrado.

- `value`: insira uma `string`, `Buffer`, `Unit8Array` ou um número inteiro de 0 a 255.
- `byteOffset`: opcional. Insira um número inteiro.
- `encoding`: opcional. Insira uma das seguintes opções se `value` for uma `string`: `utf8`, `hex`, `base64` e `base64url`. O padrão é `utf8`.
- `Buffer.prototype.lastIndexOf(value[, byteOffset][, encoding])`

Procure o último `value` em `Buffer`. Exibirá `index` se for encontrado; exibirá `-1` se não for encontrado.

- `value`: insira uma `string`, `Buffer`, `Unit8Array` ou um número inteiro de 0 a 255.
- `byteOffset`: opcional. Insira um número inteiro.

- `encoding`: opcional. Insira uma das seguintes opções se `value` for uma string: `utf8`, `hex`, `base64` e `base64url`. O padrão é `utf8`.
- `Buffer.prototype.readInt8(offset)`

Leia `Int8` em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `Buffer.prototype.readIntBE(offset, byteLength)`

Leia `Int` como `big-endian` em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `byteLength`: opcional. Insira um número inteiro de 1 a 6.
- `Buffer.prototype.readInt16BE(offset)`

Leia `Int16` como `big-endian` em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `Buffer.prototype.readInt32BE(offset)`

Leia `Int32` como `big-endian` em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `Buffer.prototype.readIntLE(offset, byteLength)`

Leia `Int` como `little-endian` em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.readInt16LE(offset)`

Leia `Int16` como `little-endian` em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `Buffer.prototype.readInt32LE(offset)`

Leia `Int32` como `little-endian` em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `Buffer.prototype.readUInt8(offset)`

Leia `UInt8` em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `Buffer.prototype.readUIntBE(offset, byteLength)`

Leia `UInt` como big-endian em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.readUInt16BE(offset)`

Leia `UInt16` como big-endian em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `Buffer.prototype.readUInt32BE(offset)`

Leia `UInt32` como big-endian em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `Buffer.prototype.readUIntLE(offset, byteLength)`

Leia `UInt` como little-endian em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.readUInt16LE(offset)`

Leia `UInt16` como little-endian em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `Buffer.prototype.readUInt32LE(offset)`

Leia `UInt32` como little-endian em `offset` de `Buffer`.

- `offset`: insira um número inteiro.
- `Buffer.prototype.readDoubleBE([offset])`

Leia um double de 64 bits como big-endian em `offset` de `Buffer`.

- `offset`: opcional. Insira um número inteiro.
- `Buffer.prototype.readDoubleLE([offset])`

Leia um double de 64 bits como little-endian em `offset` de `Buffer`.

- `offset`: opcional. Insira um número inteiro.

- `Buffer.prototype.readFloatBE([offset])`

Leia um float de 32 bits como big-endian em `offset` de `Buffer`.

- `offset`: opcional. Insira um número inteiro.

- `Buffer.prototype.readFloatLE([offset])`

Leia um float de 32 bits como little-endian em `offset` de `Buffer`.

- `offset`: opcional. Insira um número inteiro.

- `Buffer.prototype.subarray([start[, end]])`

Exibe uma cópia do `Buffer` que está deslocado e recortado com novos `start` e `end`.

- `start`: opcional. Insira um número inteiro. O padrão é 0.
- `end`: opcional. Insira um número inteiro. O padrão é o tamanho do buffer.

- `Buffer.prototype.swap16()`

Troque a ordem de bytes da matriz de `Buffer` tratando-a como uma matriz de números de 16 bits. O tamanho do `Buffer` deve ser divisível por 2, ou você receberá um erro.

- `Buffer.prototype.swap32()`

Troque a ordem de bytes da matriz de `Buffer` tratando-a como uma matriz de números de 32 bits. O tamanho do `Buffer` deve ser divisível por 4, ou você receberá um erro.

- `Buffer.prototype.swap64()`

Troque a ordem de bytes da matriz de `Buffer` tratando-a como uma matriz de números de 64 bits. O tamanho do `Buffer` deve ser divisível por 8, ou você receberá um erro.

- `Buffer.prototype.toJSON()`

Exibe `Buffer` como JSON.

- `Buffer.prototype.toString([encoding[, start[, end]])`

Converta `Buffer`, de `start` em `end` em string codificada.

- `encoding`: opcional. Insira uma das seguintes opções: `utf8`, `hex`, `base64` ou `base64url`. O padrão é `utf8`.
- `start`: opcional. Insira um número inteiro. O padrão é 0.
- `end`: opcional. Insira um número inteiro. O padrão é o tamanho do buffer.

- `Buffer.prototype.write(string[, offset[, length]][, encoding])`

Escreva `string` codificado em `Buffer` se houver espaço ou um `string` truncado se não houver espaço suficiente.

- `string`: insira uma `string`.
- `offset`: opcional. Insira um número inteiro. O padrão é 0.
- `length`: opcional. Insira um número inteiro. O padrão é o tamanho da `string`.
- `encoding`: opcional. Opcionalmente, insira uma das seguintes opções: `utf8`, `hex`, `base64` ou `base64url`. O padrão é `utf8`.
- `Buffer.prototype.writeInt8(value, offset, byteLength)`

Escreva `Int8` `value` de `byteLength` em `offset` para `Buffer`.

- `value`: insira um número inteiro.
- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeIntBE(value, offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando `big-endian`.

- `value`: insira um número inteiro.
- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeInt16BE(value, offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando `big-endian`.

- `value`: insira um número inteiro.
- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeInt32BE(value, offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando `big-endian`.

- `value`: insira um número inteiro.
- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeIntLE(offset, byteLength)`

Escrever código de função

Escreva `value` em `offset` para `Buffer`, usando little-endian.

- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeInt16LE(offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando little-endian.

- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeInt32LE(offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando little-endian.

- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeUInt8(value, offset, byteLength)`

Escreva `UInt8 value` de `byteLength` em `offset` para `Buffer`.

- `value`: insira um número inteiro.
- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeUIntBE(value, offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando big-endian.

- `value`: insira um número inteiro.
- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeUInt16BE(value, offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando big-endian.

- `value`: insira um número inteiro.
- `offset`: insira um número inteiro.
- `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeUInt32BE(value, offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando `big-endian`.

- `value`: insira um número inteiro.
 - `offset`: insira um número inteiro.
 - `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeUIntLE(value, offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando `little-endian`.

- `value`: insira um número inteiro.
 - `offset`: insira um número inteiro.
 - `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeUInt16LE(value, offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando `little-endian`.

- `value`: insira um número inteiro.
 - `offset`: insira um número inteiro.
 - `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeUInt32LE(value, offset, byteLength)`

Escreva `value` em `offset` para `Buffer`, usando `little-endian`.

- `value`: insira um número inteiro.
 - `offset`: insira um número inteiro.
 - `byteLength`: insira um número inteiro de 1 a 6.
- `Buffer.prototype.writeDoubleBE(value, [offset])`

Escreva `value` em `offset` para `Buffer`, usando `big-endian`.

- `value`: insira um número inteiro.
 - `offset`: opcional. Insira um número inteiro. O padrão é 0.
- `Buffer.prototype.writeDoubleLE(value, [offset])`

Escreva `value` em `offset` para `Buffer`, usando `little-endian`.

- `value`: insira um número inteiro.
 - `offset`: opcional. Insira um número inteiro. O padrão é 0.
- `Buffer.prototype.writeFloatBE(value, [offset])`

Escreva `value` em `offset` para `Buffer`, usando `big-endian`.

- `value`: insira um número inteiro.
- `offset`: opcional. Insira um número inteiro. O padrão é 0.
- `Buffer.prototype.writeFloatLE(value, [offset])`

Escreva `value` em `offset` para `Buffer`, usando `little-endian`.

- `value`: insira um número inteiro.
- `offset`: opcional. Insira um número inteiro. O padrão é 0.

Os seguintes métodos de instância são compatíveis:

- `buffer[index]`

Obtenha e defina o octeto (byte) em `index` em `Buffer`.

- Obtenha um número de 0 a 255. Ou defina um número de 0 a 255.

As seguintes propriedades de instância são compatíveis:

- `buffer`

Obtenha o objeto `ArrayBuffer` para o `buffer`.

- `byteOffset`

Obtenha o `byteOffset` do objeto `Arraybuffer` do `buffer`.

- `length`

Obtenha a contagem de bytes do `buffer`.

Note

Todos os métodos do módulo de `buffer` são novos no runtime 2.0 do JavaScript.

String de consulta

Note

O [objeto de evento CloudFront Functions](#) analisa automaticamente as cadeias de consulta de URL para você. Isso significa que na maioria dos casos você não precisa usar este módulo.

O módulo cadeia de consulta (`querystring`) fornece métodos para analisar e formatar cadeias de consulta de URL. Você pode carregar o módulo usando `require('querystring')`. O módulo fornece os seguintes métodos:

`querystring.escape(string)`

O URL codifica a `string` fornecida, retornando uma cadeia de consulta escapada. O método é usado por `querystring.stringify()` e não deve ser usado diretamente.

`querystring.parse(string[, separator[, equal[, options]])`

Analisa uma cadeia de consulta (`string`) e retorna um objeto.

O parâmetro `separator` é uma substring para delimitar pares de chaves e valores na cadeia de consulta. O padrão é `&`.

O parâmetro `equal` é uma substring para delimitar chaves e valores na cadeia de consulta. O padrão é `=`.

O parâmetro `options` é um objeto com as seguintes chaves:

`decodeURIComponent` *function*

Uma função para decodificar caracteres codificados por percentual na cadeia de consulta. O padrão é `querystring.unescape()`.

`maxKeys` *number*

O número máximo de chaves a serem analisadas. O padrão é `1000`. Use um valor de `0` para remover as limitações para as chaves de contagem.

Por padrão, os caracteres codificados por percentual dentro da cadeia de consulta são assumidos para usar a codificação UTF-8. Sequências UTF-8 inválidas são substituídas pelo caractere de substituição U+FFFD.

Por exemplo, para a seguinte cadeia de consulta:

```
'name=value&abc=xyz&abc=123'
```

O valor de retorno de `queryString.parse()` é:

```
{
  name: 'value',
  abc: ['xyz', '123']
}
```

`queryString.decode()` é um alias para `queryString.parse()`.

`queryString.stringify(object[, separator[, equal[, options]])`

Serializa um `object` e retorna uma cadeia de consulta.

O parâmetro `separator` é uma substring para delimitar pares de chaves e valores na cadeia de consulta. O padrão é `&`.

O parâmetro `equal` é uma substring para delimitar chaves e valores na cadeia de consulta. O padrão é `=`.

O parâmetro `options` é um objeto com as seguintes chaves:

`encodeURIComponent` *function*

A função a ser usada para converter caracteres URL-inseguros para codificação percentual na cadeia de consulta. O padrão é `queryString.escape()`.

Por padrão, os caracteres que exigem percentual de codificação dentro da cadeia de consulta são codificados como UTF-8. Para usar uma codificação diferente, especifique a opção `encodeURIComponent`.

Por exemplo, para o seguinte código:

```
queryString.stringify({ name: 'value', abc: ['xyz', '123'], anotherName: '' });
```

O valor de retorno é:

```
'name=value&abc=xyz&abc=123&anotherName='
```

`querystring.encode()` é um alias para `querystring.stringify()`.

`querystring.unescape(string)`

Decodifica caracteres codificados por porcentagem de URL na `string` fornecida, retornando uma cadeia de consulta sem escapamento. Esse método é usado por `querystring.parse()` e não deve ser usado diretamente.

Criptografia

O módulo criptográfico (`crypto`) fornece auxiliares de código de autenticação de mensagens (HMAC) padrão e hash. Você pode carregar o módulo usando `require('crypto')`.

Métodos de hash

`crypto.createHash(algorithm)`

Cria e retorna um objeto hash que você pode usar para gerar resumos de hash usando o algoritmo fornecido: `md5`, `sha1` ou `sha256`.

`hash.update(data)`

Atualiza o conteúdo de hash com os `data` fornecidos.

`hash.digest([encoding])`

Calcula o resumo de todos os dados passados usando `hash.update()`. A codificação pode ser `hex`, `base64` ou `base64url`.

Métodos de HMAC

`crypto.createHmac(algorithm, secret key)`

Cria e retorna um objeto HMAC que usa o `algorithm` e a `secret key` fornecidos. O algoritmo pode ser `md5`, `sha1` ou `sha256`.

`hmac.update(data)`

Atualiza o conteúdo HMAC com os `data` fornecidos.

`hmac.digest([encoding])`

Calcula o resumo de todos os dados passados usando `hmac.update()`. A codificação pode ser `hex`, `base64` ou `base64url`.

Recursos restritos

Os seguintes recursos de linguagem JavaScript não são compatíveis ou são restritos devido a preocupações de segurança.

Avaliação dinâmica do código

A avaliação dinâmica do código não é compatível. Ambos os construtores `eval()` e `Function` lançam um erro se forem tentados. Por exemplo, `const sum = new Function('a', 'b', 'return a + b')` lança um erro.

Temporizadores

As funções `setTimeout()`, `setImmediate()` e `clearTimeout()` não são compatíveis. Não há provisão para adiar ou renderizar dentro de uma execução de função. Sua função deve ser executada de forma síncrona até a conclusão.

Data e carimbos de data/hora

Por razões de segurança, não há acesso a temporizadores de alta resolução. Todos os métodos `Date` para consultar a hora atual sempre retornam o mesmo valor durante o tempo de vida de uma única execução de função. O carimbo de data/hora retornado é o momento em que a função começou a ser executada. Consequentemente, você não pode medir o tempo decorrido em sua função.

Acesso ao sistema de arquivos

Não há acesso ao sistema de arquivos. Por exemplo, não há módulo `fs` para acesso ao sistema de arquivos como no `Node.js`.

Acesso a processos

Não há acesso a processos. Por exemplo, não há um objeto global `process` para processar o acesso às informações como no `Node.js`.

Variáveis de ambiente

Não há acesso às variáveis de ambiente. Em vez disso, é possível usar o `KeyValueStore` do CloudFront para criar um datastore centralizado de pares de chave-valor para o CloudFront Functions. O `KeyValueStore` do CloudFront permite atualizações dinâmicas nos dados de configuração sem a necessidade de implantar alterações no código. Para ter mais informações, consulte [KeyValueStore do Amazon CloudFront](#).

Acesso à rede

Não há suporte para chamadas de rede. Por exemplo, XHR, HTTP (S) e socket não são compatíveis.

Métodos auxiliares para armazenamentos de chave-valor

Esta seção se aplicará se você usar o [Armazenamento de chave-valor do CloudFront](#) para incluir chave-valor na função criada. O CloudFront Functions tem um módulo que oferece três métodos auxiliares para ler valores do armazenamento de chave-valor.

Para usar esse módulo no código da função, verifique se você [associou um armazenamento de chave-valor](#) à função.

Em seguida, inclua as seguintes declarações nas primeiras linhas do código da função:

```
import cf from 'cloudfront';
const kvsId = "key value store ID";
const kvsHandle = cf.kvs(kvsId);
```

Seu *ID de armazenamento de chave-valor* pode apresentar a seguinte aparência:
a1b2c3d4-5678-90ab-cdef-EXAMPLE1

Método do **get()**

Use esse método para retornar o valor do nome da chave especificado.

Solicitação

```
get("key", options);
```

- **key**: o nome da chave cujo valor precisa ser buscado.
- **options**: existe uma opção, **format**. Isso garante que a função analise os dados corretamente.

Possíveis valores:

- **string**: (padrão) codificado em UTF8.
- **json**
- **bytes**: buffer de dados binários brutos.

Exemplo de solicitação

```
const value = await kvsHandle.get("myFunctionKey", { format: "string"});
```

Resposta

A resposta é uma `promise` que se resolve para um valor no formato solicitado usando `options`. Por padrão, o valor é retornado como uma `string`.

Tratamento de erros

O método `get()` retornará um erro quando a chave solicitada não existir no armazenamento de chave-valor associado. Para gerenciar esse caso de uso, é possível adicionar um bloco `try` e `catch` ao seu código.

Método do `exists()`

Use esse método para identificar se a chave existe ou não no armazenamento de chave-valor.

Solicitação

```
exists("key");
```

Exemplo de solicitação

```
const exist = await kvsHandle.exists("myFunctionkey");
```

Resposta

A resposta é uma `promise` que retorna um booleano (`true` ou `false`). Esse valor especifica se a chave existe ou não no armazenamento de chave-valor.

Método do `meta()`

Use esse método para retornar metadados sobre o armazenamento de chave-valor.

Solicitação

```
meta();
```

Exemplo de solicitação

```
const meta = await kvsHandle.meta();
```

Resposta

A resposta é uma *promise* que é resolvida em um objeto com as seguintes propriedades:

- `creationDateTime`: a data e a hora no formato ISO 8601 em que o armazenamento de chave-valor foi criado.
- `lastUpdatedDateTime`: a data e a hora em que o valor da chave armazenado foi sincronizado pela última vez a partir da origem, no formato ISO 8601. O valor não inclui o tempo de propagação até a borda.
- `keyCount`: o número total de chaves no KVS após a última sincronização da origem.

Exemplo de resposta

```
{keyCount:3,creationDateTime:2023-11-30T23:07:55.765Z,lastUpdatedDateTime:2023-12-15T03:57:52.4
```

Código de exemplo para CloudFront Functions

Esses exemplos abrangem diferentes casos de uso, desde armazenamento em cache e segurança até manipulação de URL e validação de tokens. É possível usar o CloudFront Functions para aprimorar a entrega de conteúdo e personalizar o comportamento das distribuições do CloudFront.

Cada função de exemplo inclui uma breve descrição e um link para o código correspondente no GitHub. Também é possível encontrar esses exemplos no [repositório amazon-cloudfront-functions](#) no GitHub.

Para começar a escrever código de função para o CloudFront Functions, consulte os seguintes exemplos:

Tópicos

- [Adicionar um cabeçalho Cache-Control à resposta](#)
- [Adicionar um cabeçalho CORS \(compartilhamento de recursos de origem cruzada\) à resposta](#)
- [Adicionar cabeçalho CORS \(compartilhamento de recursos de origem cruzada\) à solicitação](#)
- [Adicionar cabeçalhos de segurança à resposta](#)
- [Adicionar um cabeçalho True-Client-IP à solicitação](#)
- [Redirecionar o visualizador para uma nova URL](#)

- [Adicionar index.html aos URLs de solicitação sem um nome de arquivo](#)
- [Validar um token simples na solicitação](#)
- [Utilizar async e await](#)
- [Normalizar parâmetros de string de consulta](#)
- [Usar pares de chave-valor em uma função](#)

Adicionar um cabeçalho Cache-Control à resposta

A função de resposta do visualizador a seguir adiciona um cabeçalho HTTP Cache-Control à resposta. O cabeçalho usa a diretiva max-age para dizer aos navegadores da Web para armazenar em cache a resposta por um máximo de dois anos (63.072.000 segundos). Para obter mais informações, consulte [Cache-Control](#) no site do MDN Web Docs.

[Veja este exemplo no GitHub.](#)

JavaScript runtime 2.0

```
async function handler(event) {
  const response = event.response;
  const headers = response.headers;

  if (response.statusCode >= 200 && response.statusCode < 400) {
    // Set the cache-control header
    headers['cache-control'] = {value: 'public, max-age=63072000'};
  }

  // Return response to viewers
  return response;
}
```

JavaScript runtime 1.0

```
function handler(event) {
  var response = event.response;
  var headers = response.headers;

  if (response.statusCode >= 200 && response.statusCode < 400) {
    // Set the cache-control header
    headers['cache-control'] = {value: 'public, max-age=63072000'};
  }
}
```

```
// Return response to viewers
return response;
}
```

Adicionar um cabeçalho CORS (compartilhamento de recursos de origem cruzada) à resposta

A função de resposta do visualizador a seguir adicionará um cabeçalho HTTP `Access-Control-Allow-Origin` à resposta se ela ainda não contiver esse cabeçalho. Esse cabeçalho faz parte do [compartilhamento de recursos de origem cruzada \(CORS\)](#). O valor do cabeçalho (*) diz aos navegadores da Web para permitir que o código de qualquer origem acesse esse recurso. Para obter mais informações, consulte [Access-Control-Allow-Origin](#) no site do MDN Web Docs.

[Veja este exemplo no GitHub.](#)

JavaScript runtime 2.0

```
async function handler(event) {
  const request = event.request;
  const response = event.response;

  // If Access-Control-Allow-Origin CORS header is missing, add it.
  // Since JavaScript doesn't allow for hyphens in variable names, we use the
  dict["key"] notation.
  if (!response.headers['access-control-allow-origin'] &&
  request.headers['origin']) {
    response.headers['access-control-allow-origin'] = {value:
  request.headers['origin'].value};
    console.log("Access-Control-Allow-Origin was missing, adding it now.");
  }

  return response;
}
```

JavaScript runtime 1.0

```
function handler(event) {
  var response = event.response;
  var headers = response.headers;

  // If Access-Control-Allow-Origin CORS header is missing, add it.
```



```
// Since JavaScript doesn't allow for hyphens in variable names, we use the
dict["key"] notation.
if (!headers['access-control-allow-origin']) {
  headers['access-control-allow-origin'] = {value: "*"};
  console.log("Access-Control-Allow-Origin was missing, adding it now.");
}

return response;
}
```

Adicionar cabeçalho CORS (compartilhamento de recursos de origem cruzada) à solicitação

A função de solicitação do visualizador a seguir adicionará um cabeçalho HTTP Origin à solicitação se ela ainda não contiver esse cabeçalho. Esse cabeçalho faz parte do [compartilhamento de recursos de origem cruzada \(CORS\)](#). Este exemplo define o valor do cabeçalho para o valor no cabeçalho Host da solicitação. Para obter mais informações, consulte [Origin](#) no site do MDN Web Docs.

[Veja este exemplo no GitHub.](#)

JavaScript runtime 2.0

```
async function handler(event) {
  const request = event.request;
  const headers = request.headers;
  const host = request.headers.host.value;

  // If origin header is missing, set it equal to the host header.
  if (!headers.origin)
    headers.origin = {value: `https://${host}`};

  return request;
}
```

JavaScript runtime 1.0

```
function handler(event) {
  var request = event.request;
  var headers = request.headers;
  var host = request.headers.host.value;

  // If origin header is missing, set it equal to the host header.
```

```
if (!headers.origin)
  headers.origin = {value: `https://${host}`};

return request;
}
```

Adicionar cabeçalhos de segurança à resposta

A função de resposta do visualizador a seguir adiciona à resposta vários cabeçalhos HTTP comuns relacionados à segurança. Para obter mais informações, consulte as seguintes páginas no site do MDN Web Docs:

- [Strict-Transport-Security](#)
- [Content-Security-Policy](#)
- [X-Content-Type-Options](#)
- [X-Frame-Options](#)
- [X-XSS-Protection](#)

[Veja este exemplo no GitHub.](#)

JavaScript runtime 2.0

```
async function handler(event) {
  const response = event.response;
  const headers = response.headers;

  // Set HTTP security headers
  // Since JavaScript doesn't allow for hyphens in variable names, we use the
  dict["key"] notation
  headers['strict-transport-security'] = { value: 'max-age=63072000;
includeSubdomains; preload'};
  headers['content-security-policy'] = { value: "default-src 'none'; img-src
'self'; script-src 'self'; style-src 'self'; object-src 'none'; frame-ancestors
'none'"};
  headers['x-content-type-options'] = { value: 'nosniff'};
  headers['x-frame-options'] = {value: 'DENY'};
  headers['x-xss-protection'] = {value: '1; mode=block'};
  headers['referrer-policy'] = {value: 'same-origin'};

  // Return the response to viewers
```

```
    return response;
}
```

JavaScript runtime 1.0

```
function handler(event) {
    var response = event.response;
    var headers = response.headers;

    // Set HTTP security headers
    // Since JavaScript doesn't allow for hyphens in variable names, we use the
    dict["key"] notation
    headers['strict-transport-security'] = { value: 'max-age=63072000;
includeSubdomains; preload'};
    headers['content-security-policy'] = { value: "default-src 'none'; img-src
'self'; script-src 'self'; style-src 'self'; object-src 'none'"};
    headers['x-content-type-options'] = { value: 'nosniff'};
    headers['x-frame-options'] = {value: 'DENY'};
    headers['x-xss-protection'] = {value: '1; mode=block'};

    // Return the response to viewers
    return response;
}
```

Adicionar um cabeçalho True-Client-IP à solicitação

A função de solicitação do visualizador a seguir adiciona um cabeçalho HTTP True-Client-IP à solicitação, com o endereço IP do visualizador como o valor do cabeçalho. Quando o CloudFront envia uma solicitação para uma origem, a origem pode determinar o endereço IP do host do CloudFront que enviou a solicitação, mas não o endereço IP do visualizador (cliente) que enviou a solicitação original para o CloudFront. Esta função adiciona o cabeçalho True-Client-IP para que a origem possa ver o endereço IP do visualizador.

Important

Para garantir que o CloudFront inclua esse cabeçalho nas solicitações de origem, você deve adicioná-lo à lista de cabeçalhos permitidos em uma [política de solicitação de origem](#).

[Veja este exemplo no GitHub.](#)

JavaScript runtime 2.0

```
async function handler(event) {
  var request = event.request;
  var clientIP = event.viewer.ip;

  //Add the true-client-ip header to the incoming request
  request.headers['true-client-ip'] = {value: clientIP};

  return request;
}
```

JavaScript runtime 1.0

```
function handler(event) {
  var request = event.request;
  var clientIP = event.viewer.ip;

  //Add the true-client-ip header to the incoming request
  request.headers['true-client-ip'] = {value: clientIP};

  return request;
}
```

Redirecionar o visualizador para uma nova URL

A função de solicitação do visualizador a seguir gera uma resposta para redirecionar o visualizador para um URL específico do país quando a solicitação vem de um país específico. Esta função depende do valor do cabeçalho `CloudFront-Viewer-Country` para determinar o país do visualizador.

Important

Para que essa função funcione, você deve configurar o CloudFront para adicionar o cabeçalho `CloudFront-Viewer-Country` às solicitações recebidas adicionando-o aos cabeçalhos permitidos em uma [política de cache](#) ou a uma [política de solicitação de origem](#).

Este exemplo redireciona o visualizador para uma URL específica da Alemanha quando a solicitação do visualizador vem da Alemanha. Se a solicitação do visualizador não for proveniente da Alemanha, a função retornará a solicitação original e não modificada.

[Veja este exemplo no GitHub.](#)

JavaScript runtime 2.0

```
async function handler(event) {
  const request = event.request;
  const headers = request.headers;
  const host = request.headers.host.value;
  const country = Symbol.for('DE'); // Choose a country code
  const newurl = `https://${host}/de/index.html`; // Change the redirect URL to
  your choice

  if (headers['cloudfront-viewer-country']) {
    const countryCode = Symbol.for(headers['cloudfront-viewer-country'].value);
    if (countryCode === country) {
      const response = {
        statusCode: 302,
        statusDescription: 'Found',
        headers:
          { "location": { "value": newurl } }
      }

      return response;
    }
  }
  return request;
}
```

JavaScript runtime 1.0

```
function handler(event) {
  var request = event.request;
  var headers = request.headers;
  var host = request.headers.host.value;
  var country = 'DE' // Choose a country code
  var newurl = `https://${host}/de/index.html` // Change the redirect URL to your
  choice

  if (headers['cloudfront-viewer-country']) {
```

```
var countryCode = headers['cloudfront-viewer-country'].value;
if (countryCode === country) {
  var response = {
    statusCode: 302,
    statusDescription: 'Found',
    headers:
      { "location": { "value": newurl } }
  }

  return response;
}
return request;
}
```

Para ter mais informações sobre regravações e redirecionamentos, consulte [Como lidar com regravações e redirecionamentos usando funções de borda](#) no Workshop Studio da AWS.

Adicionar index.html aos URLs de solicitação sem um nome de arquivo

A função de solicitação do visualizador a seguir anexa `index.html` a solicitações que não incluem um nome de arquivo nem extensão no URL. Essa função pode ser útil para aplicações de página única ou sites gerados estaticamente hospedados em um bucket do Amazon S3.

[Veja este exemplo no GitHub.](#)

JavaScript runtime 2.0

```
async function handler(event) {
  const request = event.request;
  const uri = request.uri;

  // Check whether the URI is missing a file name.
  if (uri.endsWith('/')) {
    request.uri += 'index.html';
  }
  // Check whether the URI is missing a file extension.
  else if (!uri.includes('.')) {
    request.uri += '/index.html';
  }

  return request;
}
```

```
}
```

JavaScript runtime 1.0

```
function handler(event) {
  var request = event.request;
  var uri = request.uri;

  // Check whether the URI is missing a file name.
  if (uri.endsWith('/')) {
    request.uri += 'index.html';
  }
  // Check whether the URI is missing a file extension.
  else if (!uri.includes('.')) {
    request.uri += '/index.html';
  }

  return request;
}
```

Validar um token simples na solicitação

A função de solicitação do visualizador a seguir valida um [Token Web JSON \(JWT\)](#) na string de consulta de uma solicitação. Se o token for válido, a função retornará a solicitação original e não modificada para o CloudFront. Se o token não for válido, a função vai gerar uma resposta de erro. Esta função usa o módulo `crypto`. Para ter mais informações, consulte [Módulos integrados](#).

Esta função pressupõe que as solicitações contenham um valor JWT em um parâmetro de cadeia de consulta chamado `jwt`. O ID `KeyValueStore` do CloudFront é opcional no código a seguir.

[Veja este exemplo no GitHub.](#)

JavaScript runtime 2.0

```
import crypto from 'crypto';
import cf from 'cloudfront';

//Response when JWT is not valid.
const response401 = {
  statusCode: 401,
  statusDescription: 'Unauthorized'
};
```

```
// Remember to associate the KVS with your function before calling the const kvsKey
= 'jwt.secret'.
// https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/kvs-with-
functions-associate.html
const kvsKey = 'jwt.secret';
// set to true to enable console logging
const loggingEnabled = false;

function jwt_decode(token, key, noVerify, algorithm) {
  // check token
  if (!token) {
    throw new Error('No token supplied');
  }
  // check segments
  const segments = token.split('.');
  if (segments.length !== 3) {
    throw new Error('Not enough or too many segments');
  }

  // All segment should be base64
  const headerSeg = segments[0];
  const payloadSeg = segments[1];
  const signatureSeg = segments[2];

  // base64 decode and parse JSON
  const payload = JSON.parse(_base64urlDecode(payloadSeg));

  if (!noVerify) {
    const signingMethod = 'sha256';
    const signingType = 'hmac';

    // Verify signature. `sign` will return base64 string.
    const signingInput = [headerSeg, payloadSeg].join('.');

    if (!_verify(signingInput, key, signingMethod, signingType, signatureSeg)) {
      throw new Error('Signature verification failed');
    }

    // Support for nbf and exp claims.
    // According to the RFC, they should be in seconds.
    if (payload.nbf && Date.now() < payload.nbf*1000) {
      throw new Error('Token not yet active');
    }
  }
}
```



```
    }

    if (payload.exp && Date.now() > payload.exp*1000) {
      throw new Error('Token expired');
    }
  }

  return payload;
}

//Function to ensure a constant time comparison to prevent
//timing side channels.
function _constantTimeEquals(a, b) {
  if (a.length !== b.length) {
    return false;
  }

  let xor = 0;
  for (let i = 0; i < a.length; i++) {
    xor |= (a.charCodeAt(i) ^ b.charCodeAt(i));
  }

  return 0 === xor;
}

function _verify(input, key, method, type, signature) {
  if(type === "hmac") {
    return _constantTimeEquals(signature, _sign(input, key, method));
  }
  else {
    throw new Error('Algorithm type not recognized');
  }
}

function _sign(input, key, method) {
  return crypto.createHmac(method, key).update(input).digest('base64url');
}

function _base64urlDecode(str) {
  return Buffer.from(str, 'base64url')
}

async function handler(event) {
  let request = event.request;
```

```
//Secret key used to verify JWT token.
//Update with your own key.
const secret_key = await getSecret()

if(!secret_key) {
  return response401;
}

// If no JWT token, then generate HTTP redirect 401 response.
if(!request.querystring.jwt) {
  log("Error: No JWT in the querystring");
  return response401;
}

const jwtToken = request.querystring.jwt.value;

try{
  jwt_decode(jwtToken, secret_key);
}
catch(e) {
  log(e);
  return response401;
}

//Remove the JWT from the query string if valid and return.
delete request.querystring.jwt;
log("Valid JWT token");
return request;
}

// get secret from key value store
async function getSecret() {
  // initialize cloudfront kv store and get the key value
  try {
    const kvsHandle = cf.kvs();
    return await kvsHandle.get(kvsKey);
  } catch (err) {
    log(`Error reading value for key: ${kvsKey}, error: ${err}`);
    return null;
  }
}
}
```

```
function log(message) {
  if (loggingEnabled) {
    console.log(message);
  }
}
```

Utilizar async e await

As funções do JavaScript runtime 2.0 do CloudFront Functions oferecem as sintaxes `async` e `await` para lidar com objetos `Promise`. Promises representam resultados atrasados que podem ser acessados por meio da palavra-chave `await` nas funções marcadas como `async`. Várias novas funções do WebCrypto usam Promises.

Para obter mais informações sobre objetos `Promise`, consulte [Promise](#).

Note

É necessário usar o JavaScript runtime 2.0 para os exemplos de código a seguir.

```
async function answer() {
  return 42;
}

// Note: async, await can be used only inside an async function.

async function handler(event) {
  // var answer_value = answer(); // returns Promise, not a 42 value
  let answer_value = await answer(); // resolves Promise, 42
  console.log("Answer"+answer_value);
  event.request.headers['answer'] = { value : ""+answer_value };
  return event.request;
}
```

O exemplo de código JavaScript a seguir mostra como visualizar Promises com o método de cadeia `then`. É possível usar `catch` para visualizar erros.

```
async function answer() {
  return 42;
}
```

```
async function squared_answer() {
  return answer().then(value => value * value)
}
// note async, await can be used only inside async function
async function handler(event) {
  // var answer_value = answer(); // returns Promise, not a 42 value
  let answer_value = await squared_answer(); // resolves Promise, 42
  console.log("Answer"+answer_value);
  event.request.headers['answer'] = { value : ""+answer_value };
  return event.request;
}
```

Normalizar parâmetros de string de consulta

É possível normalizar os parâmetros da string de consulta para melhorar a taxa de acertos de cache.

O exemplo a seguir funciona com JavaScript runtime 1.0 e 2.0. O exemplo mostra como melhorar a taxa de acertos de cache colocando as strings de consulta em ordem alfabética antes que o CloudFront encaminhe as solicitações para a origem.

```
function handler(event) {
  var qs=[];
  for (var key in event.request.querystring) {
    if (event.request.querystring[key].multiValue) {
      event.request.querystring[key].multiValue.forEach((mv) => {qs.push(key +
"=" + mv.value)});
    } else {
      qs.push(key + "=" + event.request.querystring[key].value);
    }
  }
};

event.request.querystring = qs.sort().join('&');

return event.request;
}
```

Usar pares de chave-valor em uma função

É possível usar pares de chave-valor de um [armazenamento de chave-valor](#) em uma função.

O exemplo mostra uma função que usa o conteúdo do URL na solicitação HTTP para pesquisar um caminho personalizado no armazenamento de chave-valor. Depois, o CloudFront usa esse caminho personalizado para fazer a solicitação. Essa função ajuda a gerenciar os vários caminhos que fazem parte de um site.

Por exemplo, se o blog anterior tiver um caminho de origem `/blog-v1` e o novo blog tiver um caminho de origem `/blog-v2`, essa função poderá pesquisar o caminho do URL da solicitação recebida e reescrever o caminho do URL (`/blog-v1`) para a nova versão do blog (`/blog-v2`).

Note

É necessário usar o JavaScript runtime 2.0 para o exemplo de código a seguir.

```
import cf from 'cloudfront';

// This fails if there is no key value store associated with the function
const kvsHandle = cf.kvs();

// Remember to associate the KVS with your function before referencing KVS in your
// code.
// https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/kvs-with-
// functions-associate.html
async function handler(event) {
  const request = event.request;
  // Use the first segment of the pathname as key
  // For example http(s)://domain/key/something/else
  const pathSegments = request.uri.split('/')
  const key = pathSegments[1]
  try {
    // Replace the first path of the pathname with the value of the key
    // For example http(s)://domain/value/something/else
    pathSegments[1] = await kvsHandle.get(key);
    const newUri = pathSegments.join('/');
    console.log(`${request.uri} -> ${newUri}`)
    request.uri = newUri;
  } catch (err) {
    // No change to the pathname if the key is not found
    console.log(`${request.uri} | ${err}`);
  }
  return request;
}
```

```
}
```

Criar funções

Você deve criar uma função em dois estágios:

1. Crie o código da função como JavaScript. É possível usar o exemplo padrão do console do CloudFront ou crie seu próprio. Para obter mais informações, consulte os tópicos a seguir.
 - [Escrever código de função](#)
 - [the section called “Estrutura de eventos”](#)
 - [Código de exemplo para CloudFront Functions](#)
2. Use o CloudFront para criar a função e incluir o código. O código existe dentro da função (não como referência).

Console

Como criar uma função do

1. Faça login no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions> e selecione a página Funções.
2. Escolha a opção Criar função.
3. Insira um nome de função exclusivo na Conta da AWS, selecione a versão do JavaScript e escolha Continuar. A página de detalhes da nova função é exibida.

Note

Para usar [pares de chave-valor](#) na função, será necessário selecionar JavaScript runtime 2.0.

4. Na seção Código da função, selecione a guia Criar e insira o código da função. O exemplo de código incluído na guia Criar ilustra a sintaxe básica do código da função.
5. Escolha Salvar alterações.
6. Se o código da função usar pares de chave-valor, será necessário associar um armazenamento de valores de chave.

É possível associar o armazenamento de chave-valor durante a criação inicial da função. Ou é possível associá-lo posteriormente, [atualizando a função](#).

Para associar um armazenamento de chave-valor agora, siga estas etapas:

- Acesse a seção **Associar KeyValueStore** e selecione **Associar KeyValueStore** existente.
- Selecione o armazenamento de chave-valor que contém os pares de chave-valor na função e escolha **Associar KeyValueStore**.

O CloudFront associa imediatamente o armazenamento à função. Não é necessário salvar a função.

CLI

Se você usa a CLI, normalmente primeiro cria o código da função em um arquivo e, depois, cria a função com a AWS CLI.

Como criar uma função do

1. Crie o código da função em um arquivo e armazene-o em um diretório ao qual o computador possa se conectar.
2. Execute o comando conforme mostrado no exemplo. Este exemplo usa a notação `fileb://` para transmitir o arquivo. Ele também inclui quebras de linha para tornar o comando mais legível.

```
aws cloudfront create-function \  
  --name MaxAge \  
  --function-config '{"Comment":"Max Age 2 years","Runtime":"cloudfront-  
js-2.0","KeyValueStoreAssociations":{"Quantity":1,"Items":  
[{"KeyValueStoreARN":"arn:aws:cloudfront::111122223333:key-value-store/  
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"}]}' \  
  --function-code fileb://function-max-age-v1.js
```

Observações

- **Runtime**: a versão do JavaScript. Para usar [pares de chave-valor](#) na função, é necessário especificar a versão 2.0.
- **KeyValueStoreAssociations**: se a função usar pares de chave-valor, será possível associar o armazenamento de chave-valor durante a criação inicial da função. Ou é possível associá-lo posteriormente, usando `update-function`. A

Quantity é sempre 1 porque cada função pode ter apenas um armazenamento de chave-valor associado a ela.

Quando o comando é bem-sucedido, a saída é semelhante à seguinte:

```
ETag: ETVABCEXAMPLE
FunctionSummary:
  FunctionConfig:
    Comment: Max Age 2 years
    Runtime: cloudfront-js-2.0
    KeyValueStoreAssociations= \
      {Quantity=1, \
        Items=[{KeyValueStoreARN='arn:aws:cloudfront::111122223333:key-value-
store/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111'}]} \
  FunctionMetadata:
    CreatedTime: '2021-04-18T20:38:56.915000+00:00'
    FunctionARN: arn:aws:cloudfront::111122223333:function/MaxAge
    LastModifiedTime: '2023-11-19T20:38:56.915000+00:00'
    Stage: DEVELOPMENT
  Name: MaxAge
  Status: UNPUBLISHED
Location: https://cloudfront.amazonaws.com/2020-05-31/function/
arn:aws:cloudfront::function/MaxAge
```

A maioria das informações é repetida a partir da solicitação. Outras informações são adicionadas pelo CloudFront.

Observações

- ETag: esse valor muda sempre que você modifica o armazenamento de valores de chave. Use esse valor e o nome da função para referenciar a função futuramente. Sempre use a ETag atual.
- FunctionARN: o ARN da função do CloudFront.
- 111122223333: a Conta da AWS.
- Stage: o estágio da função (LIVE ou DEVELOPMENT).
- Status: o status da função (PUBLISHED ou UNPUBLISHED).

Depois de criar a função, ela é adicionada ao estágio DEVELOPMENT. Recomendamos [testar sua função](#) antes de [publicá-la](#). Depois de publicar a função, ela muda para o estágio LIVE.

Testar funções

Antes de implantar a função no estágio ativo (produção), é possível testar a função para verificar se ela funciona conforme o esperado. Para testar uma função, especifique um objeto de evento que represente uma solicitação ou uma resposta HTTP que a distribuição do CloudFront pode receber em produção.

O CloudFront Functions faz o seguinte:

1. Executa a função usando o objeto de evento fornecido como entrada.
2. Retorna o resultado da função (o objeto de evento modificado), bem como os logs de função ou as mensagens de erro e a utilização de computação da função. Para obter mais informações sobre as métricas de utilização de computação, consulte [the section called “Noções básicas de utilização de computação”](#).

Sumário

- [Configurar o objeto de evento](#)
- [Testar a função](#)
- [Noções básicas de utilização de computação](#)

Configurar o objeto de evento

Antes de testar uma função, é necessário configurar o objeto de evento com o qual testá-la. Há várias opções.

Opção 1: configurar um objeto de evento sem salvá-lo

É possível configurar um objeto de evento no editor visual no console do CloudFront e não salvá-lo.

É possível usar esse objeto de evento para testar a função no console do CloudFront, mesmo que ele não esteja salvo.

Opção 2: criar um objeto de evento no editor visual

É possível configurar um objeto de evento no editor visual no console do CloudFront e não salvá-lo. É possível criar dez objetos de evento para cada função para poder, por exemplo, testar diferentes entradas possíveis.

Ao criar o objeto de evento dessa forma, é possível usar o objeto de evento para testar a função no console do CloudFront. Não é possível usá-lo para testar a função usando uma API ou um SDK da AWS.

Opção 3: criar um objeto de evento usando um editor de texto

É possível usar um editor de texto para criar um objeto de evento no formato JSON. Para obter informações sobre a estrutura de um objeto de evento, consulte [Estrutura de eventos](#).

É possível usar esse objeto de evento para testar a função usando a CLI. Mas não é possível usá-lo para testar a função no console do CloudFront.

Como criar um objeto de evento (opção 1 ou 2)

1. Faça login no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions> e selecione a página Funções.

Escolha a função que você deseja testar.
2. Na página de detalhes da função, selecione a guia Testar.
3. Em Tipo de evento, selecione uma das seguintes opções:
 - Selecione Solicitação do visualizador se a função modificar uma solicitação HTTP ou gerar uma resposta com base na solicitação. A seção Solicitação é exibida.
 - Selecione Resposta do visualizador. As seções Solicitação e Resposta são exibidas.
4. Preencha todos os campos a serem incluídos no evento. É possível selecionar Editar JSON para visualizar o JSON bruto.
5. (Opcional) Para salvar o evento, escolha Salvar e, em Salvar evento de teste, insira um nome e escolha Salvar.

Também é possível selecionar Editar JSON, copiar o JSON bruto e salvá-lo no próprio arquivo, fora do CloudFront.

Como criar um objeto de evento (opção 3)

Crie o objeto de evento usando um editor de texto. Armazene o arquivo em um diretório ao qual o computador possa se conectar.

Siga estas diretrizes:

- Omita os campos `distributionDomainName`, `distributionId` e `requestId`.
- Os nomes de cabeçalhos, os cookies e as strings de consulta devem estar em letras minúsculas.

Uma opção para criar um objeto de evento dessa forma é criar um exemplo usando o editor visual. O exemplo deve estar formatado corretamente. Depois, copie o JSON bruto, cole-o em um editor de texto e salve o arquivo.

Para ter mais informações sobre a estrutura de um evento, consulte [Estrutura de eventos](#).

Testar a função

Você pode testar uma função no console do CloudFront ou com a AWS Command Line Interface (AWS CLI).

Console

Para testar a função

1. Faça login no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions> e selecione a página Funções.
2. Escolha a função que você deseja testar.
3. Selecione a guia Testar.
4. Verifique se o evento correto é exibido. Para sair do evento exibido no momento, selecione outro evento no campo Selecionar evento de teste.
5. Escolha Testar função. O console mostra a saída da função, incluindo os logs da função e a utilização de computação.

CLI

É possível testar uma função usando o comando `aws cloudfront test-function`.

Para testar a função

1. Abra a janela de linha de comando.
2. Execute o comando a seguir no diretório que contém o arquivo especificado.

Este exemplo usa a notação `fileb://` para transmitir o arquivo de objeto de evento. Ele também inclui quebras de linha para tornar o comando mais legível.

```
aws cloudfront test-function \  
  --name MaxAge \  
  --if-match ETVABCEXAMPLE \  
  --event-object fileb://event-maxage-test01.json \  
  --stage DEVELOPMENT
```

Observações

- Você faz referência à função pelo nome e pela ETag (no parâmetro `if-match`). Você faz referência ao objeto de evento pela localização no sistema de arquivos.
- O estágio pode ser `DEVELOPMENT` ou `LIVE`.

Quando o comando é bem-sucedido, a saída é semelhante à seguinte:

```
TestResult:  
  ComputeUtilization: '21'  
  FunctionErrorMessage: ''  
  FunctionExecutionLogs: []  
  FunctionOutput: '{"response":{"headers":{"cloudfront-functions":  
{"value":"generated-by-CloudFront-Functions"},"location":{"value":"https://  
aws.amazon.com/cloudfront/"}},"statusDescription":"Found","cookies":  
{},"statusCode":302}}'  
  FunctionSummary:  
    FunctionConfig:  
      Comment: MaxAge function  
      Runtime: cloudfront-js-2.0  
      KeyValueStoreAssociations= \  
        {Quantity=1, \  
          Items=[{KeyValueStoreARN='arn:aws:cloudfront::111122223333:key-value-  
store/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111'}]} \  
    FunctionMetadata:
```

```
CreatedTime: '2021-04-18T20:38:56.915000+00:00'  
FunctionARN: arn:aws:cloudfront::111122223333:function/MaxAge  
LastModifiedTime: '2023-17-20T10:38:57.057000+00:00'  
Stage: DEVELOPMENT  
Name: MaxAge  
Status: UNPUBLISHED
```

Observações

- `FunctionExecutionLogs` contém uma lista de linhas de log que a função escreveu em instruções `console.log()` (se houver).
- `ComputeUtilization` contém informações sobre como executar a função. Consulte [the section called “Noções básicas de utilização de computação”](#).
- O `FunctionOutput` contém o objeto de evento que a função retornou.

Noções básicas de utilização de computação

Compute utilization (Utilização de computação) é a quantidade de tempo que a função levou para ser executada como uma porcentagem do tempo máximo permitido. Por exemplo, um valor de 35 significa que a função foi concluída em 35% do tempo máximo permitido.

Se uma função exceder continuamente o tempo máximo permitido, o CloudFront limitará a função. A lista a seguir explica a probabilidade de uma função ser limitada com base no valor de utilização da computação.

Valor de utilização de computação:

- De 1 a 50: a função está confortavelmente abaixo do tempo máximo permitido e deve funcionar sem controle de utilização.
- De 51 a 70: a função está próxima do tempo máximo permitido. É aconselhável otimizar o código da função.
- De 71 a 100: a função está muito próxima ou excede o tempo máximo permitido. É provável que o CloudFront restrinja essa função se você a associar a uma distribuição.

Atualizar funções

É possível atualizar uma função a qualquer momento. As alterações são feitas somente na versão da função que está no estágio DEVELOPMENT. Para copiar as atualizações do estágio DEVELOPMENT em LIVE, é necessário [publicar a função](#).

É possível atualizar o código de uma função no console do CloudFront ou com a AWS Command Line Interface (AWS CLI).

Console

Como atualizar um código de função

1. Faça login no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions> e selecione a página Funções.

Escolha a função a ser atualizada.

2. Escolha Editar e faça as seguintes alterações:
 - Atualize todos os campos na seção Detalhes.
 - Altere ou remova o armazenamento de valores de chave associado. Para obter mais informações sobre armazenamentos de chave-valor, consulte [the section called “KeyValueStore do CloudFront”](#).
 - Altere o código da função. Selecione a guia Criar, faça alterações e, depois, escolha Salvar alterações para salvar as alterações no código.

CLI

Para atualizar o código da função

1. Abra a janela de linha de comando.
2. Execute o seguinte comando .

Este exemplo usa a notação `fileb://` para transmitir o arquivo. Ele também inclui quebras de linha para tornar o comando mais legível.

```
aws cloudfront update-function \
  --name MaxAge \
  --function-config '{"Comment":"Max Age 2 years","Runtime":"cloudfront-
js-2.0","KeyValueStoreAssociations":{"Quantity":1,"Items":
[{"KeyValueStoreARN":"arn:aws:cloudfront::111122223333:key-value-store/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"}]}' \
  --function-code fileb://function-max-age-v1.js \
  --if-match ETVABCEXAMPLE
```

Observações

- É necessário identificar a função pelo nome e pela ETag (no parâmetro `if-match`). Use a ETag atual. É possível obtê-la usando uma operação de descrição.
- É necessário incluir o `function-code`, mesmo que não deseje alterá-lo.
- Tenha cuidado com a `function-config`. É necessário transmitir tudo o que deseja manter na configuração. Especificamente, trate o armazenamento de chave-valor da seguinte forma:
 - Para reter a associação do armazenamento de valores de chave existente (se houver), especifique o nome do armazenamento existente.
 - Para alterar a associação, especifique o nome do novo armazenamento de valores de chave.
 - Para remover a associação, omita o parâmetro `KeyValueStoreAssociations`.

Quando o comando é bem-sucedido, a saída é semelhante à seguinte:

```
ETag: ETVXYZEXAMPLE
FunctionSummary:
  FunctionConfig:
    Comment: Max Age 2 years \
    Runtime: cloudfront-js-2.0 \
    KeyValueStoreAssociations= \
      {Quantity=1, \
        Items=[{KeyValueStoreARN='arn:aws:cloudfront::111122223333:key-value-
store/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111'}]} \
    FunctionMetadata: \
```

```
CreatedTime: '2021-04-18T20:38:56.915000+00:00' \  
FunctionARN: arn:aws:cloudfront::11112223333:function/MaxAge \  
LastModifiedTime: '2023-12-19T23:41:15.389000+00:00' \  
Stage: DEVELOPMENT \  
Name: MaxAge \  
Status: UNPUBLISHED
```

A maioria das informações é repetida a partir da solicitação. Outras informações são adicionadas pelo CloudFront.

Observações

- ETag: esse valor muda sempre que você modifica o armazenamento de valores de chave.
- FunctionARN: o ARN da função do CloudFront.
- Stage: o estágio da função (LIVE ou DEVELOPMENT).
- Status: o status da função (PUBLISHED ou UNPUBLISHED).

Publicar as funções

Quando você publica a função, ela é copiada do estágio DEVELOPMENT para LIVE.

Se nenhum comportamento de cache estiver associado à função, a publicação dela permitirá associá-la a um comportamento de cache. Você só pode associar comportamentos de cache a funções que estão na etapa LIVE.

Important

- Antes de publicar, recomendamos [testar a função](#).
- Quando você publicar a função, todos os comportamentos de cache associados a ela começarão automaticamente a usar a cópia recém-publicada assim que as distribuições terminarem de ser implantadas.

Você pode publicar uma função no console do CloudFront ou com a AWS CLI.

Console

Como publicar uma função

1. Faça login no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions> e selecione a página Funções.
2. Escolha a função a ser atualizada.
3. Selecione a guia Publicar e, depois, Publicar função. Se a função já estiver anexada a um ou mais comportamentos de cache, selecione Publicar e atualizar.
4. (Opcional) Para ver as distribuições associadas à função, selecione Associated CloudFront distributions (Distribuições associadas do CloudFront) para expandir essa seção.

Quando for bem-sucedido, será exibido um banner na parte superior da página que diz **Nome da função** publicada com êxito. Você também pode escolher a guia Build (Criar) e, em seguida, escolher Live (Ao vivo) para ver a versão ao vivo do código de função.

CLI

Como publicar uma função

1. Abra a janela de linha de comando.
2. Execute o seguinte comando `aws cloudfront publish-function`. No exemplo, as quebras de linha são fornecidas para tornar o exemplo mais legível.

```
aws cloudfront publish-function \  
  --name MaxAge \  
  --if-match ETVXYZEXAMPLE
```

Quando o comando é bem-sucedido, a saída é semelhante à seguinte:

```
FunctionSummary:  
  FunctionConfig:  
    Comment: Max Age 2 years  
    Runtime: cloudfront-js-2.0  
  FunctionMetadata:  
    CreatedTime: '2021-04-18T21:24:21.314000+00:00'  
    FunctionARN: arn:aws:cloudfront::111122223333:function/ExampleFunction  
    LastModifiedTime: '2023-12-19T23:41:15.389000+00:00'  
    Stage: LIVE
```

Name: MaxAge
Status: UNASSOCIATED

Associar funções a distribuições

Para usar uma função com uma distribuição, associe a função a um ou mais comportamentos de cache na distribuição. Você pode associar uma função a vários comportamentos de cache em várias distribuições.

É possível associar uma função a qualquer um dos seguintes itens:

- Um comportamento de cache existente.
- Um novo comportamento de cache em uma distribuição existente.
- Um novo comportamento de cache em uma nova distribuição.

Quando você associa uma função a um comportamento de cache, você deve selecionar um tipo de evento. O tipo de evento determina quando o CloudFront executa a função.

É possível escolher entre os seguintes tipos de evento:

- Solicitação do visualizador: a função é executada quando o CloudFront recebe uma solicitação de um visualizador.
- Resposta do visualizador: a função é executada antes que o CloudFront retorne uma resposta ao visualizador.

Não é possível usar tipos de evento voltados para a origem (solicitação de origem e resposta de origem) com o CloudFront Functions. Em vez disso, você pode usar o Lambda@Edge. Para ter mais informações, consulte [Eventos do CloudFront que podem acionar uma função do Lambda@Edge](#).

Note

Antes de associar uma função, você deve [publicá-la](#) na fase LIVE.

É possível associar uma função a uma distribuição no console do CloudFront ou à AWS Command Line Interface (AWS CLI). O procedimento a seguir mostra como associar uma função a um comportamento de cache existente.

Console

Como associar uma função a um comportamento de cache existente

1. Faça login no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions> e selecione a página Funções.
2. Escolha a função que você deseja associar.
3. Na página Função, selecione a guia Publicar.
4. Selecione a função Publish (Publicar).
5. Escolha Add association. Na caixa de diálogo exibida, selecione uma distribuição, um tipo de evento e/ou um comportamento de cache.

Para o tipo de evento, selecione quando deseja que essa função seja executada:

- Solicitação do visualizador: execute a função sempre que o CloudFront receber uma solicitação.
 - Resposta do visualizador: execute a função sempre que o CloudFront exibir uma resposta.
6. Para salvar a configuração, selecione Adicionar associação.

O CloudFront associa a distribuição à função. Aguarde alguns minutos para que a distribuição associada termine a implantação. É selecionar Visualizar distribuição na página de detalhes da função para conferir o andamento.

CLI

Como associar uma função a um comportamento de cache existente

1. Abra a janela de linha de comando.
2. Insira o comando a seguir para salvar a configuração da distribuição cujo comportamento de cache você deseja associar a uma função. Esse comando salva a configuração de distribuição em um arquivo chamado `dist-config.yaml`. Para usar esse comando, faça o seguinte:
 - Substitua *DistributionID* pelo ID da distribuição.
 - Execute o comando em uma linha. No exemplo, as quebras de linha são fornecidas para tornar o exemplo mais legível.

```
aws cloudfront get-distribution-config \  
  --id DistributionID \  
  --output yaml > dist-config.yaml
```

Quando o comando é bem-sucedido, a AWS CLI não retorna nenhuma saída.

3. Abra o arquivo chamado `dist-config.yaml` que você criou. Edite o arquivo para fazer as alterações a seguir.
 - a. Renomeie o campo `ETag` para `IfMatch`, mas não altere o valor do campo.
 - b. No comportamento do cache, localize o objeto chamado `FunctionAssociations`. Atualize esse objeto para adicionar uma associação de função. A sintaxe YAML para uma associação de função se parece com o exemplo a seguir.
 - O exemplo a seguir mostra um tipo de evento de solicitação de visualizador (`trigger`). Para usar um tipo de evento de resposta do visualizador, substitua `viewer-request` por `viewer-response`.
 - Substitua `arn:aws:cloudfront::111122223333:function/ExampleFunction` pelo nome do recurso da Amazon (ARN) da função que você está associando a esse comportamento de cache. Para obter o ARN da função, você pode usar o comando `aws cloudfront list-functions`.

```
FunctionAssociations:  
  Items:  
    - EventType: viewer-request  
      FunctionARN: arn:aws:cloudfront::111122223333:function/ExampleFunction  
  Quantity: 1
```

- c. Depois de fazer essas alterações, salve o arquivo.
4. Use o comando a seguir para atualizar a distribuição, adicionando a associação de função. Para usar esse comando, faça o seguinte:
 - Substitua `DistributionID` pelo ID da distribuição.
 - Execute o comando em uma linha. No exemplo, as quebras de linha são fornecidas para tornar o exemplo mais legível.

```
aws cloudfront update-distribution \  
  --id DistributionID \  
  --cli-input-yaml file://dist-config.yaml
```

Quando o comando é bem-sucedido, você vê uma saída como a seguinte que descreve a distribuição que foi atualizada apenas com a associação de função. O exemplo de saída a seguir é truncado para legibilidade.

```
Distribution:  
  ARN: arn:aws:cloudfront::111122223333:distribution/EBEDLT3BGRBBW  
  ... truncated ...  
DistributionConfig:  
  ... truncated ...  
DefaultCacheBehavior:  
  ... truncated ...  
FunctionAssociations:  
  Items:  
  - EventType: viewer-request  
    FunctionARN: arn:aws:cloudfront::111122223333:function/ExampleFunction  
    Quantity: 1  
  ... truncated ...  
DomainName: d111111abcdef8.cloudfront.net  
Id: EDFDVBD6EXAMPLE  
LastModifiedTime: '2021-04-19T22:39:09.158000+00:00'  
Status: InProgress  
ETag: E2VJGGQEG1JT8S
```

O Status da distribuição muda para `InProgress` enquanto a distribuição é reimplantada. Quando a nova configuração de distribuição atingir um local da borda do CloudFront, esse local começará a usar a função associada. Quando a distribuição estiver totalmente implantada, o Status mudará de volta para `Deployed`. Isso indica que a função do CloudFront associada está ativa em todos os locais da borda do CloudFront no mundo todo. Normalmente, isso demora alguns minutos.

KeyValueStore do Amazon CloudFront

O KeyValueStore do CloudFront é um datastore de valor de chave seguro, global e de baixa latência que permite acesso de leitura por meio do [CloudFront Functions](#), viabilizando uma lógica personalizada avançada nos locais da borda do CloudFront.

Com o KeyValueCollection do CloudFront, você faz atualizações no código da função e nos dados associados a uma função, de modo independente uns dos outros. Essa separação simplifica o código da função e facilita a atualização dos dados sem a necessidade de implantar alterações no código.

Note

Para usar o CloudFront KeyValueCollection, a função do CloudFront deve usar o [runtime 2.0 do JavaScript](#).

Veja a seguir o procedimento geral para usar pares de chave-valor:

- Crie armazenamentos de chave-valor e preencha-os com um conjunto de pares de chave-valor. É possível adicionar seus armazenamentos de chave-valor a um bucket do Amazon S3 ou inseri-los manualmente.
- Associe os armazenamentos de chave-valor ao CloudFront Functions.
- No código da função, use o nome da chave para recuperar o valor associado a ela ou para avaliar se existe uma chave. Consulte mais informações sobre o uso de pares de chave-valor no código da função e sobre métodos auxiliares em [the section called “Métodos auxiliares para armazenamentos de chave-valor”](#).

Casos de uso

É possível usar pares de chave-valor para os seguintes exemplos:

- Regravações ou redirecionamentos de URL: o par de chave-valor pode conter os URLs regravados ou os URLs de redirecionamento.
- Testes A/B e sinalizadores de recursos: é possível criar uma função para realizar experimentos atribuindo uma porcentagem do tráfego a uma versão específica do site.
- Autorização de acesso: é possível implementar o controle de acesso para permitir ou negar solicitações com base nos critérios definidos e nos dados armazenados no armazenamento de chave-valor.

Formatos compatíveis de valores

É possível armazenar o valor em um par de chave-valor em qualquer um dos seguintes formatos:

- String
- String codificada por bytes
- JSON

Segurança

O CloudFront Functions e todos os dados de armazenamento de chave-valor são tratados com segurança, da seguinte forma:

- O CloudFront criptografa cada armazenamento de chave-valor em repouso e em trânsito (ao ler ou gravar no armazenamento de chave-valor) ao chamar as operações da API [KeyValueStore do CloudFront](#).
- Quando a função é executada, o CloudFront descriptografa cada par de chave-valor na memória nos locais da borda do CloudFront.

Para começar a usar o KeyValueStore do CloudFront, consulte os tópicos a seguir.

Tópicos

- [Trabalhar com o armazenamento de chave-valor](#)
- [Trabalhar com dados de chave-valor](#)
- Para ter mais informações sobre os conceitos básicos do KeyValueStore do CloudFront, consulte a postagem do blog [Apresentação do KeyValueStore do Amazon CloudFront](#).AWS

Trabalhar com o armazenamento de chave-valor

É necessário criar um armazenamento de chave-valor para armazenar os pares de chave-valor que você deseja usar no CloudFront Functions.


Depois de criar os armazenamentos de chave-valor e adicionar os pares de chave-valor, você pode usar os elementos de chave-valor no código da função do CloudFront.

Para começar, consulte os seguintes tópicos:

Tópicos

- [Criar um armazenamento de chave-valor](#)


- [Associar o armazenamento de chave-valor à função](#)
- [Atualizar um armazenamento de chave-valor](#)
- [Obter uma referência a um armazenamento de chave-valor](#)
- [Excluir um armazenamento de chave-valor](#)
- [Formato de arquivo para pares de chave-valor](#)

 Note

O JavaScript runtime 2.0 inclui alguns métodos auxiliares para trabalhar com chave-valor no código da função. Para ter mais informações, consulte [the section called “Métodos auxiliares para armazenamentos de chave-valor”](#).

Criar um armazenamento de chave-valor

Você pode criar um armazenamento de chave-valor e os pares de chave-valor ao mesmo tempo. Também é possível criar um armazenamento de chave-valor vazio e adicionar pares de chave-valor posteriormente.

 Note

Se você especificar sua fonte de dados a partir de um bucket do Amazon S3, deverá ter as permissões `s3:GetObject` e `s3:GetBucketLocation` para esse bucket. Se você não tiver essas permissões, o CloudFront não poderá criar seu armazenamento de chave-valor com êxito.

Decida se deseja adicionar pares de chave-valor ao mesmo tempo em que cria o armazenamento de chave-valor. É possível importar os pares de chave-valor usando o console do CloudFront, a API do CloudFront ou os SDKs da AWS. No entanto, só é possível importar o arquivo de pares de chave-valor ao criar inicialmente o armazenamento de chave-valor.

Para criar um arquivo de pares de chave-valor, consulte [Formato de arquivo para pares de chave-valor](#).

Console

Como criar um armazenamento de chave-valor

1. Faça login no AWS Management Console e abra a página Functions (Funções) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions>.
2. Escolha a guia KeyValueStores e selecione Criar KeyValueStore.
3. Informe um nome e uma descrição opcional para o armazenamento de valores de chave.
4. Preencha o URI do S3:
 - Se você tem um arquivo de pares de chave-valor, insira o caminho para o bucket do Amazon S3 no qual armazenou o arquivo.
 - Deixe esse campo em branco se planeja inserir os pares de chave-valor manualmente.
5. Escolha Criar. O armazenamento de chave-valor existe agora.

A página de detalhes do novo armazenamento de valores de chave é exibida. As informações na página incluem o ID e o ARN do armazenamento de valores de chave.

- O ID é uma string de caracteres aleatória exclusiva na Conta da AWS.
- O ARN tem esta sintaxe:

Conta da AWS:key-value-store/o ID de armazenamentos de chave-valor

6. Veja a seção Pares de chave-valor. Se você importou um arquivo, essa seção mostrará alguns pares de chave-valor. Você pode fazer o seguinte:
 - Se você importou um arquivo, também poderá adicionar mais valores manualmente.
 - Se você não importou um arquivo de um bucket do Amazon S3 e deseja adicionar pares de chave-valor agora, poderá concluir a próxima etapa.
 - É possível ignorar essa etapa e adicionar os pares de chave-valor posteriormente.
7. Como adicionar os pares agora:
 - a. Selecione Adicionar pares de chave-valor.
 - b. Selecione Adicionar par e insira um nome e um valor. Repita essa etapa para adicionar mais pares.
 - c. Ao terminar, selecione Salvar alterações para salvar todos os pares de chave-valor no armazenamento de valores de chave. Na caixa de diálogo exibida, selecione Concluído.

- Para associar agora o armazenamento de valores de chave a uma função agora, preencha a seção Funções associadas. Para obter mais informações, consulte [???](#) ou [???](#).

Também é possível associar a função posteriormente, na página de detalhes do armazenamento de valores de chave ou na página de detalhes da função.

AWS CLI

Como criar um armazenamento de chave-valor

- Execute o comando a seguir para criar um armazenamento de chave-valor e importar os pares de chave-valor de um bucket do Amazon S3.

```
aws cloudfront create-key-value-store \  
  --name=keyvaluestore1 \  
  --comment="This is my key value store file" \  
  --import-source=SourceType=S3,SourceARN=arn:aws:s3:::amzn-s3-demo-  
bucket1/kvs-input.json
```

Resposta

```
{  
  "ETag": "ETVABCEXAMPLE",  
  "Location": "https://cloudfront.amazonaws.com/2020-05-31/key-value-store/  
arn:aws:cloudfront::123456789012:key-value-store/8aa76c93-3198-462c-aaf6-  
example",  
  "KeyValueStore": {  
    "Name": "keyvaluestore1",  
    "Id": "8aa76c93-3198-462c-aaf6-example",  
    "Comment": "This is my key value store file",  
    "ARN": "arn:aws:cloudfront::123456789012:key-value-  
store/8aa76c93-3198-462c-aaf6-example",  
    "Status": "PROVISIONING",  
    "LastModifiedTime": "2024-08-06T22:19:10.813000+00:00"  
  }  
}
```

API

Como criar um armazenamento de chave-valor

1. Use a operação [CreateKeyValueStore do CloudFront](#). A operação usa vários parâmetros:
 - Um nome do armazenamento de chave-valor.
 - Um parâmetro `comment` que inclua um comentário.
 - Um parâmetro `import-source` que permita importar pares de chave-valor de um arquivo armazenado em um bucket do Amazon S3. É possível importar de um arquivo somente durante a criação inicial do armazenamento de chave-valor. Para ter informações sobre a estrutura de arquivos, consulte [the section called “Formato de arquivo para pares de chave-valor”](#).

A resposta da operação inclui as seguintes informações:

- Os valores transmitidos na solicitação, incluindo o nome atribuído.
- Dados, como o horário de criação.
- Uma ETag (por exemplo, ETVABCEXAMPLE), o ARN que inclui o nome do armazenamento de chave-valor (por exemplo, `arn:aws:cloudfront::123456789012:key-value-store/keyvaluestore1`).

Você usará alguma combinação da ETag, do ARN e do nome para trabalhar com o armazenamento de chave-valor de forma programática.

Status do armazenamento de chave-valor

Ao criar um armazenamento de chave-valor, o armazenamento de dados pode apresentar os seguintes valores de status.

Value	Descrição
Provisionamento	O armazenamento de chave-valor foi criado e o CloudFront está processando a fonte de dados especificada.
Ready	O armazenamento de chave-valor foi criado e o CloudFront processou com êxito a fonte de dados especificada.

Value	Descrição
Falha na importação	O CloudFront não conseguiu processar a fonte de dados especificada. Esse status pode aparecer se o formato do arquivo não for válido ou exceder o limite de tamanho. Para ter mais informações, consulte Formato de arquivo para pares de chave-valor .

Associar o armazenamento de chave-valor à função

Depois de criar o armazenamento de chave-valor, é possível atualizar a função para associá-la a esse armazenamento. É necessário fazer essa associação para usar os pares de chave-valor desse armazenamento nessa função. As seguintes regras se aplicam:

- Uma função pode ter apenas um armazenamento de chave-valor.
- É possível associar o mesmo armazenamento de chave-valor a várias funções.

Console

Como associar o armazenamento de chave-valor a uma função

1. Faça login no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions> e selecione a página Funções.
2. Escolha o nome da função.
3. Acesse a seção Associar KeyValueStore e selecione Associar KeyValueStore existente.
4. Selecione o armazenamento de chave-valor que contém os pares de chave-valor na função e escolha Associar KeyValueStore.

O CloudFront associa imediatamente o armazenamento à função. Não é necessário salvar a função.

5. Para especificar um armazenamento de chave-valor diferente, escolha Atualizar KeyValueStore associado, selecione outro nome de armazenamento de chave-valor e escolha Associar KeyValueStore.

Para ter mais informações, consulte [the section called “Atualizar funções”](#).

AWS CLI

Como associar o armazenamento de chave-valor a uma função

- Execute o comando a seguir para atualizar a função *MaxAge* e associar um recurso de armazenamento de chave-valor.

```
aws cloudfront update-function \  
  --name MaxAge \  
  --function-config '{"Comment":"Max Age 2 years","Runtime":"cloudfront-  
js-2.0","KeyValueStoreAssociations":{"Quantity":1,"Items":  
[{"KeyValueStoreARN":"arn:aws:cloudfront::123456789012:key-value-  
store/8aa76c93-3198-462c-aaf6-example"}]}' \  
  --function-code fileb://function-max-age-v1.js \  
  --if-match ETVABCEXAMPLE
```

- Para associar um armazenamento de chave-valor a uma função, especifique o parâmetro `KeyValueStoreAssociations` e o ARN do armazenamento de chave-valor.
- Para alterar a associação, especifique outro ARN de armazenamento de chave-valor.
- Para remover a associação, remova o parâmetro `KeyValueStoreAssociations`.

Para ter mais informações, consulte [the section called “Atualizar funções”](#).

API

Como associar o armazenamento de chave-valor a uma função

- Use a operação de API [UpdateFunction](#). Para ter mais informações, consulte [the section called “Atualizar funções”](#).

Observações

- Se você modificar um armazenamento de chave-valor sem alterar os pares de chave-valor, ou se você modificar somente os pares de chave-valor no armazenamento de chave-valor, não será necessário associar esse armazenamento novamente. Você também não precisa republicar a função.

No entanto, recomendamos que você teste a função para verificar se ela funciona conforme o esperado. Para ter mais informações, consulte [Testar funções](#).

- É possível visualizar todas as funções que usam armazenamentos de chave-valor específicos. No console do CloudFront, selecione a página de detalhes do armazenamento de chave-valor.

Atualizar um armazenamento de chave-valor

Ao atualizar um armazenamento de chave-valor, é possível alterar os pares de chave-valor ou alterar a associação entre esse armazenamento e a função.

Console

Como atualizar um armazenamento de chave-valor

1. Faça login no AWS Management Console e abra a página Functions (Funções) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions>.
2. Selecione a guia KeyValueStores.
3. Selecione o armazenamento de chave-valor que deseja atualizar.
 - Para atualizar os pares de chave-valor, selecione Editar na seção Pares de chave-valor. É possível adicionar ou excluir qualquer par de chave-valor. Também é possível alterar o valor de um par de chave-valor existente. Ao concluir, escolha Salvar alterações.
 - Para atualizar a associação desse armazenamento de chave-valor, selecione Ir para funções. Para ter mais informações, consulte [the section called “Associar o armazenamento de chave-valor à função”](#).

AWS CLI

Como atualizar um armazenamento de chave-valor

1. Alterar pares de chave-valor: é possível adicionar mais pares de chave-valor, excluir um ou mais pares de chave-valor e alterar o valor de um par existente. Para ter mais informações, consulte [Trabalhar com dados de chave-valor](#).

2. Alterar a associação da função para o armazenamento de chave-valor: para atualizar a associação da função para o armazenamento de chave-valor, consulte [Associar o armazenamento de chave-valor à função](#).


 Tip

Você precisará do ARN do armazenamento de valores de chave. Para ter mais informações, consulte [the section called “Obter uma referência a um armazenamento de chave-valor”](#).

API

Como atualizar um armazenamento de chave-valor

1. Alterar pares de chave-valor: é possível adicionar mais pares de chave-valor, excluir um ou mais pares de chave-valor e alterar o valor de um par existente. Para ter mais informações, consulte [Trabalhar com dados de chave-valor](#).
2. Alterar a associação da função para o armazenamento de chave-valor: para atualizar a associação da função para o armazenamento de chave-valor, use a operação de API [UpdateFunction](#). Para ter mais informações, consulte [the section called “Atualizar funções”](#).

 Tip

Você precisará do ARN do armazenamento de valores de chave. Para ter mais informações, consulte [the section called “Obter uma referência a um armazenamento de chave-valor”](#).

Obter uma referência a um armazenamento de chave-valor

Para trabalhar com os armazenamentos de chave-valor de forma programática, é necessário ter a ETag e o nome do armazenamento de chave-valor.

Para ter os dois valores, é possível usar a AWS Command Line Interface (AWS CLI) ou a API do CloudFront.

AWS CLI

Como obter uma referência do armazenamento de chave-valor

1. Para exibir uma lista de armazenamentos de chave-valor, execute o comando para encontrar o nome do armazenamento de chave-valor que deseja alterar.

```
aws cloudfront list-key-value-stores
```

2. Na resposta, encontre o nome do armazenamento de chave-valor desejado.

Resposta

```
{
  "KeyValueStoreList": {
    "Items": [
      {
        "Name": "keyvaluestore3",
        "Id": "37435e19-c205-4271-9e5c-example3",
        "ARN": "arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-example3",
        "Status": "READY",
        "LastModifiedTime": "2024-05-08T14:50:18.876000+00:00"
      },
      {
        "Name": "keyvaluestore2",
        "Id": "47970d59-6408-474d-b850-example2",
        "ARN": "arn:aws:cloudfront::123456789012:key-value-store/47970d59-6408-474d-b850-example2",
        "Status": "READY",
        "LastModifiedTime": "2024-05-30T21:06:22.113000+00:00"
      },
      {
        "Name": "keyvaluestore1",
        "Id": "8aa76c93-3198-462c-aaf6-example",
        "ARN": "arn:aws:cloudfront::123456789012:key-value-store/8aa76c93-3198-462c-aaf6-example",
        "Status": "READY",
        "LastModifiedTime": "2024-08-06T22:19:30.510000+00:00"
      }
    ]
  }
}
```



```
}
```

3. Execute o comando a seguir para exibir a ETag do armazenamento de chave-valor especificado.

```
aws cloudfront describe-key-value-store \  
  --name=keyvaluestore1
```

Resposta

```
{  
  "ETag": "E3UN6WX5RR02AG",  
  "KeyValueStore": {  
    "Name": "keyvaluestore1",  
    "Id": "8aa76c93-3198-462c-aaf6-example",  
    "Comment": "This is an example KVS",  
    "ARN": "arn:aws:cloudfront::123456789012:key-value-  
store/8aa76c93-3198-462c-aaf6-example",  
    "Status": "READY",  
    "LastModifiedTime": "2024-08-06T22:19:30.510000+00:00"  
  }  
}
```

API

Como obter uma referência do armazenamento de chave-valor

1. Use a operação de API [CloudFront ListKeyValueStores](#) para retornar uma lista de armazenamentos de valores-chave. Encontre o nome do armazenamento de chave-valor que deseja alterar.
2. Use a operação de API [CloudFront DescribeKeyValueStore](#) e especifique o nome do armazenamento de chave-valor que você retornou da etapa anterior.

A resposta inclui um UUID, o ARN do armazenamento de chave-valor e a ETag do armazenamento de chave-valor.

- Uma ETag, como E3UN6WX5RR02AG.
- O UUID tem 128 bits, como 8aa76c93-3198-462c-aaf6-example.

- O ARN inclui o número da Conta da AWS, o key-value-store constante e o UUID, como o seguinte exemplo:

```
arn:aws:cloudfront::123456789012:key-value-store/8aa76c93-3198-462c-aaf6-example
```

Consulte mais informações sobre a operação `DescribeKeyValueStore` em [the section called “Sobre o KeyValueStore do CloudFront”](#).

Excluir um armazenamento de chave-valor

É possível excluir o armazenamento de chave-valor usando a API ou o console do Amazon CloudFront.

Console

Como excluir um armazenamento de chave-valor

1. Faça login no AWS Management Console e abra a página Functions (Funções) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions>.
2. Escolha o nome da função.
3. Na seção KeyValueStore associado, verifique se um armazenamento de chave-valor está associado à função. Se estiver, remova a associação escolhendo Dissociar KeyValueStore e selecione Remover associação.
4. No painel de navegação, selecione a página Funções e escolha a guia KeyValueStores.
5. Selecione o armazenamento de chave-valor que deseja excluir e selecione Excluir.

AWS CLI

Como excluir um armazenamento de chave-valor

1. Obtenha a ETag e o nome dos armazenamentos de chave-valor. Para ter mais informações, consulte [the section called “Obter uma referência a um armazenamento de chave-valor”](#).
2. Verifique se os armazenamentos de chave-valor estão associados a uma função. Se estiver, remova a associação. Para obter mais informações sobre essas duas etapas, consulte [???](#).
3. Depois que você tiver o nome e a ETag do armazenamento de chave-valor e ele não estiver mais associado a uma função, você poderá excluí-lo.

Execute o comando a seguir para excluir o armazenamento de chave-valor especificado.

```
aws cloudfront delete-key-value-store \  
  --name=keyvaluestore1 \  
  --if-match=E3UN6WX5RR02AG
```

API

Como excluir um armazenamento de chave-valor

1. Obtenha a ETag e o nome dos armazenamentos de chave-valor. Para ter mais informações, consulte [the section called “Obter uma referência a um armazenamento de chave-valor”](#).
2. Verifique se os armazenamentos de chave-valor estão associados a uma função. Se estiver, remova a associação. Para obter mais informações sobre essas duas etapas, consulte [???](#).
3. Para excluir o armazenamento de chave-valor, use a operação de API [DeleteKeyValueStore](#) do CloudFront.

Formato de arquivo para pares de chave-valor

Ao criar um arquivo codificado em UTF-8, use o seguinte formato JSON:

```
{  
  "data": [  
    {  
      "key": "key1",  
      "value": "value"  
    },  
    {  
      "key": "key2",  
      "value": "value"  
    }  
  ]  
}
```

Seu arquivo não pode incluir chaves duplicadas. Se você especificou um arquivo inválido em seu bucket do Amazon S3, é possível atualizar o arquivo para remover quaisquer duplicatas e, em seguida, tentar criar seu armazenamento de chave-valor novamente.

Para ter mais informações, consulte [Criar um armazenamento de chave-valor](#).

Note

O arquivo da fonte de dados e os pares de chave-valor têm os seguintes limites:

- Tamanho do arquivo: 5 MB
- Tamanho da chave: 512 caracteres
- Tamanho da chave: 1.024 caracteres

Trabalhar com dados de chave-valor

Este tópico descreve como adicionar pares de chave-valor a um armazenamento de chave-valor existente. Para incluir pares de chave-valor ao criar inicialmente armazenamentos de chave-valor, consulte [the section called “Criar um armazenamento de chave-valor”](#).

Tópicos

- [Trabalhar com pares de chave-valor \(console\)](#)
- [Sobre o KeyValueStore do CloudFront](#)
- [Trabalhar com pares de chave-valor \(AWS CLI\)](#)
- [Trabalhar com pares de chave-valor \(API\)](#)

Trabalhar com pares de chave-valor (console)

É possível usar o console do CloudFront para trabalhar com os seus pares de chave-valor.

Como trabalhar com pares de chave-valor

1. Faça login no AWS Management Console e abra a página Functions (Funções) no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home#/functions>.
2. Selecione a guia KeyValueStores.
3. Selecione o armazenamento de chave-valor que deseja alterar.
4. Na seção Pares de chave-valor, selecione Editar.
5. É possível adicionar um par de chave-valor, excluir um par de chave-valor ou alterar o valor de um par existente.

6. Ao concluir, escolha Salvar alterações.

Sobre o KeyValueCollection do CloudFront

Tip

A API KeyValueCollection do CloudFront é um serviço global que usa o Signature Version 4A (SigV4A) para autenticação. O uso de credenciais temporárias com o SigV4A requer tokens de sessão da versão 2. Para ter mais informações, consulte [Usar credenciais temporárias com a API KeyValueCollection do CloudFront](#).

Se você estiver usando a AWS Command Line Interface (AWS CLI) ou seu próprio código para chamar a API KeyValueCollection do CloudFront, consulte as seções a seguir.

Quando você trabalha com um armazenamento de chave-valor e seus pares de chave-valor, o serviço que você chama depende do seu caso de uso:

- Para trabalhar com pares de chave-valor em um armazenamento de chave-valor existente, use o serviço KeyValueCollection do CloudFront.
- Para incluir pares de chave-valor no armazenamento de chave-valor ao criá-lo inicialmente, use o serviço CloudFront.

Tanto a API do CloudFront quanto a API KeyValueCollection do CloudFront têm uma operação `DescribeKeyValueCollection`. Você pode chamá-los por diferentes motivos. Para entender as diferenças, consulte a tabela a seguir.

	API <code>DescribeKeyValueCollection</code> do CloudFront	API KeyValueCollection <code>DescribeKeyValueCollection</code> do CloudFront
Dados sobre o armazenamento de chave-valor	Retorna dados, como o status e a data em que o próprio armazenamento de chave-valor foi modificado pela última vez.	Retorna dados sobre o conteúdo do recurso de armazenamento: os pares de chave-valor no armazenamento e o tamanho do conteúdo.

	API DescribeKeyValueStore do CloudFront	API KeyValueStore DescribeKeyValueStore do CloudFront
Dados que identificam o armazenamento de chave-valor	Exibe uma ETag, o UUID e o ARN do armazenamento de chave-valor.	Exibe uma ETag e o ARN do armazenamento de chave-valor.

Observações

- Cada operação DescribeKeyValueStore exibe uma ETag diferente. As ETags não são intercambiáveis.
- Ao chamar uma operação de API para concluir uma ação, é necessário especificar a ETag da API apropriada. Por exemplo, na operação [DeleteKey](#) do KeyValueStore do CloudFront, é possível especificar a ETag exibida da operação [DescribeKeyValueStore](#) do KeyValueStore do CloudFront.

Trabalhar com pares de chave-valor (AWS CLI)

É possível executar os comandos da AWS Command Line Interface a seguir no KeyValueStore do CloudFront.

Sumário

- [Listar pares de chave-valor](#)
- [Obter pares de chave-valor](#)
- [Descrever um armazenamento de chave-valor](#)
- [Criar um par de chave-valor](#)
- [Excluir um par de chave-valor](#)
- [Atualizar pares de chave-valor](#)

Listar pares de chave-valor

Para listar pares de chave-valor no armazenamento de chave-valor, execute o comando a seguir.

```
aws cloudfront-keyvaluestore list-keys \  
  --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-  
example
```

Resposta

```
{  
  "Items": [  
    {  
      "Key": "key1",  
      "Value": "value1"  
    }  
  ]  
}
```

Obter pares de chave-valor

Para obter um par de chave-valor no armazenamento de chave-valor, execute o comando a seguir.

```
aws cloudfront-keyvaluestore get-key \  
  --key=key1 \  
  --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-  
example
```

Resposta

```
{  
  "Key": "key1",  
  "Value": "value1",  
  "ItemCount": 1,  
  "TotalSizeInBytes": 11  
}
```

Descrever um armazenamento de chave-valor

Para descrever um armazenamento de chave-valor, execute o comando a seguir.

```
aws cloudfront-keyvaluestore describe-key-value-store \  
  --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-  
example
```

Resposta

```
{
  "ETag": "KV1F83G8C2AR07P",
  "ItemCount": 1,
  "TotalSizeInBytes": 11,
  "KvsARN": "arn:aws:cloudfront::123456789012:key-value-store/37435e19-
c205-4271-9e5c-example",
  "Created": "2024-05-08T07:48:45.381000-07:00",
  "LastModified": "2024-08-05T13:50:58.843000-07:00",
  "Status": "READY"
}
```

Criar um par de chave-valor

Para criar um par de chave-valor no armazenamento de chave-valor, execute o comando a seguir.

```
aws cloudfront-keyvaluestore put-key \
  --if-match=KV1PA6795UKMFR9 \
  --key=key2 \
  --value=value2 \
  --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-
example
```

Resposta

```
{
  "ETag": "KV13V1IB3VIYZZH",
  "ItemCount": 3,
  "TotalSizeInBytes": 31
}
```

Excluir um par de chave-valor

Para excluir um par de chave-valor, execute o comando a seguir.

```
aws cloudfront-keyvaluestore delete-key \
  --if-match=KV13V1IB3VIYZZH \
  --key=key1 \
  --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-
example
```


Saída

```
{
  "ETag": "KV1VC38T7YXB528",
  "ItemCount": 2,
  "TotalSizeInBytes": 22
}
```

Atualizar pares de chave-valor

É possível usar o comando `update-keys` para atualizar mais de um par de chave-valor. Por exemplo, para excluir um par de chave-valor e criar outro, execute o comando a seguir.

```
aws cloudfront-keyvaluestore update-keys \
  --if-match=KV2EUQ1WTGCTBG2 \
  --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-
example \
  --deletes '[{"Key":"key2"}]' \
  --puts '[{"Key":"key3","Value":"value3"}]'
```

Resposta

```
{
  "ETag": "KV3AEGXETSR30VB",
  "ItemCount": 3,
  "TotalSizeInBytes": 28
}
```

Trabalhar com pares de chave-valor (API)

Siga esta seção para trabalhar com os pares de chave-valor de forma programática.

Sumário

- [Obter uma referência a um armazenamento de chave-valor](#)
- [Alterar pares de chave-valor em um armazenamento de chave-valor](#)
- [Exemplo de código de KeyValueStore do CloudFront](#)

Obter uma referência a um armazenamento de chave-valor

Ao usar a API `KeyValueStore` do CloudFront para chamar uma operação de gravação, é necessário especificar o ARN e a ETag do armazenamento de chave-valor. Para obter esses dados, faça o seguinte:

Como obter uma referência a um armazenamento de chave-valor

1. Use a operação de API [CloudFront ListKeyValueStores](#) para obter uma lista de armazenamentos de chave-valor. Encontre o armazenamento de chave-valor que deseja alterar.
2. Use a [operação de API CloudFrontKeyValueStore DescribeKeyValueStore](#) e especifique o armazenamento de chave-valor da etapa anterior.

A resposta inclui o ARN e a ETag do armazenamento de chave-valor.

- O ARN inclui o número da Conta da AWS, o `key-value-store` constante e o UUID, como o seguinte exemplo:

```
arn:aws:cloudfront::123456789012:key-value-store/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

- Uma ETag que se parece com o seguinte exemplo:

```
ETVABCEXAMPLE2
```

Alterar pares de chave-valor em um armazenamento de chave-valor

É possível especificar o armazenamento de chave-valor que contém o par de chave-valor que você deseja atualizar.

Veja as seguintes operações da API `KeyValueStore` do CloudFront:

- [CloudFrontKeyValueStore DeleteKey](#): exclui um par de chave-valor
- [CloudFrontKeyValueStore GetKey](#): retorna um par de chave-valor
- [CloudFrontKeyValueStore ListKeys](#): retorna uma lista de pares de chave-valor
- [CloudFrontKeyValueStore PutKey](#): é possível realizar as seguintes tarefas:
 - Criar um par de chave-valor em um armazenamento de chave-valor especificando um novo nome e valor da chave.

- Defina um valor diferente em um par de chave-valor especificando um nome de chave existente e um novo valor de chave.
- [CloudFrontKeyJsonValueStore UpdateKeys](#): é possível executar uma ou mais das seguintes ações em uma operação de tudo ou nada:
 - Excluir um ou mais pares de chave-valor.
 - Criar um ou mais pares de chave-valor.
 - Definir um valor diferente em um ou mais pares de chave-valor existentes.

Exemplo de código de KeyValueStore do CloudFront

Example

O código a seguir mostra como chamar a operação de API `DescribeKeyValueStore` para um armazenamento de chave-valor.

```
const {
  CloudFrontKeyJsonValueStoreClient,
  DescribeKeyValueStoreCommand,
} = require("@aws-sdk/client-cloudfront-keyvaluestore");

require("@aws-sdk/signature-v4-crt");

(async () => {
  try {
    const client = new CloudFrontKeyJsonValueStoreClient({
      region: "us-east-1"
    });
    const input = {
      KvsARN: "arn:aws:cloudfront::123456789012:key-value-store/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
    };
    const command = new DescribeKeyValueStoreCommand(input);

    const response = await client.send(command);
  } catch (e) {
    console.log(e);
  }
})();
```

Personalização na borda com o Lambda@Edge

O Lambda@Edge é uma extensão do AWS Lambda. O Lambda@Edge é um serviço de computação que permite executar funções que personalizam o conteúdo fornecido pelo Amazon CloudFront. É possível criar funções do Node.js ou Python no console do Lambda em uma Região da AWS, Leste dos EUA (Norte da Virgínia).

Depois de criar a função, é possível adicionar acionadores ao console do Lambda ou do CloudFront para que as funções sejam executadas nos locais da AWS mais próximos do visualizador, sem provisionar ou gerenciar servidores. Se desejar, você poderá usar as operações de API do Lambda e do CloudFront para configurar as funções e os acionadores de forma programática.

O Lambda@Edge é automaticamente dimensionado, desde algumas solicitações por dia até milhares por segundo. O processamento de solicitações em locais da AWS mais próximos do visualizador, em vez de servidores de origem, reduz significativamente a latência e melhora a experiência do usuário.

Tópicos

- [Como o Lambda@Edge funciona com solicitações e respostas](#)
- [Maneiras de usar o Lambda@Edge](#)
- [Começar a usar funções do Lambda@Edge](#)
- [Configurar permissões e perfis do IAM para o Lambda@Edge](#)
- [Escrever e criar uma função do Lambda@Edge](#)
- [Adicionar acionadores para uma função do Lambda@Edge](#)
- [Testar e depurar as funções do Lambda@Edge](#)
- [Excluir funções e réplicas do Lambda@Edge](#)
- [Estrutura de eventos do Lambda@Edge](#)
- [Trabalho com solicitações e respostas](#)
- [Funções de exemplo do Lambda@Edge](#)

Como o Lambda@Edge funciona com solicitações e respostas

Ao associar uma distribuição do CloudFront a uma função do Lambda@Edge, o CloudFront intercepta solicitações e respostas nos pontos de presença do CloudFront. É possível executar funções do Lambda quando ocorrem os seguintes eventos do CloudFront:

- Quando o CloudFront receber uma solicitação de um visualizador (solicitação do visualizador)
- Antes do CloudFront encaminhar uma solicitação para a origem (solicitação da origem)
- Quando o CloudFront receber uma resposta da origem (resposta da origem)
- Antes do CloudFront retornar a resposta para o visualizador (resposta do visualizador)

Se você estiver usando o AWS WAF, a solicitação do visualizador do Lambda@Edge será executada após a aplicação de qualquer regra do AWS WAF.

Para ter mais informações, consulte [Trabalho com solicitações e respostas](#) e [Estrutura de eventos do Lambda@Edge](#).

Maneiras de usar o Lambda@Edge

Há diversos usos para o processamento do Lambda@Edge com sua distribuição do Amazon CloudFront, como os seguintes exemplos:

- Uma função do Lambda pode inspecionar cookies e reescrever URLs para que os usuários vejam diferentes versões de um site para testes A/B.
- O CloudFront pode retornar objetos diferentes aos visualizadores dependendo do dispositivo que estão usando, verificando o cabeçalho `User-Agent`, que inclui informações sobre os dispositivos. Por exemplo, o CloudFront pode retornar imagens diferentes com base no tamanho da tela do seu dispositivo. Da mesma forma, a função pode considerar o valor do cabeçalho `Referer` e fazer com que o CloudFront retorne imagens com a menor resolução disponível a bots.
- Ou você pode verificar cookies para outros critérios. Por exemplo, em um site de varejo que vende roupas, se você usar cookies para indicar a cor de uma jaqueta escolhida por um usuário, a função do Lambda poderá alterar a solicitação para que o CloudFront retorne a imagem de uma jaqueta na cor selecionada.
- Uma função do Lambda pode gerar respostas HTTP quando ocorrerem os eventos da solicitação de origem ou do visualizador do CloudFront.
- Uma função pode inspecionar cabeçalhos ou tokens de autorização e inserir um cabeçalho para controlar o acesso ao seu conteúdo antes de o CloudFront encaminhar uma solicitação para a origem.
- Uma função do Lambda também pode fazer chamadas de rede para recursos externos a fim de confirmar as credenciais do usuário ou obter conteúdo adicional para personalizar uma resposta.

Consulte mais informações, incluindo exemplos de código em [Funções de exemplo do Lambda@Edge](#).

Consulte mais informações sobre como configurar o Lambda@Edge no console em [Tutorial: criar uma função básica do Lambda@Edge](#).

Começar a usar funções do Lambda@Edge

Com o Lambda@Edge, é possível usar acionadores do CloudFront para invocar uma função do Lambda. Quando você associa uma distribuição do CloudFront a uma função do Lambda, o CloudFront [intercepta solicitações e respostas](#) nos pontos de presença do CloudFront e executa a função. As funções do Lambda podem melhorar a segurança ou personalizar informações próximas aos visualizadores para melhorar a performance.

A lista a seguir fornece uma visão geral básica de como criar e usar as funções do Lambda com o CloudFront.

Visão geral: como criar e usar funções do Lambda com o CloudFront

1. Crie uma função do Lambda na região Leste dos EUA (N. da Virgínia).
2. Salve e publique uma versão numerada da função.

Para alterar a função, edite a versão \$LATEST da função na região Leste dos EUA (Norte da Virgínia). A seguir, antes de configurá-la para funcionar com o CloudFront, publique uma nova versão numerada.

3. Associe a função a uma distribuição do CloudFront e ao comportamento de cache. Depois, especifique um ou mais eventos do CloudFront (triggers) que fazem com que a função seja executada. Por exemplo, você pode criar um trigger para que a função seja executada quando o CloudFront receber uma solicitação de um visualizador.
4. Quando você cria um acionador, o Lambda cria réplicas da função em locais da AWS em todo o mundo.

Tip

Consulte mais informações em [criar e atualizar funções](#), [a estrutura de eventos](#) e [adicionar acionadores do CloudFront](#). Você também pode encontrar mais ideias e obter amostras de código em [Funções de exemplo do Lambda@Edge](#).

Consulte um tutorial detalhado no seguinte tópico:

Tópicos

- [Tutorial: criar uma função básica do Lambda@Edge](#)

Tutorial: criar uma função básica do Lambda@Edge

Este tutorial mostra como começar a usar o Lambda@Edge criando e configurando um exemplo de função do Node.js que é executado no CloudFront. Este exemplo adiciona cabeçalhos de segurança HTTP a uma resposta quando o CloudFront recupera um arquivo. (Isso pode melhorar a segurança e a privacidade de um site.)

Você não precisa de seu próprio site para esse tutorial. No entanto, ao optar por criar sua própria solução do Lambda@Edge, você segue etapas semelhantes e escolhe entre as mesmas opções.

Tópicos

- [Etapa 1: cadastrar-se com uma Conta da AWS](#)
- [Etapa 2: Criar uma distribuição do CloudFront](#)
- [Etapa 3: criar sua função](#)
- [Etapa 4: adicionar um acionador do CloudFront para executar a função](#)
- [Etapa 5: verificar se a função é executada](#)
- [Etapa 6: solução de problemas](#)
- [Etapa 7: apagar recursos de exemplo](#)
- [Informações relacionadas](#)

Etapa 1: cadastrar-se com uma Conta da AWS

Se ainda não tiver feito isso, cadastre-se para uma Conta da AWS. Para ter mais informações, consulte [Cadastrar-se em uma Conta da AWS](#).

Etapa 2: Criar uma distribuição do CloudFront

Antes de criar o exemplo de função do Lambda@Edge, você deve ter um ambiente do CloudFront para trabalhar com o que inclui uma origem para servir conteúdo.

Para este exemplo, crie uma distribuição do CloudFront que use um bucket do Amazon S3 como a origem da distribuição. Se já tiver um ambiente para usar, você pode ignorar esta etapa.

Como criar uma distribuição do CloudFront com uma origem do Amazon S3

1. Crie um bucket do Amazon S3 com um ou dois arquivos, como arquivos de imagem, para exemplo de conteúdo. Para obter ajuda, siga as etapas em [Fazer upload do conteúdo no Amazon S3](#). Certifique-se de ter definido permissões para conceder acesso público de leitura aos objetos do seu bucket.
2. Crie uma distribuição do CloudFront e adicione o bucket do S3 como uma origem, seguindo as etapas em [Criar uma distribuição na Web do CloudFront](#). Se você já tiver uma distribuição, pode adicionar o bucket como origem para essa distribuição.

Tip

Anote seu ID de distribuição. Posteriormente neste tutorial, ao adicionar um acionador do CloudFront para a função, você deve escolher o ID da distribuição em uma lista suspensa; por exemplo, E653W22221KDDL.

Etapa 3: criar sua função

Nesta etapa, você cria uma função do Lambda, começando com um modelo de esquema no console do Lambda. A função adiciona o código para atualizar os cabeçalhos de segurança na distribuição do CloudFront.

Criar uma função do Lambda

1. Faça login no AWS Management Console e abra o console do AWS Lambda em <https://console.aws.amazon.com/lambda/>.

Important

Verifique se você está na Região da AWS US-East-1 (Norte da Virgínia) (us-east-1). Você deve estar nessa região para criar as funções do Lambda@Edge.

2. Escolha a opção Criar função.
3. Na página Create function (Criar função), escolha Use a blueprint (Usar um esquema) e filtre os esquemas do CloudFront inserindo **cloudfront** no campo de pesquisa.

Note

Os esquemas do CloudFront estão disponíveis somente na região US-East-1 (N. Virgínia) (Leste dos EUA-1 (Norte da Virgínia)) (us-east-1).

4. Escolha o esquema Modify HTTP response header (Modificar cabeçalho de resposta HTTP) como modelo para sua função.
5. Insira as seguintes informações sobre sua função:
 - Nome da função: insira um nome para a função.
 - Perfil de execução: escolha como definir as permissões para a função. Para usar o modelo básico de política de permissões recomendado do Lambda@Edge, selecione Criar uma função de modelos de política da AWS.
 - Nome do perfil: insira um nome para o perfil que o modelo de política cria.
 - Modelos de política: o Lambda adiciona automaticamente o modelo de política Permissões básicas do Lambda@Edge porque você escolheu um esquema do CloudFront como a base da função. Esse modelo de política adiciona permissões de função de execução que permitem que o CloudFront execute sua função do Lambda para você em locais do CloudFront em todo o mundo. Para ter mais informações, consulte [Configurar permissões e perfis do IAM para o Lambda@Edge](#).
6. Selecione Criar função na parte inferior da página.
7. No painel Implantar no Lambda@Edge exibido, escolha Cancelar. (Para este tutorial, você deve modificar o código da função antes de implantá-la no Lambda@Edge.)
8. Role para baixo até a seção Code source (Origem do código) da página.
9. Substitua o código do modelo por uma função que modifica os cabeçalhos de segurança que sua origem retorna. Por exemplo, você pode usar um código semelhante ao seguinte:

```
'use strict';
export const handler = (event, context, callback) => {

    //Get contents of response
    const response = event.Records[0].cf.response;
    const headers = response.headers;

    //Set new headers
```

```
headers['strict-transport-security'] = [{key: 'Strict-Transport-Security',
value: 'max-age= 63072000; includeSubdomains; preload'}];
headers['content-security-policy'] = [{key: 'Content-Security-Policy', value:
"default-src 'none'; img-src 'self'; script-src 'self'; style-src 'self'; object-
src 'none'"}];
headers['x-content-type-options'] = [{key: 'X-Content-Type-Options', value:
'nosniff'}];
headers['x-frame-options'] = [{key: 'X-Frame-Options', value: 'DENY'}];
headers['x-xss-protection'] = [{key: 'X-XSS-Protection', value: '1;
mode=block'}];
headers['referrer-policy'] = [{key: 'Referrer-Policy', value: 'same-origin'}];

//Return modified response
callback(null, response);
};
```

10. Selecione File (Arquivo), Save (Salvar) para salvar o código atualizado.

11. Escolha Implantar.

Prossiga para a próxima seção para adicionar um trigger do CloudFront para executar a função.

Etapa 4: adicionar um acionador do CloudFront para executar a função

Agora que você tem uma função do Lambda para atualizar os cabeçalhos de segurança, configure o trigger do CloudFront que executa a função para adicionar cabeçalhos em qualquer resposta recebida pelo CloudFront da origem para sua distribuição.

Como configurar o trigger do CloudFront para sua função

1. No console do Lambda, na página de Function overview (Visão geral da função) da função, escolha Add trigger (Adicionar acionador).
2. Em Trigger configuration (Configuração do acionador), escolha CloudFront.
3. Escolha Implantar no Lambda@Edge.
4. No painel Implantar no Lambda@Edge, em Configurar acionador do CloudFront, insira as seguintes informações:
 - Distribuição: o ID de distribuição do CloudFront a ser associado à função. Na lista suspensa, escolha o ID da distribuição.
 - Comportamento de cache: o comportamento de cache para usar com o acionador. Para este exemplo, deixe o valor definido como *, que indica o comportamento de cache padrão da

distribuição. Para obter mais informações, consulte [Configurações de comportamento de cache](#)[Referência de configurações da distribuição](#) no tópico.

- Evento do CloudFront: o acionador que especifica quando a função é executada. Queremos que os cabeçalhos de segurança da função sejam executados sempre que o CloudFront retornar uma resposta da origem. Na lista suspensa, escolha Resposta da origem. Para ter mais informações, consulte [Adicionar acionadores para uma função do Lambda@Edge](#).
5. Marque a caixa de seleção Confirmar implantação no Lambda@Edge.
 6. Escolha Deploy (Implantar) para adicionar o trigger e replicar a função para os locais da AWS em todo o mundo.
 7. Aguarde até que a função seja replicada. Isso normalmente demora vários minutos.

Você pode verificar se a replicação foi concluída [acessando o console do CloudFront](#) e visualizando a distribuição. Aguarde o status de distribuição mudar de Implantando para uma data e hora, o que significa que sua função foi replicada. Para verificar se a função funciona, siga as etapas na próxima seção.

Etapa 5: verificar se a função é executada

Agora que você criou a função do Lambda e configurou um trigger para executá-la em uma distribuição do CloudFront, verifique se a função está realizando o que você espera. Neste exemplo, verificamos os cabeçalhos HTTP que o CloudFront retorna, para garantir que os cabeçalhos de segurança estejam adicionados.

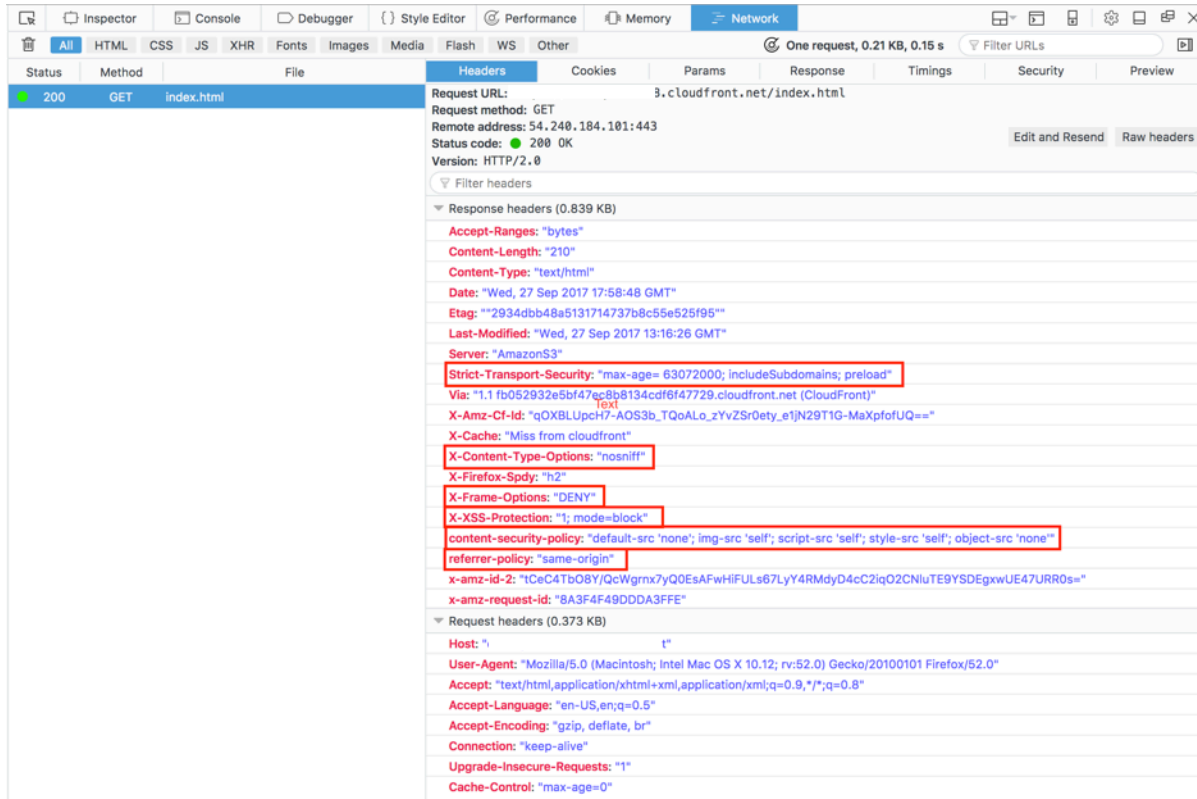
Para verificar se a sua função do Lambda@Edge adiciona cabeçalhos de segurança

1. Em um navegador, insira o URL para um arquivo no seu bucket do S3. Por exemplo, você pode usar um URL semelhante a `https://d111111abcdef8.cloudfront.net/image.jpg`.

Para obter mais informações sobre o nome de domínio do CloudFront a ser usado no arquivo URL, consulte [Personalizar o formato do URL para arquivos no CloudFront](#).

2. Abra a barra de ferramentas do desenvolvedor da Web do navegador. Por exemplo, em sua janela do navegador Chrome, abra o menu de contexto (clique com o botão direito do mouse) e escolha Inspect (Inspeccionar).
3. Escolha a guia Network.
4. Recarregue a página para visualizar sua imagem e, em seguida, escolha uma solicitação HTTP no painel esquerdo. Você vê os cabeçalhos HTTP exibidos em um painel separado.

5. Verifique a lista de cabeçalhos HTTP para verificar se os cabeçalhos de segurança esperados são incluídos na lista. Por exemplo, você poderá ver cabeçalhos semelhantes aos mostrados na captura de tela a seguir:



Se os cabeçalhos de segurança forem incluídos na sua lista de cabeçalhos, excelente! Você criou com êxito sua primeira função do Lambda@Edge. Se o CloudFront retornar erros ou se houver outros problemas, continue na próxima etapa para solucionar os problemas.

Etapa 6: solução de problemas

Se o CloudFront retornar erros ou não adicionar os cabeçalhos de segurança conforme esperado, você poderá investigar a execução da função verificando o CloudWatch Logs. Certifique-se de usar os logs armazenados no local da AWS que estão mais próximos do local em que a função é executada.

Por exemplo, se você visualizar o arquivo de Londres, tente alterar a região no console do CloudWatch para EU (Londres).

Como examinar os CloudWatch Logs para sua função do Lambda@Edge

1. Faça login no AWS Management Console e abra o console do CloudWatch em <https://console.aws.amazon.com/cloudwatch/>.
2. Altere a Region (Região) para o local que é mostrado quando você visualiza o arquivo no navegador. Aqui é onde a função está sendo executada.
3. No painel esquerdo, selecione Logs para visualizar os logs da sua distribuição.

Para ter mais informações, consulte [Monitorar métricas do CloudFront com o Amazon CloudWatch](#).

Etapa 7: apagar recursos de exemplo

Se você criar um bucket do Amazon S3 e a distribuição do CloudFront apenas para este tutorial, exclua os recursos da AWS alocados para não acumular mais cobranças. Depois que você excluir os seus recursos da AWS, qualquer conteúdo que você adicionou não ficará mais disponível.

Tarefas

- [Exclua o bucket do S3](#)
- [Excluir a função Lambda](#)
- [Exclua a distribuição do CloudFront](#)

Exclua o bucket do S3

Antes de excluir o bucket do Amazon S3, verifique se o log está desativado para o bucket. Caso contrário, a AWS continuará gravando logs para o bucket à medida que você o excluir.

Para desabilitar o registro em log para um bucket

1. Abra o console do Amazon S3 em <https://console.aws.amazon.com/s3/>.
2. Selecione o bucket e escolha Properties (Propriedades).
3. Em Properties (Propriedades), escolha Logging (Registro).
4. Desmarque a caixa de seleção Enabled (Habilitado).
5. Escolha Save (Salvar).

Agora, você pode excluir seu bucket. Para obter mais informações, consulte [Exclusão de um bucket](#) no Guia do usuário do console do Amazon Simple Storage Service.

Excluir a função Lambda

Para ter instruções sobre como excluir a associação da função do Lambda e, opcionalmente, a própria função, consulte [Excluir funções e réplicas do Lambda@Edge](#).

Exclua a distribuição do CloudFront

Antes de excluir uma distribuição do CloudFront, você deve desabilitá-la. Uma distribuição desabilitada deixa de ser funcional e não acumula encargos. É possível habilitar uma distribuição desabilitada a qualquer momento. Depois que você excluir uma distribuição desabilitada, ela deixará de estar disponível.

Como desabilitar e excluir uma distribuição do CloudFront

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Selecione a distribuição que você deseja desabilitar e escolha Disable (Desabilitar).
3. Quando a confirmação for solicitada, escolha Yes, Disable (Sim, desabilitar).
4. Selecione a distribuição desabilitada e escolha Delete (Excluir).
5. Quando a confirmação for solicitada, escolha Yes, Delete (Sim, excluir).

Informações relacionadas

Agora que você tem uma ideia básica de como as funções do Lambda@Edge funcionam, saiba mais lendo o seguinte:

- [Funções de exemplo do Lambda@Edge](#)
- [Práticas recomendadas de design do Lambda@Edge](#)
- [Reducing Latency and Shifting Compute to the Edge with Lambda@Edge](#)

Configurar permissões e perfis do IAM para o Lambda@Edge

Para configurar o Lambda@Edge, é necessário ter as seguintes permissões e perfis do IAM para o AWS Lambda:

- [Permissões do IAM](#): essas permissões autorizam você a criar a função do Lambda e a associá-la à distribuição do CloudFront.
- [Perfil de execução da função do Lambda](#) (perfil do IAM): as entidades principais do serviço do Lambda assumem esse perfil para executar a função.

- [Perfis vinculados ao serviço para o Lambda@Edge](#): os perfis vinculados ao serviço permitem que Serviços da AWS específicos repliquem funções do Lambda para Regiões da AWS e habilitem o CloudWatch para usar arquivos de log do CloudFront.

Permissões do IAM necessárias para associar funções do Lambda às distribuições do CloudFront

Além das permissões do IAM necessárias para o Lambda, você precisa das seguintes permissões para associar as funções do Lambda às distribuições do CloudFront:

- `lambda:GetFunction`: concede permissão para recebimento de informações de configuração para a função do Lambda e de um URL pré-assinado para baixar um arquivo `.zip` que contém a função.
- `lambda:EnableReplication*`: concede permissão à política de recurso para que o serviço de replicação do Lambda possa obter a configuração e o código da função.
- `lambda:DisableReplication*`: concede permissão à política de recurso para que o serviço de replicação do Lambda possa excluir a função.

Important

É necessário adicionar o asterisco (*) ao final das ações `lambda:EnableReplication*` e `lambda:DisableReplication*`.

- Para o recurso, especifique o ARN da versão da função que você deseja executar quando ocorrer um evento do CloudFront, conforme mostrado no seguinte exemplo:

```
arn:aws:lambda:us-east-1:123456789012:function:TestFunction:2
```
- `iam:CreateServiceLinkedRole`: concede permissão para criar um perfil vinculado ao serviço que o Lambda@Edge usa para replicar funções do Lambda no CloudFront. Depois que você configurar o Lambda@Edge pela primeira vez, o perfil vinculado ao serviço será criado automaticamente para você. Não é necessário adicionar essa permissão a outras distribuições que usam o Lambda@Edge.
- `cloudfront:UpdateDistribution` ou `cloudfront:CreateDistribution`: concede permissão para atualizar ou criar uma distribuição.

Para obter mais informações, consulte os tópicos a seguir.

- [Identity and Access Management para Amazon CloudFront](#)
- [Permissões de acesso a recursos do Lambda](#) no Guia do desenvolvedor do AWS Lambda

Função de execução de função para primários de serviço

É necessário criar um perfil do IAM que as entidades principais do serviço `lambda.amazonaws.com` e `edgelambda.amazonaws.com` possam assumir ao executarem a função.

Tip

Ao criar a função no console do Lambda, você pode optar por criar um perfil de execução usando um modelo de política da AWS. Essa etapa adiciona automaticamente as permissões necessárias do `Lambda@Edge` para executar a função. Consulte a [Etapa 5 do tutorial Criação de uma função do Lambda@Edge simples](#).

Consulte mais informações sobre como criar um perfil do IAM manualmente em [Criar funções e anexar políticas \(console\)](#) no Guia do usuário do IAM.

Example Exemplo: política de confiança do perfil

É possível adicionar esse perfil na guia Relação de confiança no console do IAM. Não adicione essa política na guia Permissões.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "lambda.amazonaws.com",
          "edgelambda.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```


Consulte mais informações sobre as permissões que você precisa conceder ao perfil de execução em [Permissões de acesso a recursos do Lambda](#) no Guia do desenvolvedor do AWS Lambda.

Observações

- Por padrão, sempre que um evento do CloudFront aciona uma função do Lambda, os dados são gravados no CloudWatch Logs. Se você quiser usar esses logs, a função de execução precisará de permissão para registrar dados no CloudWatch Logs. É possível usar o `AWSLambdaBasicExecutionRole` predefinido para conceder permissão ao perfil de execução.

Para obter mais informações sobre o CloudWatch Logs, consulte [the section called “Logs de funções de borda”](#).

- Se o código da função do Lambda acessar outros recursos da AWS, como a leitura de um objeto em um bucket do S3, o perfil de execução precisará de permissão para executar essa ação.

Funções vinculadas ao serviço para o Lambda@Edge

O Lambda@Edge usa [perfis vinculados ao serviço](#) do IAM. Uma função vinculada ao serviço é um tipo exclusivo de função do IAM vinculada diretamente a um serviço. As funções vinculadas a serviços são predefinidas pelo serviço e incluem todas as permissões de que ele precisa para chamar outros serviços da AWS em seu nome.

O Lambda@Edge usa os seguintes perfis vinculados ao serviço do IAM:

- `AWSServiceRoleForLambdaReplicator`: o Lambda@Edge usa essa função para permitir que ele mesmo replique funções para Regiões da AWS.

Quando você adiciona um acionador do Lambda@Edge ao CloudFront pela primeira vez, um perfil chamado `AWSServiceRoleForLambdaReplicator` é criado automaticamente para permitir que o Lambda@Edge replique funções para Regiões da AWS. Esse perfil é necessário para usar as funções do Lambda@Edge. O ARN do perfil `AWSServiceRoleForLambdaReplicator` é semelhante a este exemplo:

```
arn:aws:iam::123456789012:role/aws-service-role/  
replicator.lambda.amazonaws.com/AWSServiceRoleForLambdaReplicator
```

- `AWSServiceRoleForCloudFrontLogger`: o CloudFront usa esse perfil para enviar arquivos de log ao CloudWatch. É possível usar arquivos de log para depurar erros de validação do Lambda@Edge.

O perfil `AWSServiceRoleForCloudFrontLogger` é criado automaticamente quando você adiciona a associação da função do Lambda@Edge para permitir que o CloudFront envie arquivos de log de erros do Lambda@Edge ao CloudWatch. O ARN para a função `AWSServiceRoleForCloudFrontLogger` é semelhante a:

```
arn:aws:iam::account_number:role/aws-service-role/  
logger.cloudfront.amazonaws.com/AWSServiceRoleForCloudFrontLogger
```

Uma função vinculada a serviço facilita a configuração e o uso do Lambda@Edge, pois você não precisa adicionar manualmente as permissões necessárias. Lambda@Edge define as permissões de suas funções vinculadas ao serviço e apenas Lambda@Edge pode assumir as funções. As permissões definidas incluem a política de confiança e a política de permissões. Não é possível anexar a política de permissões a nenhuma outra entidade do IAM.

Você deve remover todos os recursos do CloudFront ou do Lambda@Edge associados para poder excluir a função vinculada ao serviço. Isso ajuda a proteger seus recursos do Lambda@Edge de modo que você não remova um perfil vinculado ao serviço que ainda seja necessário para acessar os recursos ativos.

Para obter mais informações sobre funções vinculadas ao serviço, consulte [Funções vinculadas ao serviço do CloudFront](#).

Permissões de função vinculada ao serviço para o Lambda@Edge

O Lambda@Edge usa duas funções vinculadas a serviços: `AWSServiceRoleForLambdaReplicator` e `AWSServiceRoleForCloudFrontLogger`. As seções a seguir descrevem as permissões para cada uma dessas funções.

Sumário

- [Permissões de função vinculada ao serviço para o replicador do Lambda](#)
- [Permissões de função vinculada ao serviço para o CloudFront Logger](#)

Permissões de função vinculada ao serviço para o replicador do Lambda

Essa função vinculada ao serviço permite que o Lambda replique funções do Lambda@Edge para Regiões da AWS.

A função vinculada ao serviço `AWSServiceRoleForLambdaReplicator` confia no serviço `replicator.lambda.amazonaws.com` para presumir a função.

A política de permissões da função permite que o `Lambda@Edge` conclua as seguintes ações nos recursos especificados:

- `lambda:CreateFunction` no `arn:aws:lambda:*:*:function:*`
- `lambda>DeleteFunction` no `arn:aws:lambda:*:*:function:*`
- `lambda:DisableReplication` no `arn:aws:lambda:*:*:function:*`
- `iam:PassRole` no `all AWS resources`
- `cloudfront:ListDistributionsByLambdaFunction` no `all AWS resources`

Permissões de função vinculada ao serviço para o CloudFront Logger

Esse perfil vinculado ao serviço permite que o CloudFront envie arquivos de log por push ao CloudWatch para que seja possível depurar erros de validação do `Lambda@Edge`.

A função vinculada ao serviço `AWSServiceRoleForCloudFrontLogger` confia no serviço `logger.cloudfront.amazonaws.com` para presumir a função.

A política de permissões do perfil permite que o `Lambda@Edge` conclua as seguintes ações no recurso `arn:aws:logs:*:*:log-group:/aws/cloudfront/*` especificado:

- `logs:CreateLogGroup`
- `logs:CreateLogStream`
- `logs:PutLogEvents`

Você deve configurar permissões para permitir que uma entidade do IAM (como um usuário, grupo ou função) exclua uma função vinculada ao serviço do `Lambda@Edge`. Para ter mais informações, consulte [Service-linked role permissions](#) (Permissões de nível vinculado a serviços) no Guia do usuário do IAM.

Criação de funções vinculadas ao serviço para o `Lambda@Edge`

Normalmente, não é necessário criar manualmente as funções vinculadas a serviços para o `Lambda@Edge`. O serviço cria as funções automaticamente nas seguintes situações:

- Quando você cria um acionador pela primeira vez, o serviço cria o perfil `AWSServiceRoleForLambdaReplicator` (caso ele ainda não exista). Esse perfil permite que o Lambda replique funções do Lambda@Edge para Regiões da AWS.

Se você excluir a função vinculada ao serviço, a função será criada novamente quando você adicionar um novo gatilho para o Lambda@Edge em uma distribuição.

- Quando você atualiza ou cria uma distribuição do CloudFront que tem uma associação ao Lambda@Edge, o serviço cria o perfil `AWSServiceRoleForCloudFrontLogger` (caso o perfil ainda não exista). Esse perfil permite que o CloudFront envie arquivos de log ao CloudWatch.

Se você excluir a função vinculada ao serviço, ela será criada novamente quando você atualizar ou criar uma distribuição do CloudFront que tenha uma associação ao Lambda@Edge.

Para criar manualmente esses perfis vinculados ao serviço, execute os seguintes comandos da AWS Command Line Interface (AWS CLI):

Para criar a função `AWSServiceRoleForLambdaReplicator`

- Execute o seguinte comando .

```
aws iam create-service-linked-role --aws-service-name
replicator.lambda.amazonaws.com
```

Para criar a função `AWSServiceRoleForCloudFrontLogger`

- Execute o seguinte comando .

```
aws iam create-service-linked-role --aws-service-name
logger.cloudfront.amazonaws.com
```

Edição de funções vinculadas ao serviço do Lambda@Edge

O Lambda@Edge não permite que você edite os perfis vinculados ao serviço `AWSServiceRoleForLambdaReplicator` ou `AWSServiceRoleForCloudFrontLogger`. Depois que o serviço criar um perfil vinculado ao serviço, você não poderá alterar o nome dele, pois várias entidades podem fazer referência a ele. No entanto, é possível usar o IAM para editar a descrição

da função. Para ter mais informações, consulte [Editar uma função vinculada a serviço](#) no Guia do usuário do IAM.

Regiões da AWS compatíveis com funções vinculadas ao serviço do CloudFront

O CloudFront é compatível com as funções vinculadas ao serviço do Lambda@Edge nas seguintes Regiões da AWS:

- Leste dos EUA (Norte da Virgínia): us-east-1
- US East (Ohio) – us-east-2
- US West (N. California) – us-west-1
- US West (Oregon) – us-west-2
- Asia Pacific (Mumbai) – ap-south-1
- Ásia-Pacífico (Seul) – ap-northeast-2
- Asia Pacific (Singapore) – ap-southeast-1
- Ásia-Pacífico (Sydney) – ap-southeast-2
- Ásia-Pacífico (Tóquio) (ap-northeast-1)
- Europe (Frankfurt) – eu-central-1
- Europe (Ireland) – eu-west-1
- Europe (London) – eu-west-2
- South America (São Paulo) – sa-east-1

Escrever e criar uma função do Lambda@Edge

Para usar o Lambda@Edge, você escreve o código da função do AWS Lambda. Para ajudar você a escrever funções do Lambda@Edge, consulte os seguintes recursos:

- [Estrutura de eventos do Lambda@Edge](#): entenda a estrutura de eventos a ser usada com o Lambda@Edge.
- [Funções de exemplo do Lambda@Edge](#): funções de exemplo, como testes A/B e geração de um redirecionamento HTTP.

O modelo de programação para usar o Node.js com o Lambda@Edge é o mesmo que para usar o Lambda em uma Região da AWS. Consulte mais informações em [Criar funções do Lambda com Node.js](#) ou em [Criar funções do Lambda com Python](#) no Guia do desenvolvedor do AWS Lambda.

Na função do Lambda@Edge, inclua o parâmetro `callback` e retorne o objeto aplicável para eventos de solicitação ou resposta:

- Eventos de solicitação: inclua o objeto `cf.request` na resposta.

Se você estiver gerando uma resposta, inclua o objeto `cf.response` na resposta. Para ter mais informações, consulte [Gerar respostas de HTTP em acionadores da solicitação](#).

- Eventos de resposta: inclua o objeto `cf.response` na resposta.

Depois de escrever seu próprio código ou usar um dos exemplos, você cria a função no Lambda. Para criar uma função ou editar uma existente, consulte os seguintes tópicos:

Tópicos

- [Criar uma função do Lambda@Edge](#)
- [Editar uma função do Lambda](#)

Depois de criar a função no Lambda, você configura o Lambda para executar a função com base em eventos específicos do CloudFront, chamados acionadores. Para ter mais informações, consulte [Adicionar acionadores para uma função do Lambda@Edge](#).

Criar uma função do Lambda@Edge

Para configurar o AWS Lambda para executar funções Lambda baseadas em eventos do CloudFront, siga este procedimento.

Para criar uma função Lambda@Edge

1. Faça login no AWS Management Console e abra o console AWS Lambda em <https://console.aws.amazon.com/lambda/>.
2. Se você já tiver uma ou mais funções do Lambda, escolha Create function (Criar função).


Se você não tiver nenhuma função, escolha Get Started Now.
3. Na lista Region (Região) na parte superior da página, escolha US East (N. Virginia) (Leste dos EUA (Norte da Virgínia)).
4. Crie uma função usando seu próprio código ou crie uma função começando com um esquema do CloudFront.
 - Para criar uma função usando seu próprio código, escolha Author from scratch.

- Para exibir uma lista de esquemas do CloudFront, insira cloudfront no campo de filtro e selecione Enter.

Se você encontrar um esquema que deseja usar, selecione o nome dele.

5. Na seção Basic information, especifique os seguintes valores:
 - a. Nome: insira um nome para a função.
 - b. Função: para começar rapidamente, escolha Criar função com base em modelo(s). Também é possível selecionar Escolher uma função existente ou Criar uma função personalizada e siga as instruções para concluir as informações dessa seção.
 - c. Nome da função: insira um nome para a função.
 - d. Modelos de política: escolha Permissões básicas do Edge Lambda.
6. Se você escolheu Author from scratch na etapa 4, vá para a etapa 7.

Se você escolheu um esquema na etapa 4, a seção cloudfront permitirá que você crie um trigger, que associa essa função a um cache em uma distribuição do CloudFront e a um evento do CloudFront. Recomendamos que você escolha Remove (Remover) nesse momento, para que não haja um trigger para a função quando ela for criada. Você poderá adicionar gatilhos mais tarde.

 Tip

Recomendamos que você teste e depure a função antes de adicionar acionadores. Se você adicionar um acionador agora, a função será executada assim que for criada e concluirá a replicação para locais da AWS em todo o mundo, e a distribuição correspondente será implantada.

7. Escolha a opção Criar função.

O Lambda cria duas versões da sua função: \$LATEST e Versão 1. Você pode editar apenas a versão \$LATEST, mas o console inicialmente exibirá a Versão 1.
8. Para editar a função, escolha Version 1 na parte superior da página, sob o ARN da função. Na guia Versions, escolha \$LATEST. (Se você deixou a função e depois retornou a ela, o título do botão será Qualifiers.)
9. Na guia Configuration, escolha o Code entry type aplicável. Em seguida, siga as instruções para editar ou fazer upload do seu código.

10. Em Runtime, escolha o valor com base no código da função.
11. Na seção Tags, adicione todas as tags aplicáveis.
12. Escolha Actions e, em seguida, Publish new version.
13. Insira uma descrição para a nova versão da função.
14. Escolha Publish.
15. Teste e depure a função. Consulte mais informações sobre como testar no console do Lambda em [Invocar a função do Lambda usando o console](#) no Guia do desenvolvedor do AWS Lambda.
16. Quando você estiver pronto para que a função seja executada em eventos do CloudFront, publique outra versão e edite-a para adicionar triggers. Para ter mais informações, consulte [Adicionar acionadores para uma função do Lambda@Edge](#).

Editar uma função do Lambda

Depois de criar uma função do Lambda@Edge, você pode usar o console do Lambda para editá-la.

Observações

- A versão original é identificada como \$ LATEST.
- Você só pode editar a versão \$LATEST.
- Cada vez que você editar a versão \$LATEST, deverá publicar uma nova versão numerada.
- Você não pode criar triggers para \$LATEST.
- Ao publicar uma nova versão de uma função, o Lambda não copiará automaticamente os triggers da versão anterior para a nova versão. Você deve reproduzir os triggers para a nova versão.
- Quando você adiciona um trigger de um evento do CloudFront a uma função, se já houver um trigger para a mesma distribuição, comportamento de cache e evento para uma versão anterior da mesma função, o Lambda excluirá o trigger da versão anterior.
- Depois de fazer atualizações em uma distribuição do CloudFront, como a adição de triggers, você precisará aguardar a propagação das alterações para os pontos de presença para que as funções especificadas nos triggers funcionem.

Para editar uma função do Lambda

1. Faça login no AWS Management Console e abra o console AWS Lambda em <https://console.aws.amazon.com/lambda/>.
2. Na lista Region (Região) na parte superior da página, escolha US East (N. Virginia) (Leste dos EUA (Norte da Virgínia)).
3. Na lista de funções, selecione o nome da função.

Por padrão, o console exibirá a versão \$LATEST. Você pode visualizar versões anteriores (escolha Qualifiers), mas só poderá editar \$LATEST.

4. Na guia Code (Código), em Code entry type (Tipo de entrada de código), escolha editar o código no navegador, fazer upload de um arquivo .zip ou fazer upload de um arquivo do Amazon S3.
5. Selecione Save ou Save and test.
6. Escolha Actions e Publish new version.
7. Na caixa de diálogo Publish new version from \$LATEST, insira uma descrição da nova versão. Essa descrição aparece na lista de versões, junto com um número de versão gerado automaticamente.
8. Escolha Publish.

A nova versão se torna automaticamente a versão mais recente. O número da versão aparece em Versão no canto superior esquerdo da página.

Note

Se ainda não tiver adicionado acionadores para a função, consulte [Adicionar acionadores para uma função do Lambda@Edge](#).

9. Selecione a guia Triggers.
10. Escolha Add trigger.
11. Na caixa de diálogo Add trigger (Adicionar trigger), selecione a caixa pontilhada e escolha CloudFront.

Note

Se você já tiver criado um ou mais triggers para uma função, o CloudFront será o serviço padrão.

12. Especifique os seguintes valores para indicar quando você deseja que a função Lambda seja executada.
 - a. ID da distribuição: escolha o ID da distribuição à qual você deseja adicionar o acionador.
 - b. Comportamento de cache: escolha o comportamento de cache que especifica os objetos nos quais você deseja executar a função.
 - c. Evento do CloudFront: escolha o evento do CloudFront que faz com que a função seja executada.
 - d. Ativar o trigger e replicar: marque essa caixa de seleção para que o Lambda replique a função para Regiões da AWS globalmente.
13. Selecione Enviar.
14. Para adicionar mais triggers a essa função, repita as etapas de 10 a 13.

Consulte mais informações sobre como testar e depurar a função no console do Lambda em [Invocar a função do Lambda usando o console](#) no Guia do desenvolvedor do AWS Lambda.

Quando você estiver pronto para que a função seja executada em eventos do CloudFront, publique outra versão e edite-a para adicionar triggers. Para ter mais informações, consulte [Adicionar acionadores para uma função do Lambda@Edge](#).

Adicionar acionadores para uma função do Lambda@Edge

Gatilho do Lambda@Edge é uma combinação de distribuição do CloudFront, comportamento de cache e evento que faz com que a função seja executada. Por exemplo, você pode criar um trigger que faça com que a função seja executada quando o CloudFront receber uma solicitação de um visualizador para um determinado comportamento do cache de sua configuração da distribuição. Você pode especificar um ou mais acionadores do CloudFront.

Tip

Ao criar uma distribuição do CloudFront, você especifica as configurações que informam ao CloudFront como responder ao receber diferentes solicitações. As configurações padrão são chamadas de comportamento de cache padrão para a distribuição. Você pode configurar comportamentos de cache adicionais que definem como o CloudFront responde em circunstâncias específicas, por exemplo, quando recebe uma solicitação para um tipo de

arquivo específico. Para ter mais informações, consulte [Configurações de comportamento de cache](#).

Ao criar uma função do Lambda pela primeira vez, é possível especificar apenas um acionador. É possível adicionar mais acionadores à mesma função mais tarde usando o console do Lambda ou editando a distribuição no console do CloudFront.

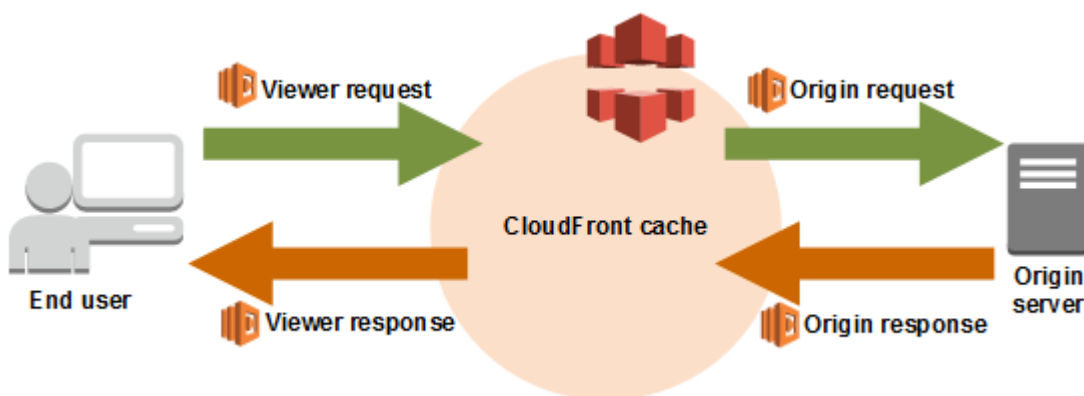
- O console do Lambda funciona bem para adicionar mais acionadores a uma função da mesma distribuição do CloudFront.
- O console do CloudFront pode ser melhor para adicionar acionadores a várias distribuições, pois é mais fácil encontrar a distribuição que você deseja atualizar. Também é possível atualizar outras configurações do CloudFront ao mesmo tempo.

Tópicos

- [Eventos do CloudFront que podem acionar uma função do Lambda@Edge](#)
- [Escolher o evento para acionar a função](#)
- [Adicionar acionadores a uma função do Lambda@Edge](#)

Eventos do CloudFront que podem acionar uma função do Lambda@Edge

Para cada comportamento do cache em uma distribuição do Amazon CloudFront, é possível adicionar até quatro acionadores (associações) que fazem com que uma função do Lambda seja executada quando ocorrem eventos específicos do CloudFront. Os triggers de CloudFront podem ser baseados em um dos quatro eventos do CloudFront, conforme mostrado no diagrama a seguir.



Os eventos do CloudFront que podem ser usados para acionar as funções do Lambda@Edge são os seguintes:

Solicitação do visualizador

A função é executada quando o CloudFront recebe uma solicitação de um visualizador antes de ele verificar se o objeto solicitado está no cache do CloudFront.

A função não é executada nos seguintes casos:

- Ao buscar uma página de erro personalizada.
- Quando o CloudFront redireciona automaticamente uma solicitação HTTP para HTTPS (quando o valor de [Política de protocolo do visualizador](#) é Redirecionar HTTP para HTTPS).

Solicitação da origem

A função é executada apenas quando o CloudFront encaminha uma solicitação para a origem. Quando o objeto solicitado está no cache do CloudFront, a função não é executada.

Resposta da origem

A função é executada depois que o CloudFront recebe uma resposta da origem e antes de ele armazenar o objeto em cache na resposta. Observe que a função é executada mesmo se um erro for retornado da origem.

A função não é executada nos seguintes casos:

- Quando o arquivo solicitado está no cache do CloudFront e não expirou.
- Quando a resposta é gerada de uma função acionada por um evento de solicitação de origem.

Resposta do visualizador

A função é executada antes de retornar o arquivo solicitado para o visualizador. Observe que a função é executada independentemente de o arquivo já estar no cache do CloudFront.

A função não é executada nos seguintes casos:

- Quando a origem retorna um código de status HTTP 400 ou posterior.
- Quando uma página de erro personalizada é retornada.
- Quando a resposta é gerada de uma função acionada por um evento de solicitação do visualizador.
- Quando o CloudFront redireciona automaticamente uma solicitação HTTP para HTTPS (quando o valor de [Política de protocolo do visualizador](#) é Redirecionar HTTP para HTTPS).

Ao adicionar vários triggers para o mesmo comportamento do cache, você pode usá-los para executar a mesma função ou executar funções diferentes para cada trigger. Você também pode associar a mesma função para mais de uma distribuição.

Note

Quando um evento do CloudFront aciona a execução de uma função do Lambda, a função deve ser concluída antes que o CloudFront possa continuar.

Por exemplo, se uma função do Lambda for acionada por um evento de solicitação do visualizador do CloudFront, este não retornará uma resposta ao visualizador nem encaminhará a solicitação à origem enquanto a execução da função do Lambda não for concluída.

Isso significa que cada solicitação que aciona uma função do Lambda aumenta a latência da solicitação. Portanto, é conveniente executar a função o mais rapidamente possível.

Escolher o evento para acionar a função

Ao decidir qual evento do CloudFront você deseja usar para acionar uma função do Lambda, considere:

Quero que o CloudFront armazene em cache objetos que são alterados por uma função do Lambda

Para armazenar em cache um objeto que foi modificado por uma função do Lambda para que o CloudFront possa fornecer o objeto no local da borda na próxima vez que ele for solicitado, use os eventos de solicitação da origem ou de resposta da origem.

Isso reduz a carga na origem, a latência de solicitações subsequentes e o custo de chamar o `Lambda@Edge` em solicitações subsequentes.

Por exemplo, para adicionar, remover ou alterar os cabeçalhos dos objetos retornados pela origem e quiser que o CloudFront armazene o resultado em cache, use o evento de resposta da origem.

Quero que a função seja executada em todas as solicitações

Se quiser executar a função para todas as solicitações que o CloudFront recebe para a distribuição, use os eventos de solicitação do visualizador ou de resposta do visualizador.

Os eventos de solicitação e de resposta da origem ocorrem somente quando um objeto solicitado não é armazenado em cache em um ponto de presença, e o CloudFront encaminha uma solicitação para a origem.

Quero que a função altere a chave de cache

Para alterar um valor que está sendo usado como base para o armazenamento em cache, use o evento de solicitação do visualizador.

Por exemplo, se uma função altera o URL para incluir a abreviação de um idioma no caminho (por exemplo, porque o usuário escolheu o idioma em uma lista suspensa), use o evento de solicitação do visualizador:

- URL na solicitação do visualizador: `https://example.com/en/index.html`
- URL quando a solicitação é proveniente de um endereço IP na Alemanha: `https://example.com/de/index.html`

O evento de solicitação do visualizador também pode ser usada se você estiver armazenando em cache com base em cookies ou cabeçalhos de solicitação.

Note

Se a função alterar cookies ou cabeçalhos, configure o CloudFront para encaminhar a parte aplicável da solicitação à origem. Para obter mais informações, consulte os tópicos a seguir.

- [Conteúdo em cache com base em cookies](#)
- [Conteúdo em cache com base nos cabeçalhos de solicitação](#)

A função afeta a resposta da origem

Para alterar a solicitação de maneira a afetar a resposta da origem, use o evento de solicitação de origem.

Normalmente, a maioria dos eventos de solicitação do visualizador não é encaminhada para a origem. O CloudFront responde a uma solicitação com um objeto que já está no cache da borda. Se a função alterar a solicitação com base em um evento de solicitação de origem, o CloudFront armazenará em cache a resposta à solicitação de origem alterada.

Adicionar acionadores a uma função do Lambda@Edge

É possível usar o console do AWS Lambda ou o console do Amazon CloudFront para adicionar um acionador à função do Lambda@Edge.

Important

É possível criar acionadores somente para versões numeradas da função (não para \$LATEST).

Lambda console

Como adicionar acionadores para eventos do CloudFront a uma função do Lambda@Edge

1. Faça login no AWS Management Console e abra o console AWS Lambda em <https://console.aws.amazon.com/lambda/>.
2. Na lista Region (Região) na parte superior da página, escolha US East (N. Virginia) (Leste dos EUA (Norte da Virgínia)).
3. Na página Functions, selecione o nome da função à qual você deseja adicionar triggers.
4. Na página de Function overview (Visão geral da função), escolha a guia Versions (Versões).
5. Escolha a versão à qual você deseja adicionar triggers.

Depois de escolher uma versão, o nome do botão mudará para Version: \$LATEST ou para Version: número da versão.


6. Selecione a guia Triggers.
7. Escolha Add trigger.
8. Em Configuração do gatilho, escolha Selecione uma origem, insira **cloudfront** e selecione CloudFront.

Note

Se você já tiver criado um ou mais triggers, o CloudFront será o serviço padrão.

9. Especifique os seguintes valores para indicar quando você deseja que a função Lambda seja executada.
 - a. Distribuição: escolha a distribuição à qual você deseja adicionar o acionador.

- b. Comportamento de cache: escolha o comportamento de cache que especifica os objetos nos quais você deseja executar a função.

 Note

Se você especificar * para o comportamento do cache, a função do Lambda será implantada no comportamento do cache padrão.

- c. Evento do CloudFront: escolha o evento do CloudFront que faz com que a função seja executada.
- d. Incluir corpo: marque essa caixa de seleção se quiser acessar o corpo da solicitação na função.
- e. Confirmar implantação no Lambda@Edge: marque essa caixa de seleção para que o AWS Lambda replique a função para Regiões da AWS globalmente.

10. Escolha Adicionar.

A função começará a processar solicitações para os eventos do CloudFront especificados quando a distribuição atualizada do CloudFront for implantada. Para determinar se uma distribuição foi implantada, escolha Distributions no painel de navegação. Quando uma distribuição estiver implantada, o valor da coluna Status da distribuição mudará de Deploying (Implantando) para a data e hora da implantação.

CloudFront console

Como adicionar acionadores para eventos do CloudFront a uma função do Lambda@Edge

1. Obtenha o Nome de região da Amazon (ARN) da função do Lambda à qual você deseja adicionar triggers:
 - a. Faça login no AWS Management Console e abra o console AWS Lambda em <https://console.aws.amazon.com/lambda/>.
 - b. Na lista de regiões na parte superior da página, escolha US East (N. Virginia) (Leste dos EUA (Norte da Virgínia)).
 - c. Na lista de funções, selecione o nome da função à qual você deseja adicionar triggers.
 - d. Na página Function overview (Visão geral da função), escolha a guia Versions (Versões) e escolha a versão numerada à qual você deseja adicionar acionadores.

- e. Selecione Copy ARN (Copiar ARN) para copiar o código para a área de transferência. O ARN da função do Lambda tem a seguinte aparência:

```
arn:aws:lambda:us-east-1:123456789012:function:TestFunction:2
```

O número no final (2, no exemplo) é o número da versão da função.

2. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
3. Na lista de distribuição, selecione o ID da distribuição à qual você deseja adicionar triggers.
4. Escolha a guia Behaviors.
5. Selecione o comportamento do cache ao qual você deseja adicionar acionadores e, em seguida, escolha Edit (Editar).
6. Em Function associations (Associações da função), na lista Function type (Tipo da função), escolha Lambda@Edge para quando a função deve ser executada: para solicitações do visualizador, respostas do visualizador, solicitações da origem ou respostas da origem.

Para ter mais informações, consulte [Escolher o evento para acionar a função](#).

7. Na caixa de texto Function ARN / Name (Função ARN/Nome), cole o ARN da função do Lambda que você deseja executar quando o evento escolhido ocorrer. Esse é o valor copiado do console do Lambda.
8. Selecione Include body (Incluir corpo) se quiser acessar o corpo da solicitação na sua função.

Se deseja apenas substituir o corpo da solicitação, você não precisa selecionar essa opção.

9. Para executar a mesma função em outros tipos de evento, repita as etapas 6 e 7.
10. Escolha Salvar alterações.
11. Para adicionar acionadores em outros comportamentos de cache para essa distribuição, repita as etapas 5 a 10.

A função começará a processar solicitações para os eventos do CloudFront especificados quando a distribuição atualizada do CloudFront for implantada. Para determinar se uma distribuição foi implantada, escolha Distributions no painel de navegação. Quando uma distribuição estiver implantada, o valor da coluna Status da distribuição mudará de Deploying (Implantando) para a data e hora da implantação.

Testar e depurar as funções do Lambda@Edge

É importante testar o código da sua função do Lambda@Edge independentemente para ter certeza de que ele conclui a tarefa pretendida e fazer testes de integração para garantir que a função funcione corretamente com o CloudFront.

Durante o teste de integração ou depois que a função foi implantada, talvez seja necessário depurar erros do CloudFront, como erros HTTP 5xx. Os erros podem ser uma resposta inválida retornada da função do Lambda, erros de execução quando a função é acionada ou erros devido a uma limitação de execução do serviço do Lambda. As seções neste tópico compartilham estratégias para determinar qual tipo de falha é o problema e, em seguida, as etapas que você pode realizar para corrigir o problema.

Note

Ao revisar os arquivos de log ou as métricas do CloudWatch durante a solução de erros, esteja ciente de que eles são exibidos ou armazenados na Região da AWS mais próxima do local em que a função foi executada. Portanto, se você tiver um site ou aplicação web com usuários no Reino Unido e, por exemplo, tiver uma função do Lambda associada à distribuição, deverá alterar a região para visualizar as métricas ou os arquivos de log do CloudWatch para a Região da AWS Londres. Para ter mais informações, consulte [the section called “ Determinar a região do Lambda@Edge”](#).

Tópicos

- [Testar as funções do Lambda@Edge](#)
- [Identificar erros de função do Lambda@Edge no CloudFront](#)
- [Solução de problemas de respostas inválidas de funções do Lambda@Edge \(erros de validação\)](#)
- [Solução de problemas de erros de execução de funções do Lambda@Edge](#)
- [Determinar a região do Lambda@Edge](#)
- [Determinar se a sua conta envia logs ao CloudWatch](#)

Testar as funções do Lambda@Edge

Há duas etapas para testar a função do Lambda: teste autônomo e teste de integração.

Testar a funcionalidade autônoma

Antes de adicionar a função do Lambda ao CloudFront, teste a funcionalidade primeiro usando os recursos de teste no console do Lambda ou usando outros métodos. Consulte mais informações sobre como testar no console do Lambda em [Invocar a função do Lambda usando o console](#) no Guia do desenvolvedor do AWS Lambda.

Testar a operação da função no CloudFront

É importante concluir o teste de integração, quando a função está associada a uma distribuição e é executada com base em um evento do CloudFront. Certifique-se de que a função seja acionada para o evento correto e retorne uma resposta válida e correta para o CloudFront. Por exemplo, verifique se a estrutura do evento está correta, se apenas os cabeçalhos válidos estão incluídos e assim por diante.

Ao iterar os testes de integração com a função no console do Lambda, consulte as etapas do tutorial do Lambda@Edge à medida que você modifica o código ou altera o trigger do CloudFront que chama a função. Por exemplo, verifique se você está trabalhando em uma versão numerada da função, conforme descrito nesta etapa do tutorial: [Etapa 4: adicionar um acionador do CloudFront para executar a função](#).

Ao fazer alterações e implantá-las, lembre-se de que a função e os triggers atualizados do CloudFront levarão vários minutos para serem replicados em todas as regiões. Isso geralmente leva alguns minutos, mas pode demorar até 15 minutos.

Você pode verificar se a replicação foi concluída acessando o console do CloudFront e visualizando a distribuição.

Como verificar se a implantação da replicação foi concluída

1. Abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha o nome da distribuição.
3. Verifique o status de distribuição para mudar de In Progress (Em progresso) para Deployed (Implantado), o que significa que sua função foi replicada. Em seguida, siga as etapas na próxima seção para verificar se a função funciona.

Esteja ciente de que o teste no console valida apenas a lógica da sua função e não aplica cotas de serviço (anteriormente conhecidas como limites) específicas do Lambda@Edge.

Identificar erros de função do Lambda@Edge no CloudFront

Depois de verificar se a lógica da função funciona corretamente, você ainda poderá ver erros HTTP 5xx quando a função for executada no CloudFront. Os erros HTTP 5xx podem ser retornados por vários motivos, que incluem erros ou outros problemas da função do Lambda no CloudFront.

- Ao usar as funções do Lambda@Edge, é possível usar gráficos no console do CloudFront para ajudar a identificar o que está causando o erro e corrigi-lo. Por exemplo, é possível ver se os erros HTTP 5xx são causados pelo CloudFront ou pelas funções do Lambda e, no caso de funções específicas, visualizar os arquivos de log relacionados para investigar o problema.
- Para solucionar a maioria dos erros HTTP do CloudFront, consulte as etapas de solução de problemas no seguinte tópico: [Solucionar problemas de códigos de status de resposta de erros no CloudFront](#).

O que causa erros na função do Lambda@Edge no CloudFront

Existem vários motivos pelos quais uma função do Lambda pode causar um erro HTTP 5xx, e as etapas de solução de problemas que você deve seguir dependem do tipo de erro. Os erros podem ser categorizados assim:

Um erro de execução de função do Lambda

Ocorre um erro de execução quando o CloudFront não obtém uma resposta do Lambda porque existem exceções não tratadas na função ou há um erro no código. Por exemplo, se o código incluir um retorno de chamada (erro).

Uma resposta inválida da função do Lambda é retornada ao CloudFront

Após a execução da função, o CloudFront recebe uma resposta do Lambda. Um erro será retornado se a estrutura do objeto da resposta não estiver em conformidade com o [Estrutura de eventos do Lambda@Edge](#) ou se a resposta contiver cabeçalhos inválidos ou outros campos inválidos.

A execução no CloudFront é limitada devido às cotas de serviço do Lambda (anteriormente conhecidas como limites)

O serviço do Lambda limita as execuções em cada região e retornará um erro se você exceder a cota. Para ter mais informações, consulte [Cotas do Lambda@Edge](#).

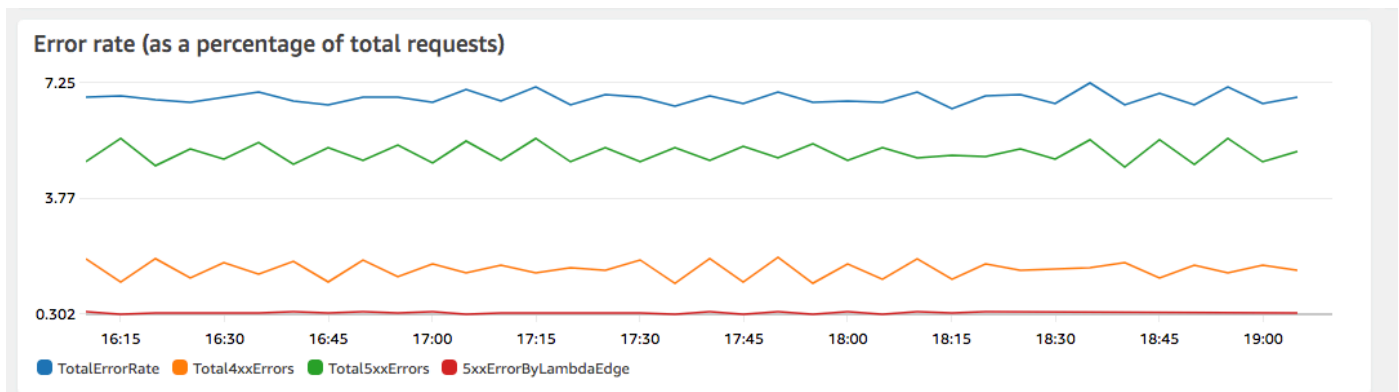
Como determinar o tipo de falha

Para ajudar a decidir onde se concentrar na depuração de erros e solucionar os problemas retornados pelo CloudFront, é útil identificar por que o CloudFront está retornando um erro HTTP. Para iniciar a busca, você pode usar os gráficos fornecidos na seção Monitoring (Monitoramento) do console do CloudFront no AWS Management Console. Para obter mais informações sobre como visualizar gráficos na seção Monitoring (Monitoramento) no console do CloudFront, consulte [Monitorar métricas do CloudFront com o Amazon CloudWatch](#).

Os seguintes gráficos podem ser especialmente úteis ao verificar se os erros estão sendo retornados por origens ou por uma função do Lambda, e para restringir o tipo de problema quando se trata de um erro de função do Lambda.

Gráfico de taxas de erro

Um dos gráficos que você pode visualizar na guia Overview (Visão geral) para cada uma das suas distribuições é um gráfico Error rates (Taxas de erro). Esse gráfico exibe a taxa de erros como porcentagem do total de solicitações que chegam à sua distribuição. O gráfico mostra o total de taxa de erros, total de erros 4xx, total de erros 5xx e total de erros 5xx provenientes de funções Lambda. Com base no tipo de erro e volume, você pode executar etapas para investigar e solucionar o problema.



- Se você encontrar erros do Lambda, você poderá investigar mais detalhadamente, observando os tipos específicos de erros retornados pela função. A guia Lambda@Edge errors (Erros do Lambda@Edge) inclui gráficos que categorizam os erros por tipo de função para ajudar a identificar o problema em uma função específica.
- Se encontrar erros do CloudFront, você poderá solucionar e trabalhar para corrigir os erros de origem ou alterar a configuração do CloudFront. Para ter mais informações, consulte [Solucionar problemas de códigos de status de resposta de erros no CloudFront](#).

Gráficos de erros de execução e repostas de funções inválidas

A guia `Lambda@Edge errors` (Erros do `Lambda@Edge`) inclui gráficos que categorizam os erros do `Lambda@Edge` em uma distribuição específica, por tipo. Por exemplo, um grafo mostra todos os erros de execução por Região da AWS.

Para facilitar a solução de problemas, você pode procurar problemas específicos ao abrir e examinar os arquivos de log para funções específicas por região.

Como visualizar os arquivos de log de uma função específica por região

1. Na guia `Erros do Lambda@Edge`, em `Funções associadas do Lambda@Edge`, escolha o nome da função e selecione `Visualizar métricas`.
2. Em seguida, na página com o nome da função, no canto superior direito, selecione `Visualizar logs da função` e escolha uma região.

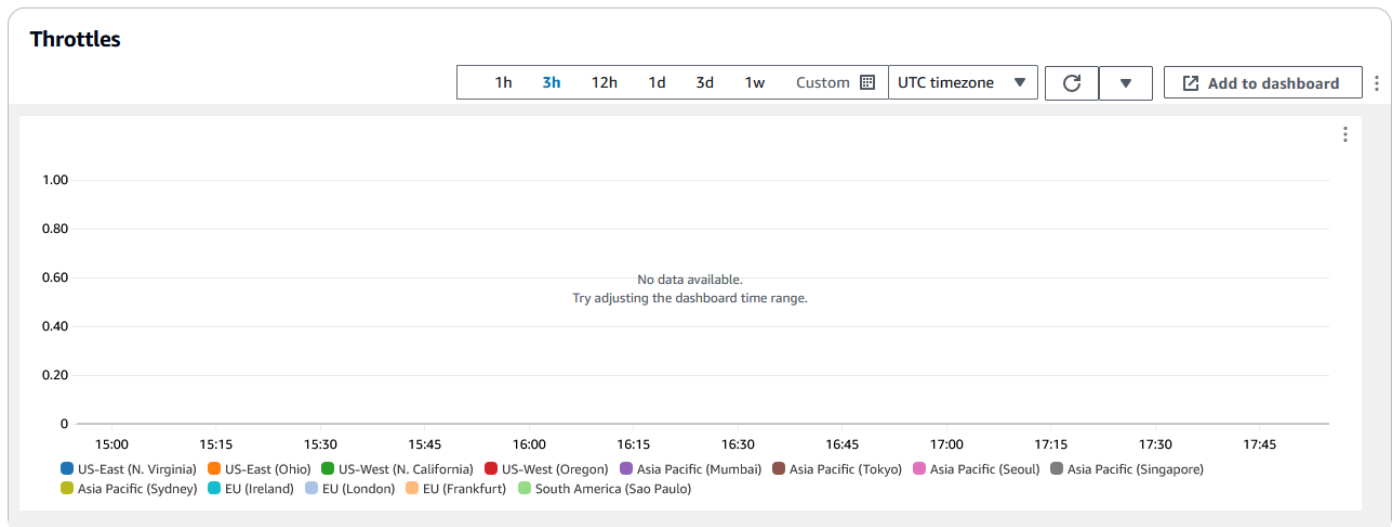
Por exemplo, se você encontrar problemas no grafo `Erros` para a região `Oeste dos EUA (Oregon)`, escolha essa região na lista suspensa. Isso abrirá o console do `Amazon CloudWatch`.

3. No console do `CloudWatch` dessa região, em `Fluxos de log`, escolha um fluxo de log para visualizar os eventos da função.

Também leia as seguintes seções neste capítulo para obter mais recomendações sobre como solucionar problemas e corrigir erros.

Gráfico de limitações

A guia `Lambda@Edge errors` (Erros do `Lambda@Edge`) também inclui um gráfico de `Throttles` (Limitações). Ocasionalmente, o serviço `Lambda` limita as invocações de função por região quando a cota (anteriormente conhecida como limite) de simultaneidade regional é atingida. Se você encontrar um erro `limit exceeded` (limite excedido), isso significa que a função atingiu uma cota que o serviço `Lambda` impõe para execuções em uma região. Para obter mais informações, incluindo como solicitar um aumento na cota, consulte [Cotas do Lambda@Edge](#).



Para obter um exemplo de como usar essas informações para solucionar erros HTTP, consulte [Quatro etapas para depurar a entrega de conteúdo na AWS](#).

Solução de problemas de respostas inválidas de funções do Lambda@Edge (erros de validação)

Se você identificar que o problema é um erro de validação do Lambda, a função do Lambda poderá estar retornando uma resposta inválida ao CloudFront. Siga as orientações nesta seção para tomar medidas para analisar a função e garantir que a resposta esteja de acordo com os requisitos do CloudFront.

O CloudFront valida a resposta de uma função do Lambda de duas maneiras:

- A resposta do Lambda deve estar de acordo com a estrutura de objeto necessária. Exemplos de estruturas de objeto inválidas incluem o seguinte: JSON não analisável, campos obrigatórios ausentes e um objeto inválido na resposta. Para obter mais informações, consulte [Estrutura de eventos do Lambda@Edge](#).
- A resposta deve incluir apenas valores de objeto válidos. Um erro ocorrerá se a resposta incluir um objeto válido, mas tiver valores sem suporte. Os exemplos incluem o seguinte: adicionar ou atualizar cabeçalhos permitidos ou somente leitura (consulte [Restrições das funções de borda](#)), exceder o tamanho máximo do corpo (consulte Restrições sobre o tamanho da resposta gerada no tópico [Erros](#) do Lambda@Edge) e caracteres ou valores inválidos (consulte o [Estrutura de eventos do Lambda@Edge](#)).

Quando o Lambda retorna uma resposta inválida ao CloudFront, as mensagens de erro são gravados em arquivos de log que o CloudFront envia por push ao CloudWatch na região em que a função do Lambda foi executada. O comportamento padrão é enviar os arquivos de log ao CloudWatch quando há uma resposta inválida. No entanto, se você tiver associado uma função do Lambda ao CloudFront antes do lançamento dessa funcionalidade, talvez ela não esteja habilitada para a função. Para obter mais informações, consulte [Determinar se a conta envia logs por push ao CloudWatch](#), mais adiante neste tópico.

O CloudFront envia arquivos de log à região correspondente ao local onde a função foi executada, no grupo de logs associado à sua distribuição. Os grupos de log têm o seguinte formato: `/aws/cloudfront/LambdaEdge/DistributionId`, em que *DistributionId* é o ID da distribuição. Para determinar a região na qual você pode encontrar os arquivos de log do CloudWatch, consulte [Determinar a região do Lambda@Edge](#) mais adiante neste tópico.

Se for possível reproduzir o erro, você poderá criar uma nova solicitação que resulte no erro e encontrar o id da solicitação em uma resposta do CloudFront com falha (cabeçalho `X-Amz-Cf-Id`) para localizar uma única falha nos arquivos de log. A entrada do arquivo de log inclui informações que podem ajudar a identificar porque o erro está sendo retornado, e também lista o id da solicitação do Lambda correspondente, para que você possa analisar a causa raiz no contexto de uma única solicitação.

Se um erro for intermitente, você poderá usar os logs de acesso do CloudFront para encontrar o id de uma solicitação que falhou e depois pesquisar as mensagens de erro correspondentes nos CloudWatch Logs. Para mais informações, consulte a seção anterior, [Determinar o tipo de falha](#).

Solução de problemas de erros de execução de funções do Lambda@Edge

Se o problema for um erro de execução do Lambda, poderá ser útil criar instruções de registro em log para funções do Lambda, gravar mensagens nos arquivos de log CloudWatch que monitoram a execução da função no CloudFront e determinar se ela está funcionando conforme o esperado. Depois, você pode pesquisar essas instruções nos arquivos de log do CloudWatch para verificar se a sua função está funcionando.

Note

Mesmo que você não tenha alterado a função do Lambda@Edge, as atualizações no ambiente de execução da função do Lambda podem afetá-la e um erro de execução poderá ser retornado. Para obter informações sobre como testar e migrar para uma versão mais

recente, consulte [Próximas atualizações no AWS Lambda e no ambiente de execução do AWS Lambda@Edge](#).

Determinar a região do Lambda@Edge

Para ver as regiões em que a função do Lambda@Edge está recebendo tráfego, visualize os gráficos das métricas da função no console do CloudFront no AWS Management Console. As métricas são exibidas para cada região da AWS. Na mesma página, é possível escolher uma região e visualizar os arquivos de log para essa região a fim de investigar problemas. Revise os arquivos de log do CloudWatch na região correta da AWS para ver os arquivos de log criados quando o CloudFront executou a função Lambda.

Para obter mais informações sobre como visualizar gráficos na seção Monitoring (Monitoramento) no console do CloudFront, consulte [Monitorar métricas do CloudFront com o Amazon CloudWatch](#).

Determinar se a sua conta envia logs ao CloudWatch

Por padrão, o CloudFront habilita o registro em log de respostas de função inválidas do Lambda e envia por push os arquivos de log para o CloudWatch usando uma das [Funções vinculadas ao serviço para o Lambda@Edge](#). Se você tiver funções do Lambda@Edge adicionadas ao CloudFront antes do lançamento do recurso de log de respostas de função inválidas do Lambda, o registro em log será habilitado quando você atualizar a configuração do Lambda@Edge, por exemplo, adicionando um trigger do CloudFront.

É possível verificar se o envio por push dos arquivos de log ao CloudWatch está habilitado para a conta, fazendo o seguinte:

- Verifique se os logs aparecem no CloudWatch: examine na região em que a função do Lambda@Edge foi executada. Para ter mais informações, consulte [Determinar a região do Lambda@Edge](#).
- Determine se o perfil vinculado ao serviço relacionado existe na sua conta do IAM: você deve ter o perfil do IAM `AWSServiceRoleForCloudFrontLogger` na sua conta. Para obter mais informações sobre essa função, consulte [Funções vinculadas ao serviço para o Lambda@Edge](#).

Excluir funções e réplicas do Lambda@Edge

Só é possível excluir uma função do Lambda@Edge quando as réplicas da função tiverem sido excluídas pelo CloudFront. As réplicas de uma função do Lambda são excluídas automaticamente nas seguintes situações:

- Depois de remover a última associação da função de todas as distribuições do CloudFront. Se mais de uma distribuição usar uma função, as réplicas serão excluídas somente depois que a associação da função for removida da última distribuição.
- Depois que você excluir a última distribuição com a qual a função estava associada.

Geralmente, as réplicas são excluídas dentro de algumas horas. Não é possível excluir manualmente réplicas de função do Lambda@Edge. Isso ajuda a evitar que uma réplica ainda em uso seja excluída, o que resultaria em um erro.

Warning

Não crie aplicações que usem réplicas de função do Lambda@Edge fora do CloudFront. Essas réplicas são excluídas quando suas associações a distribuições são removidas ou quando as próprias distribuições são excluídas. A réplica da qual um aplicativo externo depende poderá ser removida sem aviso prévio, fazendo com que ele falhe.

Como excluir uma associação de função do Lambda@Edge de uma distribuição do CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha o ID da distribuição com a associação da função do Lambda@Edge que deseja excluir.
3. Escolha a guia Behaviors.
4. Selecione o comportamento de cache que possui a associação de função do Lambda@Edge que deseja excluir e selecione Edit (Editar).
5. Em Function associations (Associações da função), Function type (Tipo da função), escolha No association (Nenhuma associação) para excluir a associação da função do Lambda@Edge.
6. Escolha Salvar alterações.

Depois de excluir uma associação de função do Lambda@Edge de uma distribuição do CloudFront, você pode, opcionalmente, excluir a função Lambda ou a versão da função do AWS Lambda.

Aguarde algumas horas após excluir a associação da função para que as réplicas da função do Lambda@Edge possam ser limpas. Depois disso, é possível excluir a função usando o console do Lambda, a AWS CLI, a API do Lambda ou um AWS SDK.

Também é possível excluir uma versão específica de uma função do Lambda se ela não tiver nenhuma distribuição do CloudFront associada a ela. Depois de remover todas as associações de uma versão de função do Lambda, aguarde algumas horas. Depois, você pode excluir a versão da função.

Estrutura de eventos do Lambda@Edge

Os tópicos a seguir descrevem os objetos de eventos de solicitação e resposta que o CloudFront passa para uma função do Lambda@Edge quando ela é acionada.

Tópicos

- [Seleção de origem dinâmica](#)
- [Eventos de solicitação](#)
- [Eventos de resposta](#)

Seleção de origem dinâmica

Você pode usar [o padrão do caminho em um comportamento de cache](#) para rotear as solicitações para uma origem, de acordo com o caminho e o nome do objeto solicitado, como `images/*.jpg`. Usando o Lambda@Edge, você também pode rotear as solicitações para uma origem com base em outras características, como os valores nos cabeçalhos de solicitação.

Essa seleção de origem dinâmica pode ser útil de várias maneiras. Por exemplo, você pode distribuir solicitações em origens em diferentes áreas geográficas para ajudar com o balanceamento de carga global. Ou você pode seletivamente rotear solicitações para diferentes origens que cada servir uma determinada função: manuseio de bot, otimização de SEO, autenticação, e assim por diante. Para obter exemplos de código que demonstram como usar esse recurso, consulte [Seleção de origem dinâmica baseada em conteúdo: exemplos](#).

No evento de solicitação de origem do CloudFront, o objeto `origin` na estrutura do evento contém informações sobre a origem para a qual a solicitação seria roteada, de acordo com o padrão de caminho. Você pode atualizar os valores no objeto `origin` para rotear uma solicitação para

outra origem. Quando você atualiza o objeto `origin`, não precisa definir a origem na distribuição. Também é possível substituir um objeto de origem do Amazon S3 por um objeto de origem personalizado e vice-versa. No entanto, só é possível especificar uma única origem por solicitação; uma origem personalizada ou uma origem do Amazon S3, mas não ambas.

Eventos de solicitação

Os tópicos a seguir mostram a estrutura do objeto que o CloudFront passa para uma função do Lambda para [eventos de solicitação do visualizador e da origem](#). Estes exemplos mostram uma solicitação GET sem corpo. Após os exemplos, está uma lista de todos os campos possíveis em eventos de solicitação de visualizador e origem.

Tópicos

- [Exemplo de solicitação de visualizador](#)
- [Exemplo de solicitação de origem](#)
- [Campos de eventos de solicitação](#)

Exemplo de solicitação de visualizador

O exemplo a seguir mostra um objeto de evento de solicitação de visualizador.

```
{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "viewer-request",
          "requestId": "4TyzHTaYwb1GX1qTfsHhEqV6HUDD_BzoBZnwfNvQc_1oF26ClkoUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "host": [
              {
                "key": "Host",
                "value": "d111111abcdef8.cloudfront.net"
              }
            ]
          },
          "user-agent": [
```

```

        {
            "key": "User-Agent",
            "value": "curl/7.66.0"
        }
    ],
    "accept": [
        {
            "key": "accept",
            "value": "*/*"
        }
    ]
},
"method": "GET",
"querystring": "",
"uri": "/"
}
}
}
]
}

```

Exemplo de solicitação de origem

O exemplo a seguir mostra um objeto de evento de solicitação de origem.

```

{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "origin-request",
          "requestId": "4TyzHTaYWb1GX1qTfsHhEqV6HUDd_BzoBZnwfncQc_1oF26C1koUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "x-forwarded-for": [
              {
                "key": "X-Forwarded-For",
                "value": "203.0.113.178"
              }
            ]
          },
        },
      },
    ],
  ],
}

```

```
    "user-agent": [
      {
        "key": "User-Agent",
        "value": "Amazon CloudFront"
      }
    ],
    "via": [
      {
        "key": "Via",
        "value": "2.0 2afae0d44e2540f472c0635ab62c232b.cloudfront.net
(CloudFront)"
      }
    ],
    "host": [
      {
        "key": "Host",
        "value": "example.org"
      }
    ],
    "cache-control": [
      {
        "key": "Cache-Control",
        "value": "no-cache"
      }
    ]
  },
  "method": "GET",
  "origin": {
    "custom": {
      "customHeaders": {},
      "domainName": "example.org",
      "keepaliveTimeout": 5,
      "path": "",
      "port": 443,
      "protocol": "https",
      "readTimeout": 30,
      "sslProtocols": [
        "TLSv1",
        "TLSv1.1",
        "TLSv1.2"
      ]
    }
  },
  "querystring": "",
```

```
        "uri": "/"
      }
    }
  }
]
```

Campos de eventos de solicitação

Os dados do objeto de evento de solicitação estão contidos em dois subobjetos: `config` (`Records.cf.config`) e `request` (`Records.cf.request`). As listas a seguir descrevem os campos de cada subobjeto.

Campos no objeto de configuração

A lista a seguir descreve os campos no objeto `config` (`Records.cf.config`).

distributionDomainName (somente leitura)

O nome de domínio da distribuição associada à solicitação.

distributionID (somente leitura)

O ID da distribuição associada à solicitação.

eventType (somente leitura)

O tipo de acionador associado à solicitação: `viewer-request` ou `origin-request`.

requestId (somente leitura)

Uma string criptografada que identifica exclusivamente uma solicitação do visualizador ao CloudFront. O valor de `requestId` também aparece nos logs de acesso do CloudFront como `x-edge-request-id`. Para ter mais informações, consulte [Configurar e usar logs padrão \(logs de acesso\)](#) e [Campos padrão de arquivo de log](#).

Campos no objeto de solicitação

A lista a seguir descreve os campos no objeto `request` (`Records.cf.request`).

clientIp (somente leitura)

O endereço IP do visualizador que fez a solicitação. Se o visualizador usar um proxy HTTP ou um load balancer para enviar a solicitação, o valor será o endereço IP do proxy ou do load balancer.

Cabeçalhos (leitura/gravação)

Os cabeçalhos na solicitação. Observe o seguinte:

- As chaves no objeto `headers` são versões em letras minúsculas de nomes de cabeçalho HTTP padrão. Usar chaves em letras minúsculas dá acesso sem diferenciar letras maiúsculas e minúsculas dos valores de cabeçalho.
- Cada objeto de cabeçalho (por exemplo, `headers["accept"]` ou `headers["host"]`) é uma matriz de pares de chave-valor. Para um determinado cabeçalho, a matriz contém um par de chave-valor para cada valor na solicitação.
- `key` contém o nome do cabeçalho com distinção entre maiúsculas e minúsculas conforme ele apareceu na solicitação HTTP; por exemplo, `Host`, `User-Agent`, `X-Forwarded-For` e assim por diante.
- `value` contém o valor do cabeçalho conforme ele apareceu na solicitação HTTP.
- Quando a função do Lambda adicionar ou modificar cabeçalhos de solicitação e você não incluir o campo `key` do cabeçalho, o `Lambda@Edge` inserirá automaticamente uma `key` de cabeçalho usando o nome de cabeçalho que você fornecer. Independentemente de como você tiver formatado o nome do cabeçalho, a chave de cabeçalho inserida automaticamente será formatada com inicial maiúscula para cada parte, separada por hifens (-).

Por exemplo, você pode adicionar um cabeçalho como o seguinte, sem um `key` de cabeçalho:

```
"user-agent": [  
  {  
    "value": "ExampleCustomUserAgent/1.X.0"  
  }  
]
```

Neste exemplo, o `Lambda@Edge` insere automaticamente `"key": "User-Agent"`.

Para obter informações sobre as restrições de uso do cabeçalho, consulte [Restrições das funções de borda](#).

method (somente leitura)

O método HTTP da solicitação.

queryString (leitura/gravação)

A string de consulta, se houver, na solicitação. Se a solicitação não incluir uma string de consulta, o objeto de evento ainda incluirá `queryString` com um valor vazio. Para obter mais informações sobre query strings, consulte [Conteúdo em cache com base em parâmetros de string de consulta](#).

uri (leitura/gravação)

O caminho relativo do objeto solicitado. Se a função do Lambda modificar o valor `uri`, observe o seguinte:

- O novo valor `uri` deve começar com uma barra (/).
- Se uma função alterar o valor `uri`, isso alterará o objeto solicitado pelo visualizador.
- Se uma função alterar o valor `uri`, isso não mudará o comportamento do cache da solicitação ou da origem para a qual a solicitação é enviada.

body (leitura/gravação)

O corpo da solicitação HTTP. A estrutura `body` pode conter os seguintes campos:

inputTruncated (somente leitura)

Um sinalizador booleano que indica se o corpo foi truncado pelo Lambda@Edge. Para ter mais informações, consulte [Restrições do corpo da solicitação com a opção de incluir corpo](#).

action (leitura/gravação)

A ação que você pretende realizar com o corpo. As opções para `action` são as seguintes:

- `read-only`: esse é o padrão. Ao retornar a resposta da função do Lambda, se `action` for somente leitura, o Lambda@Edge ignorará todas as alterações em `encoding` ou em `data`.
- `replace`: especifique isso quando quiser substituir o corpo enviado à origem.

encoding (leitura/gravação)

A codificação do corpo. Ao expor o corpo à função do Lambda, o Lambda@Edge primeiro converte o corpo em base64-encoding. Se você escolher `replace` para `action` substituir o corpo, poderá optar por usar a codificação base64 (padrão) ou `text`. Se você especificar `encoding` como `base64`, mas o corpo não for um base64 válido, o CloudFront retornará um erro.

data (leitura/gravação)

O conteúdo do corpo da solicitação.

origin (leitura/gravação) (somente eventos de origem)

A origem para a qual enviar a solicitação. A estrutura de `origin` deve conter exatamente uma origem, que pode ser uma origem personalizada ou uma origem do Amazon S3. A estrutura de origem pode conter os seguintes campos:

customHeaders (leitura/gravação) (origens personalizadas e do Amazon S3)

Você pode incluir os cabeçalhos personalizados com a solicitação ao especificar o par de nome e valor do cabeçalho para cada cabeçalho personalizado. Não é possível adicionar cabeçalhos não permitidos, e não pode haver um cabeçalho com o mesmo nome em `Records.cf.request.headers`. As [notas sobre cabeçalhos de solicitação](#) também se aplicam a cabeçalhos personalizados. Para ter mais informações, consulte [Cabeçalhos personalizados que o CloudFront não pode adicionar às solicitações da origem](#) e [Restrições das funções de borda](#).

domainName (leitura/gravação) (origens personalizadas e do Amazon S3)

O nome de domínio da origem. O nome de domínio não pode estar vazio.

- Para origens personalizadas — especifique um nome de domínio DNS, como `www.example.com`. O nome de domínio não pode incluir dois-pontos (`:`) e não pode ser um endereço IP. O nome de domínio pode ter até 253 caracteres.
- Para origens do Amazon S3: especifique o nome de domínio DNS do bucket do Amazon S3, como `amzn-s3-demo-bucket.s3.eu-west-1.amazonaws.com`. O nome pode ter até 128 caracteres e deve ser todo em minúsculas.

path (leitura/gravação) (origens personalizadas e do Amazon S3)

O caminho do diretório na origem em que a solicitação deve localizar o conteúdo. O caminho deve começar com uma barra (`/`), mas não deve terminar com uma (por exemplo, não deve terminar com `example-path/`). Apenas para origens personalizadas, o caminho deve ser codificado por URL e ter um comprimento máximo de 255 caracteres.

keepaliveTimeout (leitura/gravação) (somente origens personalizadas)

O tempo, em segundos, durante o qual o CloudFront deve tentar manter a conexão com a origem depois de receber o último pacote da resposta. O valor deve ser um número de 1 a 60, inclusive.

port (leitura/gravação) (somente origens personalizadas)

A porta à qual o CloudFront deve se conectar em sua origem personalizada. A porta deve ser 80, 443 ou um número no intervalo de 1024 a 65535, inclusive.

protocol (leitura/gravação) (somente origens personalizadas)

O protocolo de conexão que o CloudFront deve usar ao se conectar à sua origem. O valor pode ser `http` ou `https`.

readTimeout (leitura/gravação) (somente origens personalizadas)

O tempo, em segundos, que o CloudFront deve esperar por uma resposta depois de enviar uma solicitação à origem. Isso também especifica o tempo que o CloudFront deve aguardar depois de receber um pacote de resposta antes de receber o próximo pacote. O valor deve ser um número de 4 a 60, inclusive.

Se seu caso de uso exigir mais de 60 segundos, você poderá solicitar uma cota maior para `Response timeout per origin`. Para ter mais informações, consulte [Cotas gerais para distribuições](#).

sslProtocols (leitura/gravação) (somente origens personalizadas)

O protocolo SSL/TLS mínimo que o CloudFront pode usar ao estabelecer uma conexão HTTPS com a origem. Os valores podem ser qualquer um dos seguintes: `TLSv1.2`, `TLSv1.1`, `TLSv1` ou `SSLv3`.

authMethod (leitura/gravação) (somente origens do Amazon S3)

Se você estiver usando uma [identidade do acesso de origem \(OAI\)](#), defina esse campo como `origin-access-identity`. Se você não estiver usando uma OAI, defina-o como `none`. Se você definir `authMethod` como `origin-access-identity`, haverá vários requisitos:

- Você deve especificar `region` (consulte o campo a seguir).
- Use o mesmo OAI ao alterar a solicitação de uma origem do Amazon S3 para outra.
- Não é possível usar uma OAI ao alterar a solicitação de uma origem personalizada para uma origem do Amazon S3.

Note

Esse campo não aceita [controle de acesso à origem \(OAC\)](#).

region (leitura/gravação) (somente origens do Amazon S3)

A região da AWS do bucket do Amazon S3. Isso é necessário apenas quando você define `authMethod` como `origin-access-identity`.

Eventos de resposta

Os tópicos a seguir mostram a estrutura do objeto que o CloudFront passa para uma função do Lambda para [eventos de resposta do visualizador e da origem](#). Após os exemplos, está uma lista de todos os campos possíveis em eventos de resposta do visualizador e da origem.

Tópicos

- [Exemplo de resposta da origem](#)
- [Exemplo de resposta do visualizador](#)
- [Campos de evento de resposta](#)

Exemplo de resposta da origem

O exemplo a seguir mostra um objeto de evento de resposta da origem.

```
{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "origin-response",
          "requestId": "4TyzHTaYWb1GX1qTfsHhEqV6HUDD_BzoBZnwfnc_1oF26ClkoUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "x-forwarded-for": [
              {
                "key": "X-Forwarded-For",
                "value": "203.0.113.178"
              }
            ],
            "user-agent": [
              {
                "key": "User-Agent",
                "value": "Amazon CloudFront"
              }
            ],
            "via": [
```

```
    {
      "key": "Via",
      "value": "2.0 8f22423015641505b8c857a37450d6c0.cloudfront.net
(CloudFront)"
    }
  ],
  "host": [
    {
      "key": "Host",
      "value": "example.org"
    }
  ],
  "cache-control": [
    {
      "key": "Cache-Control",
      "value": "no-cache"
    }
  ]
},
"method": "GET",
"origin": {
  "custom": {
    "customHeaders": {},
    "domainName": "example.org",
    "keepaliveTimeout": 5,
    "path": "",
    "port": 443,
    "protocol": "https",
    "readTimeout": 30,
    "sslProtocols": [
      "TLSv1",
      "TLSv1.1",
      "TLSv1.2"
    ]
  }
},
"queryString": "",
"uri": "/"
},
"response": {
  "headers": {
    "access-control-allow-credentials": [
      {
        "key": "Access-Control-Allow-Credentials",
```

```
    "value": "true"
  }
],
"access-control-allow-origin": [
  {
    "key": "Access-Control-Allow-Origin",
    "value": "*"
  }
],
"date": [
  {
    "key": "Date",
    "value": "Mon, 13 Jan 2020 20:12:38 GMT"
  }
],
"referrer-policy": [
  {
    "key": "Referrer-Policy",
    "value": "no-referrer-when-downgrade"
  }
],
"server": [
  {
    "key": "Server",
    "value": "ExampleCustomOriginServer"
  }
],
"x-content-type-options": [
  {
    "key": "X-Content-Type-Options",
    "value": "nosniff"
  }
],
"x-frame-options": [
  {
    "key": "X-Frame-Options",
    "value": "DENY"
  }
],
"x-xss-protection": [
  {
    "key": "X-XSS-Protection",
    "value": "1; mode=block"
  }
]
```

```

    ],
    "content-type": [
      {
        "key": "Content-Type",
        "value": "text/html; charset=utf-8"
      }
    ],
    "content-length": [
      {
        "key": "Content-Length",
        "value": "9593"
      }
    ]
  },
  "status": "200",
  "statusDescription": "OK"
}
}
}
]
}

```

Exemplo de resposta do visualizador

O exemplo a seguir mostra um objeto de evento de resposta do visualizador.

```

{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "viewer-response",
          "requestId": "4TyzHTaYwb1GX1qTfsHhEqV6HUdd_BzoBZnwfnvQc_1oF26ClkoUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "host": [
              {
                "key": "Host",
                "value": "d111111abcdef8.cloudfront.net"
              }
            ]
          }
        }
      }
    }
  ]
}

```

```
    ],
    "user-agent": [
      {
        "key": "User-Agent",
        "value": "curl/7.66.0"
      }
    ],
    "accept": [
      {
        "key": "accept",
        "value": "*/*"
      }
    ]
  },
  "method": "GET",
  "querystring": "",
  "uri": "/"
},
"response": {
  "headers": {
    "access-control-allow-credentials": [
      {
        "key": "Access-Control-Allow-Credentials",
        "value": "true"
      }
    ],
    "access-control-allow-origin": [
      {
        "key": "Access-Control-Allow-Origin",
        "value": "*"
      }
    ],
    "date": [
      {
        "key": "Date",
        "value": "Mon, 13 Jan 2020 20:14:56 GMT"
      }
    ],
    "referrer-policy": [
      {
        "key": "Referrer-Policy",
        "value": "no-referrer-when-downgrade"
      }
    ]
  }
},
```



```
"server": [
  {
    "key": "Server",
    "value": "ExampleCustomOriginServer"
  }
],
"x-content-type-options": [
  {
    "key": "X-Content-Type-Options",
    "value": "nosniff"
  }
],
"x-frame-options": [
  {
    "key": "X-Frame-Options",
    "value": "DENY"
  }
],
"x-xss-protection": [
  {
    "key": "X-XSS-Protection",
    "value": "1; mode=block"
  }
],
"age": [
  {
    "key": "Age",
    "value": "2402"
  }
],
"content-type": [
  {
    "key": "Content-Type",
    "value": "text/html; charset=utf-8"
  }
],
"content-length": [
  {
    "key": "Content-Length",
    "value": "9593"
  }
]
},
"status": "200",
```

```
        "statusDescription": "OK"
      }
    }
  }
]
```

Campos de evento de resposta

Os dados do objeto de evento de resposta estão contidos em três subobjetos: `config` (`Records.cf.config`), `request` (`Records.cf.request`) e `response` (`Records.cf.response`). Para obter mais informações sobre os campos no objeto da solicitação, consulte [Campos no objeto de solicitação](#). As listas a seguir descrevem os campos nos subobjetos `config` e `response`.

Campos no objeto de configuração

A lista a seguir descreve os campos no objeto `config` (`Records.cf.config`).

distributionDomainName (somente leitura)

O nome de domínio da distribuição associada à resposta.

distributionID (somente leitura)

O ID da distribuição associada à resposta.

eventType (somente leitura)

O tipo de acionador associado à resposta: `origin-response` ou `viewer-response`.

requestId (somente leitura)

Uma string criptografada que identifica exclusivamente a solicitação do visualizador ao CloudFront à qual esta resposta está associada. O valor de `requestId` também aparece nos logs de acesso do CloudFront como `x-edge-request-id`. Para ter mais informações, consulte [Configurar e usar logs padrão \(logs de acesso\)](#) e [Campos padrão de arquivo de log](#).

Campos no objeto de resposta

A lista a seguir descreve os campos no objeto `response` (`Records.cf.response`). Para obter informações sobre como usar uma função do `Lambda@Edge` para gerar uma resposta HTTP, consulte [Gerar respostas de HTTP em acionadores da solicitação](#).

headers (leitura/gravação)

Os cabeçalhos na resposta. Observe o seguinte:

- As chaves no objeto `headers` são versões em letras minúsculas de nomes de cabeçalho HTTP padrão. Usar chaves em letras minúsculas dá acesso sem diferenciar letras maiúsculas e minúsculas dos valores de cabeçalho.
- Cada objeto de cabeçalho (por exemplo, `headers["content-type"]` ou `headers["content-length"]`) é uma matriz de pares de chave-valor. Para determinado cabeçalho, a matriz contém um par de chave-valor para cada valor na resposta.
- `key` contém o nome do cabeçalho com distinção entre maiúsculas e minúsculas conforme aparece na resposta HTTP; por exemplo, `Content-Type`, `Content-Length`, `Cookie`, e assim por diante.
- `value` contém o valor do cabeçalho como ele aparece na resposta HTTP.
- Quando a função do Lambda adicionar ou modificar cabeçalhos de resposta e você não incluir o campo `key` do cabeçalho, o `Lambda@Edge` inserirá automaticamente uma `key` de cabeçalho usando o nome de cabeçalho que você fornecer. Independentemente de como você tiver formatado o nome do cabeçalho, a chave de cabeçalho inserida automaticamente será formatada com inicial maiúscula para cada parte, separada por hifens (-).

Por exemplo, você pode adicionar um cabeçalho como o seguinte, sem um `key` de cabeçalho:

```
"content-type": [  
  {  
    "value": "text/html;charset=UTF-8"  
  }  
]
```

Neste exemplo, o `Lambda@Edge` insere automaticamente `"key": "Content-Type"`.

Para obter informações sobre as restrições de uso do cabeçalho, consulte [Restrições das funções de borda](#).

status

O código de status HTTP da resposta.

statusDescription

A descrição do status HTTP da resposta.

Trabalho com solicitações e respostas

Para usar as solicitações e as respostas do Lambda@Edge, consulte os seguintes tópicos:

Tópicos

- [Usar funções do Lambda@Edge com failover de origem](#)
- [Gerar respostas de HTTP em acionadores da solicitação](#)
- [Atualizar respostas de HTTP em acionadores de resposta da origem](#)
- [Acessar o corpo da solicitação escolhendo a opção de incluir o corpo](#)

Usar funções do Lambda@Edge com failover de origem

É possível usar as funções do Lambda@Edge com distribuições do CloudFront que você configurou com grupos de origens, por exemplo, para um failover de origem que você configura para ajudar a garantir a alta disponibilidade. Para usar uma função do Lambda com um grupo de origem, especifique a função em um trigger de resposta ou de solicitação de origem para um grupo de origens ao criar o comportamento de cache.

Para obter mais informações, consulte as informações a seguir.

- Criar grupos de origens: [Criar um grupo de origem](#)
- Como um failover de origem funciona com o Lambda@Edge: [Uso do failover de origem com funções do Lambda@Edge](#)

Gerar respostas de HTTP em acionadores da solicitação

Quando o CloudFront recebe uma solicitação, você pode usar uma função do Lambda para gerar uma resposta HTTP que o CloudFront retornará diretamente ao visualizador sem encaminhar a resposta para a origem. Gerar respostas HTTP reduz a carga na origem e geralmente também reduz a latência para o visualizador.

Alguns cenários comuns para gerar respostas HTTP incluem o seguinte:

- Retornar uma pequena página da web ao visualizador
- Retornar um código de status HTTP 301 ou 302 para redirecionar o usuário para outra página da web

- Retornar um código de status HTTP 401 para o visualizador quando o usuário ainda não tiver sido autenticado

Uma função do Lambda@Edge pode gerar uma resposta HTTP quando ocorrerem os seguintes eventos do CloudFront:

Eventos de solicitação de visualizador

Quando uma função for acionada por um evento de solicitação do visualizador, o CloudFront retornará a resposta ao visualizador e não a armazenará em cache.

Eventos de solicitação de origem

Quando a função for acionada por um evento de solicitação de origem, o CloudFront verificará se há uma resposta gerada anteriormente pela função no cache de borda.

- Se a resposta estiver no cache, a função não será executada e o CloudFront retornará a resposta em cache ao visualizador.
- Se a resposta não estiver no cache, a função será executada, o CloudFront retornará a resposta ao visualizador e também a armazenará em cache.

Para ver códigos de exemplo para gerar respostas HTTP, consulte [Funções de exemplo do Lambda@Edge](#). Você também pode substituir as respostas HTTP em triggers de resposta. Para ter mais informações, consulte [Atualizar respostas de HTTP em acionadores de resposta da origem](#).

Modelo de programação

Esta seção descreve o modelo de programação para usar o Lambda@Edge para gerar respostas HTTP.

Tópicos

- [Objeto da resposta](#)
- [Erros](#)
- [Campos obrigatórios](#)

Objeto da resposta

A resposta que você retornar como o parâmetro `result` do método `callback` deverá ter a seguinte estrutura (observe que apenas o campo `status` é necessário).

```
const response = {
  body: 'content',
  bodyEncoding: 'text' | 'base64',
  headers: {
    'header name in lowercase': [{
      key: 'header name in standard case',
      value: 'header value'
    }],
    ...
  },
  status: 'HTTP status code (string)',
  statusDescription: 'status description'
};
```

O objeto de resposta pode incluir os seguintes valores:

body

O corpo, se houver, que você deseja que o CloudFront retorne na resposta gerada.

bodyEncoding

A codificação para o valor que você especificou no `body`. As únicas codificações válidas são `text` e `base64`. Se você incluir `body` no objeto `response`, mas omitir `bodyEncoding`, o CloudFront tratará o corpo como texto.

Se você especificar `bodyEncoding` como `base64`, mas o corpo não for um `base64` válido, o CloudFront retornará um erro.

headers

Cabeçalhos que você deseja que o CloudFront retorne na resposta gerada. Observe o seguinte:

- As chaves no objeto `headers` são versões em letras minúsculas de nomes de cabeçalho HTTP padrão. Usar chaves em letras minúsculas dá acesso sem diferenciar letras maiúsculas e minúsculas dos valores de cabeçalho.
- Cada cabeçalho (por exemplo, `headers["accept"]` ou `headers["host"]`) é uma matriz de pares de chave-valor. Para determinado cabeçalho, a matriz contém um par de chave-valor para cada valor na resposta gerada.
- `key` (opcional) é o nome com distinção entre letras maiúsculas e minúsculas do cabeçalho da forma como ele aparece em uma solicitação HTTP, por exemplo, `accept` ou `host`.
- Especifique `value` como valor do cabeçalho.

- Se você não incluir a parte da chave do par de chave-valor do cabeçalho, o Lambda@Edge inserirá automaticamente uma chave de cabeçalho usando o nome de cabeçalho fornecido. Independentemente de como você tiver formatado o nome do cabeçalho, a chave de cabeçalho inserida será automaticamente formatada com o uso de iniciais maiúsculas para cada parte, separada por hífen (-).

Por exemplo, você pode adicionar um cabeçalho como o seguinte, sem uma chave de cabeçalho: `'content-type': [{ value: 'text/html;charset=UTF-8' }]`

Neste exemplo, o Lambda @ Edge cria a seguinte chave de cabeçalho: Content-Type.

Para obter informações sobre as restrições de uso do cabeçalho, consulte [Restrições das funções de borda](#).

status

Código de status do HTTP. Forneça o código de status como uma string. O CloudFront usa o código de status fornecido para o seguinte:

- Retornar na resposta
- Armazenar no cache de borda do CloudFront, quando a resposta tiver sido gerada por uma função acionada por um evento de solicitação de origem
- Fazer login no CloudFront [Configurar e usar logs padrão \(logs de acesso\)](#)

Se o valor de `status` não estiver entre 200 e 599, o CloudFront retornará um erro ao visualizador.

statusDescription

A descrição que você deseja que o CloudFront retorne na resposta para acompanhar o código de status HTTP. Você não precisa usar descrições padrão, como OK, para um código de status HTTP de 200.

Erros

Veja a seguir os erros possíveis das respostas HTTP geradas.

A resposta contém um corpo e especifica 204 (sem conteúdo) como status

Quando uma função for acionada por uma solicitação de visualizador, o CloudFront retornará um código de status HTTP 502 (gateway inválido) ao visualizador quando ocorrer o seguinte:

- O valor de status for 204 (sem conteúdo)
- A resposta incluir um valor para body

Isso ocorre porque o Lambda@Edge impõe a restrição opcional encontrada na RFC 2616: uma resposta HTTP 204 não precisa conter um corpo de mensagem.

Restrições ao tamanho da resposta gerada

O tamanho máximo de uma resposta gerada por uma função do Lambda depende do evento que acionou a função:

- Eventos de solicitação de visualizador: 40 KB
- Eventos de solicitação de origem: 1 MB

Se a resposta for maior que o tamanho permitido, o CloudFront retornará um código de status HTTP 502 (gateway inválido) ao visualizador.

Campos obrigatórios

O campo status é obrigatório.

Todos os demais campos são opcionais.

Atualizar respostas de HTTP em acionadores de resposta da origem

Quando o CloudFront recebe uma resposta HTTP do servidor de origem, se houver um trigger de resposta de origem associado ao comportamento de cache, você poderá modificar a resposta HTTP para substituir o que foi retornado pela origem.

Alguns cenários comuns para atualizar respostas HTTP incluem o seguinte:

- Alterar o status para definir um código de status HTTP 200 e a criação de conteúdo estático o corpo para retornar ao visualizador quando uma origem retornar um código de status de erro (4xx ou 5xx). Para obter o código de exemplo, consulte [Exemplo: usar um acionador de resposta de origem para atualizar o código do status de erro para 200](#).
- Alterar o status para definir um código de status HTTP 301 ou 302, de forma a redirecionar o usuário para outro site quando uma origem retornar um código de status de erro (4xx ou 5xx). Para obter o código de exemplo, consulte [Exemplo: usar um acionador de resposta de origem para atualizar o código do status de erro para 302](#).

Note

A função deve retornar um valor de status entre 200 e 599 (inclusive); do contrário, o CloudFront retornará um erro ao visualizador.

Você também pode substituir as respostas HTTP em eventos de solicitação do visualizador e da origem. Para ter mais informações, consulte [Gerar respostas de HTTP em acionadores da solicitação](#).

Ao trabalhar com a resposta de HTTP, o Lambda@Edge não expõe o corpo retornado pelo servidor de origem ao acionador da resposta de origem. Você pode gerar um corpo de conteúdo estático definindo-o no valor desejado ou removendo o corpo de dentro da função ao definir o valor como vazio. Se você não atualizar o campo do corpo na função, o corpo original retornado pelo servidor de origem será reapresentado ao visualizador.

Acessar o corpo da solicitação escolhendo a opção de incluir o corpo

Você pode optar por fazer com que o Lambda@Edge exponha o corpo em uma solicitação para métodos HTTP graváveis (POST, PUT, DELETE e assim por diante), para que seja possível acessá-lo na função do Lambda. Você pode escolher acesso somente leitura ou especificar que substituirá o corpo.

Para ativar essa opção, escolha Include Body (Incluir corpo) ao criar um trigger do CloudFront para sua função que seja para um evento de solicitação de visualizador ou de origem. Para obter mais informações, consulte [Adicionar acionadores para uma função do Lambda@Edge](#) ou, para saber como usar a opção Include Body (Incluir corpo) com a função, consulte [Estrutura de eventos do Lambda@Edge](#).

Os cenários em que você pode querer usar esse recurso incluem:

- Processamento de formulários da web, como formulários "entre em contato conosco", sem enviar dados de entrada do cliente de volta aos servidores de origem.
- Reunir dados de web beacons enviados por navegadores dos visualizadores e processá-los no ponto.

Para obter o código de exemplo, consulte [Funções de exemplo do Lambda@Edge](#).

Note

Se o corpo da solicitação for grande, o Lambda@Edge o trunará. Para obter informações detalhadas sobre truncamento e tamanho máximo, consulte [Restrições do corpo da solicitação com a opção de incluir corpo](#).

Funções de exemplo do Lambda@Edge

Veja os exemplos a seguir de como usar as funções do Lambda com o Amazon CloudFront.

Note

Se você escolher o runtime Node.js 18 ou posterior para a função Lambda@Edge, um arquivo `index.mjs` será criado automaticamente para você. Para usar os exemplos de código a seguir, renomeie o arquivo `index.mjs` para `index.js`.

Tópicos

- [Exemplos gerais](#)
- [Gerar respostas: exemplos](#)
- [Strings de consulta: exemplos](#)
- [Personalizar o conteúdo por cabeçalhos de país ou tipo de dispositivo: exemplos](#)
- [Seleção de origem dinâmica baseada em conteúdo: exemplos](#)
- [Atualização do status de erro: exemplos](#)
- [Acesso ao corpo da solicitação: exemplos](#)

Exemplos gerais

Os exemplos a seguir mostram as maneiras comuns de usar o Lambda@Edge no CloudFront.

Tópicos

- [Exemplo: testes A/B](#)
- [Exemplo: substituir um cabeçalho de resposta](#)

Exemplo: testes A/B

Você pode usar o exemplo a seguir para testar duas versões diferentes de uma imagem sem criar redirecionamentos nem alterar a URL. Este exemplo lê os cookies na solicitação do visualizador e modifica a URL da solicitação adequadamente. Se o visualizador não enviar um cookie com um dos valores esperados, o exemplo atribuirá aleatoriamente o visualizador a um dos URLs.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
  const request = event.Records[0].cf.request;
  const headers = request.headers;

  if (request.uri !== '/experiment-pixel.jpg') {
    // do not process if this is not an A-B test request
    callback(null, request);
    return;
  }

  const cookieExperimentA = 'X-Experiment-Name=A';
  const cookieExperimentB = 'X-Experiment-Name=B';
  const pathExperimentA = '/experiment-group/control-pixel.jpg';
  const pathExperimentB = '/experiment-group/treatment-pixel.jpg';

  /*
   * Lambda at the Edge headers are array objects.
   *
   * Client may send multiple Cookie headers, i.e.:
   * > GET /viewerRes/test HTTP/1.1
   * > User-Agent: curl/7.18.1 (x86_64-unknown-linux-gnu) libcurl/7.18.1
   * > Cookie: First=1; Second=2
   * > Cookie: ClientCode=abc
   * > Host: example.com
   *
   * You can access the first Cookie header at headers["cookie"][0].value
   * and the second at headers["cookie"][1].value.
   *
   * Header values are not parsed. In the example above,
   * headers["cookie"][0].value is equal to "First=1; Second=2"
   */
}
```

```

let experimentUri;
if (headers.cookie) {
  for (let i = 0; i < headers.cookie.length; i++) {
    if (headers.cookie[i].value.indexOf(cookieExperimentA) >= 0) {
      console.log('Experiment A cookie found');
      experimentUri = pathExperimentA;
      break;
    } else if (headers.cookie[i].value.indexOf(cookieExperimentB) >= 0) {
      console.log('Experiment B cookie found');
      experimentUri = pathExperimentB;
      break;
    }
  }
}

if (!experimentUri) {
  console.log('Experiment cookie has not been found. Throwing dice...');
  if (Math.random() < 0.75) {
    experimentUri = pathExperimentA;
  } else {
    experimentUri = pathExperimentB;
  }
}

request.uri = experimentUri;
console.log(`Request uri set to "${request.uri}"`);
callback(null, request);
};

```

Python

```

import json
import random

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    headers = request['headers']

    if request['uri'] != '/experiment-pixel.jpg':
        # Not an A/B Test
        return request

```

```
    cookieExperimentA, cookieExperimentB = 'X-Experiment-Name=A', 'X-Experiment-Name=B'
    pathExperimentA, pathExperimentB = '/experiment-group/control-pixel.jpg', '/experiment-group/treatment-pixel.jpg'
```

```
    ...
```

Lambda at the Edge headers are array objects.

Client may send multiple cookie headers. For example:

```
> GET /viewerRes/test HTTP/1.1
> User-Agent: curl/7.18.1 (x86_64-unknown-linux-gnu) libcurl/7.18.1
OpenSSL/1.0.1u zlib/1.2.3
> Cookie: First=1; Second=2
> Cookie: ClientCode=abc
> Host: example.com
```

You can access the first Cookie header at `headers["cookie"][0].value` and the second at `headers["cookie"][1].value`.

Header values are not parsed. In the example above, `headers["cookie"][0].value` is equal to `"First=1; Second=2"`

```
    ...
```

```
    experimentUri = ""
```

```
    for cookie in headers.get('cookie', []):
        if cookieExperimentA in cookie['value']:
            print("Experiment A cookie found")
            experimentUri = pathExperimentA
            break
        elif cookieExperimentB in cookie['value']:
            print("Experiment B cookie found")
            experimentUri = pathExperimentB
            break

    if not experimentUri:
        print("Experiment cookie has not been found. Throwing dice...")
        if random.random() < 0.75:
            experimentUri = pathExperimentA
        else:
            experimentUri = pathExperimentB

    request['uri'] = experimentUri
    print(f"Request uri set to {experimentUri}")
```

```
return request
```

Exemplo: substituir um cabeçalho de resposta

O exemplo abaixo mostra como alterar o valor de um cabeçalho de resposta com base no valor de outro cabeçalho.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
  const response = event.Records[0].cf.response;
  const headers = response.headers;

  const headerNameSrc = 'X-Amz-Meta-Last-Modified';
  const headerNameDst = 'Last-Modified';

  if (headers[headerNameSrc.toLowerCase()]) {
    headers[headerNameDst.toLowerCase()] = [
      headers[headerNameSrc.toLowerCase()][0],
    ];
    console.log(`Response header "${headerNameDst}" was set to ` +
      `${headers[headerNameDst.toLowerCase()][0].value}`);
  }

  callback(null, response);
};
```

Python

```
import json

def lambda_handler(event, context):
    response = event["Records"][0]["cf"]["response"]
    headers = response["headers"]

    headerNameSrc = "X-Amz-Meta-Last-Modified"
    headerNameDst = "Last-Modified"

    if headers.get(headerNameSrc.lower(), None):
        headers[headerNameDst.lower()] = [headers[headerNameSrc.lower()][0]]
```

```
print(f"Response header {headerNameDst.lower()} was set to
{headers[headerNameSrc.lower()][0]}")

return response
```

Gerar respostas: exemplos

Os exemplos a seguir mostram como você pode usar o Lambda@Edge para gerar respostas.

Tópicos

- [Exemplo: fornecer conteúdo estático \(resposta gerada\)](#)
- [Exemplo: gerar um redirecionamento de HTTP \(resposta gerada\)](#)

Exemplo: fornecer conteúdo estático (resposta gerada)

O exemplo a seguir mostra como usar uma função do Lambda para fornecer conteúdo estático de site. Isso reduz a carga no servidor de origem e a latência geral.

Note

Você pode gerar respostas HTTP para eventos de solicitação de visualizador e de solicitação de origem. Para ter mais informações, consulte [the section called “Gerar respostas de HTTP em acionadores da solicitação”](#).

Você também pode substituir ou remover o corpo da resposta de HTTP em eventos de resposta de origem. Para ter mais informações, consulte [the section called “Atualizar respostas de HTTP em acionadores de resposta da origem”](#).

Node.js

```
'use strict';

const content = `
<\!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Simple Lambda@Edge Static Content Response</title>
  </head>
```

```
<body>
  <p>Hello from Lambda@Edge!</p>
</body>
</html>
`;

exports.handler = (event, context, callback) => {
  /*
   * Generate HTTP OK response using 200 status code with HTML body.
   */
  const response = {
    status: '200',
    statusDescription: 'OK',
    headers: {
      'cache-control': [{
        key: 'Cache-Control',
        value: 'max-age=100'
      }],
      'content-type': [{
        key: 'Content-Type',
        value: 'text/html'
      }]
    },
    body: content,
  };
  callback(null, response);
};
```

Python

```
import json

CONTENT = """
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Simple Lambda@Edge Static Content Response</title>
</head>
<body>
  <p>Hello from Lambda@Edge!</p>
</body>
</html>
"""
```



```
"""  
  
def lambda_handler(event, context):  
    # Generate HTTP OK response using 200 status code with HTML body.  
    response = {  
        'status': '200',  
        'statusDescription': 'OK',  
        'headers': {  
            'cache-control': [  
                {  
                    'key': 'Cache-Control',  
                    'value': 'max-age=100'  
                }  
            ],  
            "content-type": [  
                {  
                    'key': 'Content-Type',  
                    'value': 'text/html'  
                }  
            ]  
        },  
        'body': CONTENT  
    }  
    return response
```

Exemplo: gerar um redirecionamento de HTTP (resposta gerada)

O exemplo abaixo mostra como gerar um redirecionamento HTTP.

Note

Você pode gerar respostas HTTP para eventos de solicitação de visualizador e de solicitação de origem. Para ter mais informações, consulte [Gerar respostas de HTTP em acionadores da solicitação](#).

Node.js

```
'use strict';  
  
exports.handler = (event, context, callback) => {
```

```
/*
 * Generate HTTP redirect response with 302 status code and Location header.
 */
const response = {
  status: '302',
  statusDescription: 'Found',
  headers: {
    location: [{
      key: 'Location',
      value: 'https://docs.aws.amazon.com/lambda/latest/dg/lambda-
edge.html',
    }],
  },
};
callback(null, response);
};
```

Python

```
def lambda_handler(event, context):

    # Generate HTTP redirect response with 302 status code and Location header.

    response = {
        'status': '302',
        'statusDescription': 'Found',
        'headers': {
            'location': [{
                'key': 'Location',
                'value': 'https://docs.aws.amazon.com/lambda/latest/dg/lambda-
edge.html'
            }]
        }
    }

    return response
```

Strings de consulta: exemplos

Os exemplos a seguir mostram maneiras de usar o Lambda@Edge com strings de consulta.

Tópicos

- [Exemplo: adicionar um cabeçalho com base em um parâmetro de string de consulta](#)
- [Exemplo: normalizar os parâmetros da string de consulta para melhorar a proporção de acertos no cache](#)
- [Exemplo: redirecionar usuários não autenticados para uma página de login](#)

Exemplo: adicionar um cabeçalho com base em um parâmetro de string de consulta

O exemplo a seguir mostra como obter o par chave-valor de um parâmetro de string de consulta e adicionar um cabeçalho com base nesses valores.

Node.js

```
'use strict';

const querystring = require('querystring');
exports.handler = (event, context, callback) => {
  const request = event.Records[0].cf.request;

  /* When a request contains a query string key-value pair but the origin server
   * expects the value in a header, you can use this Lambda function to
   * convert the key-value pair to a header. Here's what the function does:
   * 1. Parses the query string and gets the key-value pair.
   * 2. Adds a header to the request using the key-value pair that the function
   got in step 1.
   */

  /* Parse request querystring to get javascript object */
  const params = querystring.parse(request.querystring);

  /* Move auth param from querystring to headers */
  const headerName = 'Auth-Header';
  request.headers[headerName.toLowerCase()] = [{ key: headerName, value:
params.auth }];
  delete params.auth;

  /* Update request querystring */
  request.querystring = querystring.stringify(params);

  callback(null, request);
};
```

Python

```
from urllib.parse import parse_qs, urlencode

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    ...

    When a request contains a query string key-value pair but the origin server
    expects the value in a header, you can use this Lambda function to
    convert the key-value pair to a header. Here's what the function does:
        1. Parses the query string and gets the key-value pair.
        2. Adds a header to the request using the key-value pair that the function
    got in step 1.
    ...

    # Parse request querystring to get dictionary/json
    params = {k : v[0] for k, v in parse_qs(request['querystring']).items()}

    # Move auth param from querystring to headers
    headerName = 'Auth-Header'
    request['headers'][headerName.lower()] = [{'key': headerName, 'value':
params['auth']}]
    del params['auth']

    # Update request querystring
    request['querystring'] = urlencode(params)

    return request
```

Exemplo: normalizar os parâmetros da string de consulta para melhorar a proporção de acertos no cache

O exemplo a seguir mostra como melhorar o índice de ocorrência no cache fazendo as seguintes alterações nas strings de consulta antes que o CloudFront encaminhe as solicitações para a origem:

- Colocar os pares de chave-valor em ordem alfabética pelo nome do parâmetro.
- Alterar os pares de chave-valor para minúsculas.

Para ter mais informações, consulte [Conteúdo em cache com base em parâmetros de string de consulta](#).

Node.js

```
'use strict';

const querystring = require('querystring');

exports.handler = (event, context, callback) => {
  const request = event.Records[0].cf.request;
  /* When you configure a distribution to forward query strings to the origin and
   * to cache based on an allowlist of query string parameters, we recommend
   * the following to improve the cache-hit ratio:
   * - Always list parameters in the same order.
   * - Use the same case for parameter names and values.
   *
   * This function normalizes query strings so that parameter names and values
   * are lowercase and parameter names are in alphabetical order.
   *
   * For more information, see:
   * https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/
  QueryStringParameters.html
   */

  console.log('Query String: ', request.querystring);

  /* Parse request query string to get javascript object */
  const params = querystring.parse(request.querystring.toLowerCase());
  const sortedParams = {};

  /* Sort param keys */
  Object.keys(params).sort().forEach(key => {
    sortedParams[key] = params[key];
  });

  /* Update request querystring with normalized */
  request.querystring = querystring.stringify(sortedParams);

  callback(null, request);
};
```

Python

```
from urllib.parse import parse_qs, urlencode
```

```
def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    ...

    When you configure a distribution to forward query strings to the origin and
    to cache based on an allowlist of query string parameters, we recommend
    the following to improve the cache-hit ratio:
    Always list parameters in the same order.
    - Use the same case for parameter names and values.

    This function normalizes query strings so that parameter names and values
    are lowercase and parameter names are in alphabetical order.

    For more information, see:
    https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/
    QueryStringParameters.html
    ...
    print("Query string: ", request["querystring"])

    # Parse request query string to get js object
    params = {k : v[0] for k, v in parse_qs(request['querystring'].lower()).items()}

    # Sort param keys
    sortedParams = sorted(params.items(), key=lambda x: x[0])

    # Update request querystring with normalized
    request['querystring'] = urlencode(sortedParams)

    return request
```

Exemplo: redirecionar usuários não autenticados para uma página de login

O exemplo a seguir mostra como redirecionar os usuários para uma página de login caso não tenham inserido as credenciais.

Node.js

```
'use strict';

function parseCookies(headers) {
    const parsedCookie = {};
    if (headers.cookie) {
        headers.cookie[0].value.split(';').forEach((cookie) => {
```

```
        if (cookie) {
            const parts = cookie.split('=');
            parsedCookie[parts[0].trim()] = parts[1].trim();
        }
    });
}
return parsedCookie;
}

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const headers = request.headers;

    /* Check for session-id in request cookie in viewer-request event,
     * if session-id is absent, redirect the user to sign in page with original
     * request sent as redirect_url in query params.
     */

    /* Check for session-id in cookie, if present then proceed with request */
    const parsedCookies = parseCookies(headers);
    if (parsedCookies && parsedCookies['session-id']) {
        callback(null, request);
        return;
    }

    /* URI encode the original request to be sent as redirect_url in query params */
    const encodedRedirectUrl = encodeURIComponent(`https://${headers.host[0].value}${request.uri}?${request.querystring}`);
    const response = {
        status: '302',
        statusDescription: 'Found',
        headers: {
            location: [{
                key: 'Location',
                value: `https://www.example.com/signin?redirect_url=${encodedRedirectUrl}`,
            }],
        },
    };
    callback(null, response);
};
```

Python

```
import urllib

def parseCookies(headers):
    parsedCookie = {}
    if headers.get('cookie'):
        for cookie in headers['cookie'][0]['value'].split(';'):
            if cookie:
                parts = cookie.split('=')
                parsedCookie[parts[0].strip()] = parts[1].strip()
    return parsedCookie

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    headers = request['headers']

    """
    Check for session-id in request cookie in viewer-request event,
    if session-id is absent, redirect the user to sign in page with original
    request sent as redirect_url in query params.
    """

    # Check for session-id in cookie, if present, then proceed with request
    parsedCookies = parseCookies(headers)

    if parsedCookies and parsedCookies['session-id']:
        return request

    # URI encode the original request to be sent as redirect_url in query params
    redirectUrl = "https://%s%s?%s" % (headers['host'][0]['value'], request['uri'],
    request['querystring'])
    encodedRedirectUrl = urllib.parse.quote_plus(redirectUrl.encode('utf-8'))

    response = {
        'status': '302',
        'statusDescription': 'Found',
        'headers': {
            'location': [{
                'key': 'Location',
                'value': 'https://www.example.com/signin?redirect_url=%s' %
    encodedRedirectUrl
            }]
        }
    }
```



```
}  
return response
```

Personalizar o conteúdo por cabeçalhos de país ou tipo de dispositivo: exemplos

Os exemplos a seguir mostram como você pode usar o Lambda@Edge para personalizar o comportamento com base no local ou no tipo de dispositivo usado pelo visualizador.

Tópicos

- [Exemplo: redirecionar as solicitações do visualizador para um URL específico do país](#)
- [Exemplo: fornecer diferentes versões de um objeto com base no dispositivo](#)

Exemplo: redirecionar as solicitações do visualizador para um URL específico do país

O exemplo a seguir mostra como gerar uma resposta de redirecionamento HTTP com um URL específico do país e retornar a resposta para o visualizador. Isso é útil quando você deseja fornecer respostas específicas do país. Por exemplo:

- Se tiver subdomínios específicos do país, como `us.example.com` e `tw.example.com`, você pode gerar uma resposta de redirecionamento quando um visualizador solicitar `example.com`.
- Se estiver transmitindo um vídeo, mas não tem direitos para transmitir o conteúdo em um país específico, você pode redirecionar os usuários desse país para uma página que explica por que eles não podem visualizar o vídeo.

Observe o seguinte:

- É necessário configurar a distribuição para armazenar em cache com base no cabeçalho `CloudFront-Viewer-Country`. Para ter mais informações, consulte [Cache baseado em Cabeçalhos de solicitação selecionados](#).
- O CloudFront adiciona o cabeçalho `CloudFront-Viewer-Country` após o evento de solicitação do visualizador. Para usar este exemplo, é necessário criar um trigger para o evento de solicitação da origem.

Node.js

```
'use strict';
```

```
/* This is an origin request function */
exports.handler = (event, context, callback) => {
  const request = event.Records[0].cf.request;
  const headers = request.headers;

  /*
   * Based on the value of the CloudFront-Viewer-Country header, generate an
   * HTTP status code 302 (Redirect) response, and return a country-specific
   * URL in the Location header.
   * NOTE: 1. You must configure your distribution to cache based on the
   *         CloudFront-Viewer-Country header. For more information, see
   *         https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-
headers
   *         2. CloudFront adds the CloudFront-Viewer-Country header after the
viewer
   *         request event. To use this example, you must create a trigger for
the
   *         origin request event.
   */

  let url = 'https://example.com/';
  if (headers['cloudfront-viewer-country']) {
    const countryCode = headers['cloudfront-viewer-country'][0].value;
    if (countryCode === 'TW') {
      url = 'https://tw.example.com/';
    } else if (countryCode === 'US') {
      url = 'https://us.example.com/';
    }
  }
}

const response = {
  status: '302',
  statusDescription: 'Found',
  headers: {
    location: [{
      key: 'Location',
      value: url,
    }],
  },
};
callback(null, response);
};
```

Python

```
# This is an origin request function

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    headers = request['headers']

    ...

    Based on the value of the CloudFront-Viewer-Country header, generate an
    HTTP status code 302 (Redirect) response, and return a country-specific
    URL in the Location header.
    NOTE: 1. You must configure your distribution to cache based on the
           CloudFront-Viewer-Country header. For more information, see
           https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
          2. CloudFront adds the CloudFront-Viewer-Country header after the viewer
           request event. To use this example, you must create a trigger for the
           origin request event.

    ...

    url = 'https://example.com/'
    viewerCountry = headers.get('cloudfront-viewer-country')
    if viewerCountry:
        countryCode = viewerCountry[0]['value']
        if countryCode == 'TW':
            url = 'https://tw.example.com/'
        elif countryCode == 'US':
            url = 'https://us.example.com/'

    response = {
        'status': '302',
        'statusDescription': 'Found',
        'headers': {
            'location': [{
                'key': 'Location',
                'value': url
            }]
        }
    }

    return response
```

Exemplo: fornecer diferentes versões de um objeto com base no dispositivo

O exemplo a seguir mostra como servir diferentes versões de um objeto com base no tipo de dispositivo usado pelo usuário, por exemplo, um dispositivo móvel ou um tablet. Observe o seguinte:

- É necessário configurar a distribuição para armazenar em cache com base nos cabeçalhos CloudFront-Is-*-Viewer. Para ter mais informações, consulte [Cache baseado em Cabeçalhos de solicitação selecionados](#).
- O CloudFront adiciona os cabeçalhos CloudFront-Is-*-Viewer após o evento de solicitação do visualizador. Para usar este exemplo, é necessário criar um trigger para o evento de solicitação da origem.

Node.js

```
'use strict';

/* This is an origin request function */
exports.handler = (event, context, callback) => {
  const request = event.Records[0].cf.request;
  const headers = request.headers;

  /*
   * Serve different versions of an object based on the device type.
   * NOTE: 1. You must configure your distribution to cache based on the
   *        CloudFront-Is-*-Viewer headers. For more information, see
   *        the following documentation:
   *        https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-
headers
   *        https://docs.aws.amazon.com/console/cloudfront/cache-on-device-type
   *        2. CloudFront adds the CloudFront-Is-*-Viewer headers after the viewer
   *        request event. To use this example, you must create a trigger for
the
   *        origin request event.
   */

  const desktopPath = '/desktop';
  const mobilePath = '/mobile';
  const tabletPath = '/tablet';
  const smarttvPath = '/smarttv';

  if (headers['cloudfront-is-desktop-viewer']
```

```

    && headers['cloudfront-is-desktop-viewer'][0].value === 'true') {
      request.uri = desktopPath + request.uri;
    } else if (headers['cloudfront-is-mobile-viewer']
      && headers['cloudfront-is-mobile-viewer'][0].value === 'true') {
      request.uri = mobilePath + request.uri;
    } else if (headers['cloudfront-is-tablet-viewer']
      && headers['cloudfront-is-tablet-viewer'][0].value === 'true') {
      request.uri = tabletPath + request.uri;
    } else if (headers['cloudfront-is-smarttv-viewer']
      && headers['cloudfront-is-smarttv-viewer'][0].value === 'true') {
      request.uri = smarttvPath + request.uri;
    }
    console.log(`Request uri set to "${request.uri}"`);

    callback(null, request);
  };

```

Python

```

# This is an origin request function
def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    headers = request['headers']

    ...

    Serve different versions of an object based on the device type.
    NOTE: 1. You must configure your distribution to cache based on the
    CloudFront-Is-*-Viewer headers. For more information, see
    the following documentation:
    https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
    https://docs.aws.amazon.com/console/cloudfront/cache-on-device-type
    2. CloudFront adds the CloudFront-Is-*-Viewer headers after the viewer
    request event. To use this example, you must create a trigger for the
    origin request event.

    ...

    desktopPath = '/desktop';
    mobilePath = '/mobile';
    tabletPath = '/tablet';
    smarttvPath = '/smarttv';

    if 'cloudfront-is-desktop-viewer' in headers and headers['cloudfront-is-desktop-
viewer'][0]['value'] == 'true':

```

```
    request['uri'] = desktopPath + request['uri']
    elif 'cloudfront-is-mobile-viewer' in headers and headers['cloudfront-is-mobile-viewer'][0]['value'] == 'true':
        request['uri'] = mobilePath + request['uri']
    elif 'cloudfront-is-tablet-viewer' in headers and headers['cloudfront-is-tablet-viewer'][0]['value'] == 'true':
        request['uri'] = tabletPath + request['uri']
    elif 'cloudfront-is-smarttv-viewer' in headers and headers['cloudfront-is-smarttv-viewer'][0]['value'] == 'true':
        request['uri'] = smarttvPath + request['uri']

    print("Request uri set to %s" % request['uri'])

    return request
```

Seleção de origem dinâmica baseada em conteúdo: exemplos

Os exemplos a seguir mostram como você pode usar o Lambda@Edge para rotear para diferentes origens com base em informações na solicitação.

Tópicos

- [Exemplo: usar um acionador de solicitação de origem para alterar de uma origem personalizada para uma origem do Amazon S3](#)
- [Exemplo: usar um acionador de solicitação de origem para alterar a região de origem do Amazon S3](#)
- [Exemplo: usar um acionador de solicitação de origem para alterar de uma origem do Amazon S3 para uma origem personalizada](#)
- [Exemplo: usar um acionador de solicitação de origem para transferir gradualmente o tráfego de um bucket do Amazon S3 para outro](#)
- [Exemplo: usar um acionador de solicitação de origem para alterar o nome do domínio de origem com base no cabeçalho do país](#)

Exemplo: usar um acionador de solicitação de origem para alterar de uma origem personalizada para uma origem do Amazon S3

Essa função demonstra como um trigger origin-request pode ser usado para alterar de uma origem personalizada para uma origem do Amazon S3 da qual o conteúdo é obtido, com base nas propriedades da solicitação.

Node.js

```
'use strict';

const querystring = require('querystring');

exports.handler = (event, context, callback) => {
  const request = event.Records[0].cf.request;

  /**
   * Reads query string to check if S3 origin should be used, and
   * if true, sets S3 origin properties.
   */

  const params = querystring.parse(request.querystring);

  if (params['useS3Origin']) {
    if (params['useS3Origin'] === 'true') {
      const s3DomainName = 'amzn-s3-demo-bucket.s3.amazonaws.com';

      /* Set S3 origin fields */
      request.origin = {
        s3: {
          domainName: s3DomainName,
          region: '',
          authMethod: 'none',
          path: '',
          customHeaders: {}
        }
      };
      request.headers['host'] = [{ key: 'host', value: s3DomainName}];
    }
  }

  callback(null, request);
};
```

Python

```
from urllib.parse import parse_qs

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
```

```
'''
Reads query string to check if S3 origin should be used, and
if true, sets S3 origin properties
'''
params = {k: v[0] for k, v in parse_qs(request['querystring']).items()}
if params.get('useS3Origin') == 'true':
    s3DomainName = 'amzn-s3-demo-bucket.s3.amazonaws.com'

# Set S3 origin fields
request['origin'] = {
    's3': {
        'domainName': s3DomainName,
        'region': '',
        'authMethod': 'none',
        'path': '',
        'customHeaders': {}
    }
}
request['headers']['host'] = [{'key': 'host', 'value': s3DomainName}]
return request
```

Exemplo: usar um acionador de solicitação de origem para alterar a região de origem do Amazon S3

Esta função demonstra como um trigger origin-request pode ser usado para alterar a origem do Amazon S3 da qual o conteúdo é obtido, com base nas propriedades da solicitação.

Neste exemplo, usamos o valor do cabeçalho CloudFront-Viewer-Country para atualizar o nome do domínio de bucket do S3 para um bucket em uma região mais próxima do visualizador. Isso pode ser útil de várias maneiras:

- Reduz as latências quando a região especificada estiver mais próxima do país do visualizador.
- Fornece soberania de dados, garantindo que os dados sejam oferecidos de uma origem que esteja no mesmo país de onde veio a solicitação.

Para usar esse exemplo, você precisa fazer o seguinte:

- Configure sua distribuição para armazenar em cache com base no cabeçalho CloudFront-Viewer-Country. Para ter mais informações, consulte [Cache baseado em Cabeçalhos de solicitação selecionados](#).

- Crie um gatilho para essa função no evento de solicitação de origem. O CloudFront adiciona o cabeçalho `CloudFront-Viewer-Country` após o evento de solicitação do visualizador. Portanto, para usar este exemplo, você precisa garantir que a função seja executada para uma solicitação de origem.

Note

O código de exemplo a seguir usa a mesma identidade do acesso de origem (OAI) para todos os buckets do S3 que você está usando para sua origem. Para ter mais informações, consulte [Visão geral da identidade do acesso de origem](#).

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
  const request = event.Records[0].cf.request;

  /**
   * This blueprint demonstrates how an origin-request trigger can be used to
   * change the origin from which the content is fetched, based on request
   properties.
   * In this example, we use the value of the CloudFront-Viewer-Country header
   * to update the S3 bucket domain name to a bucket in a Region that is closer to
   * the viewer.
   *
   * This can be useful in several ways:
   *   1) Reduces latencies when the Region specified is nearer to the viewer's
   *      country.
   *   2) Provides data sovereignty by making sure that data is served from an
   *      origin that's in the same country that the request came from.
   *
   * NOTE: 1. You must configure your distribution to cache based on the
   *         CloudFront-Viewer-Country header. For more information, see
   *         https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-
headers
   *         2. CloudFront adds the CloudFront-Viewer-Country header after the
viewer
   *         request event. To use this example, you must create a trigger for
the
```

```

*           origin request event.
*/

const countryToRegion = {
  'DE': 'eu-central-1',
  'IE': 'eu-west-1',
  'GB': 'eu-west-2',
  'FR': 'eu-west-3',
  'JP': 'ap-northeast-1',
  'IN': 'ap-south-1'
};

if (request.headers['cloudfront-viewer-country']) {
  const countryCode = request.headers['cloudfront-viewer-country'][0].value;
  const region = countryToRegion[countryCode];

  /**
   * If the viewer's country is not in the list you specify, the request
   * goes to the default S3 bucket you've configured.
   */
  if (region) {
    /**
     * If you've set up OAI, the bucket policy in the destination bucket
     * should allow the OAI GetObject operation, as configured by default
     * for an S3 origin with OAI. Another requirement with OAI is to provide
     * the Region so it can be used for the SIGV4 signature. Otherwise, the
     * Region is not required.
     */
    request.origin.s3.region = region;
    const domainName = `amzn-s3-demo-bucket-in-${region}.s3.
${region}.amazonaws.com`;
    request.origin.s3.domainName = domainName;
    request.headers['host'] = [{ key: 'host', value: domainName }];
  }
}

callback(null, request);
};

```

Python

```

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

```

```
...
```

This blueprint demonstrates how an origin-request trigger can be used to change the origin from which the content is fetched, based on request properties.

In this example, we use the value of the CloudFront-Viewer-Country header to update the S3 bucket domain name to a bucket in a Region that is closer to the viewer.

This can be useful in several ways:

- 1) Reduces latencies when the Region specified is nearer to the viewer's country.
- 2) Provides data sovereignty by making sure that data is served from an origin that's in the same country that the request came from.

NOTE: 1. You must configure your distribution to cache based on the CloudFront-Viewer-Country header. For more information, see <https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers>

2. CloudFront adds the CloudFront-Viewer-Country header after the viewer request event. To use this example, you must create a trigger for the origin request event.

```
...
```

```
countryToRegion = {  
  'DE': 'eu-central-1',  
  'IE': 'eu-west-1',  
  'GB': 'eu-west-2',  
  'FR': 'eu-west-3',  
  'JP': 'ap-northeast-1',  
  'IN': 'ap-south-1'  
}
```

```
viewerCountry = request['headers'].get('cloudfront-viewer-country')
```

```
if viewerCountry:
```

```
  countryCode = viewerCountry[0]['value']  
  region = countryToRegion.get(countryCode)
```

```
# If the viewer's country is not in the list you specify, the request  
# goes to the default S3 bucket you've configured
```

```
if region:
```

```
  ...
```

```
    If you've set up OAI, the bucket policy in the destination bucket  
    should allow the OAI GetObject operation, as configured by default  
    for an S3 origin with OAI. Another requirement with OAI is to provide
```

```

    the Region so it can be used for the SIGV4 signature. Otherwise, the
    Region is not required.
    ...
    request['origin']['s3']['region'] = region
    domainName = 'amzn-s3-demo-bucket-in-{0}.s3.
{0}.amazonaws.com'.format(region)
    request['origin']['s3']['domainName'] = domainName
    request['headers']['host'] = [{'key': 'host', 'value': domainName}]

return request

```

Exemplo: usar um acionador de solicitação de origem para alterar de uma origem do Amazon S3 para uma origem personalizada

Esta função demonstra como um trigger de origem-solicitação pode ser usado para alterar a origem personalizada de onde o conteúdo é obtido, com base nas propriedades da solicitação.

Node.js

```

'use strict';

const querystring = require('querystring');

exports.handler = (event, context, callback) => {
  const request = event.Records[0].cf.request;

  /**
   * Reads query string to check if custom origin should be used, and
   * if true, sets custom origin properties.
   */

  const params = querystring.parse(request.querystring);

  if (params['useCustomOrigin']) {
    if (params['useCustomOrigin'] === 'true') {

      /* Set custom origin fields*/
      request.origin = {
        custom: {
          domainName: 'www.example.com',
          port: 443,
          protocol: 'https',

```

```
        path: '',
        sslProtocols: ['TLSv1', 'TLSv1.1'],
        readTimeout: 5,
        keepaliveTimeout: 5,
        customHeaders: {}
    }
};
request.headers['host'] = [{ key: 'host', value: 'www.example.com'}];
}
}
callback(null, request);
};
```

Python

```
from urllib.parse import parse_qs

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    # Reads query string to check if custom origin should be used, and
    # if true, sets custom origin properties

    params = {k: v[0] for k, v in parse_qs(request['queryString']).items()}

    if params.get('useCustomOrigin') == 'true':
        # Set custom origin fields
        request['origin'] = {
            'custom': {
                'domainName': 'www.example.com',
                'port': 443,
                'protocol': 'https',
                'path': '',
                'sslProtocols': ['TLSv1', 'TLSv1.1'],
                'readTimeout': 5,
                'keepaliveTimeout': 5,
                'customHeaders': {}
            }
        }
        request['headers']['host'] = [{'key': 'host', 'value':
'www.example.com'}]

    return request
```

Exemplo: usar um acionador de solicitação de origem para transferir gradualmente o tráfego de um bucket do Amazon S3 para outro

Essa função demonstra como transferir o tráfego de um bucket do Amazon S3 para outro de forma gradual e controlada.

Node.js

```
'use strict';

function getRandomInt(min, max) {
  /* Random number is inclusive of min and max*/
  return Math.floor(Math.random() * (max - min + 1)) + min;
}

exports.handler = (event, context, callback) => {
  const request = event.Records[0].cf.request;
  const BLUE_TRAFFIC_PERCENTAGE = 80;

  /**
   * This Lambda function demonstrates how to gradually transfer traffic from
   * one S3 bucket to another in a controlled way.
   * We define a variable BLUE_TRAFFIC_PERCENTAGE which can take values from
   * 1 to 100. If the generated randomNumber less than or equal to
  BLUE_TRAFFIC_PERCENTAGE, traffic
   * is re-directed to blue-bucket. If not, the default bucket that we've
  configured
   * is used.
   */

  const randomNumber = getRandomInt(1, 100);

  if (randomNumber <= BLUE_TRAFFIC_PERCENTAGE) {
    const domainName = 'blue-bucket.s3.amazonaws.com';
    request.origin.s3.domainName = domainName;
    request.headers['host'] = [{ key: 'host', value: domainName}];
  }
  callback(null, request);
};
```

Python

```
import math
```

```
import random

def getRandomInt(min, max):
    # Random number is inclusive of min and max
    return math.floor(random.random() * (max - min + 1)) + min

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    BLUE_TRAFFIC_PERCENTAGE = 80

    ...

    This Lambda function demonstrates how to gradually transfer traffic from
    one S3 bucket to another in a controlled way.
    We define a variable BLUE_TRAFFIC_PERCENTAGE which can take values from
    1 to 100. If the generated randomNumber less than or equal to
    BLUE_TRAFFIC_PERCENTAGE, traffic
    is re-directed to blue-bucket. If not, the default bucket that we've configured
    is used.
    ...

    randomNumber = getRandomInt(1, 100)

    if randomNumber <= BLUE_TRAFFIC_PERCENTAGE:
        domainName = 'blue-bucket.s3.amazonaws.com'
        request['origin']['s3']['domainName'] = domainName
        request['headers']['host'] = [{'key': 'host', 'value': domainName}]

    return request
```

Exemplo: usar um acionador de solicitação de origem para alterar o nome do domínio de origem com base no cabeçalho do país

Esta função demonstra como você pode alterar o nome de domínio de origem com base no cabeçalho `CloudFront-Viewer-Country`, de forma que o conteúdo seja fornecido de origem mais próxima do país do visualizador.

A implementação dessa funcionalidade para sua distribuição pode ter vantagens, como as seguintes:

- Redução das latências quando a região especificada estiver mais próxima do país do visualizador
- Fornecimento da soberania de dados garantindo que os dados sejam fornecidos de uma origem que esteja no mesmo país de onde veio a solicitação

Observe que, para habilitar essa funcionalidade, você deve configurar sua distribuição para o cache com base no cabeçalho `CloudFront-Viewer-Country`. Para ter mais informações, consulte [the section called “Cache baseado em Cabeçalhos de solicitação selecionados”](#).

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;

    if (request.headers['cloudfront-viewer-country']) {
        const countryCode = request.headers['cloudfront-viewer-country'][0].value;
        if (countryCode === 'GB' || countryCode === 'DE' || countryCode === 'IE' )
        {
            const domainName = 'eu.example.com';
            request.origin.custom.domainName = domainName;
            request.headers['host'] = [{key: 'host', value: domainName}];
        }
    }

    callback(null, request);
};
```

Python

```
def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    viewerCountry = request['headers'].get('cloudfront-viewer-country')
    if viewerCountry:
        countryCode = viewerCountry[0]['value']
        if countryCode == 'GB' or countryCode == 'DE' or countryCode == 'IE':
            domainName = 'eu.example.com'
            request['origin']['custom']['domainName'] = domainName
            request['headers']['host'] = [{'key': 'host', 'value': domainName}]
    return request
```


Atualização do status de erro: exemplos

Os exemplos a seguir fornecem orientações sobre como você pode usar o Lambda@Edge para alterar o status de erro retornado para os usuários.

Tópicos

- [Exemplo: usar um acionador de resposta de origem para atualizar o código do status de erro para 200](#)
- [Exemplo: usar um acionador de resposta de origem para atualizar o código do status de erro para 302](#)

Exemplo: usar um acionador de resposta de origem para atualizar o código do status de erro para 200

Esta função demonstra como você pode atualizar o status da resposta para 200 e gerar conteúdo do corpo estático para retornar ao visualizador no cenário a seguir:

- A função é acionada em uma resposta da origem.
- O status da resposta do servidor de origem é um código de status de erro (4xx ou 5xx).

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
  const response = event.Records[0].cf.response;

  /**
   * This function updates the response status to 200 and generates static
   * body content to return to the viewer in the following scenario:
   * 1. The function is triggered in an origin response
   * 2. The response status from the origin server is an error status code (4xx or
5xx)
   */

  if (response.status >= 400 && response.status <= 599) {
    response.status = 200;
    response.statusDescription = 'OK';
    response.body = 'Body generation example';
  }
}
```

```
    callback(null, response);  
};
```

Python

```
def lambda_handler(event, context):  
    response = event['Records'][0]['cf']['response']  
  
    '''  
    This function updates the response status to 200 and generates static  
    body content to return to the viewer in the following scenario:  
    1. The function is triggered in an origin response  
    2. The response status from the origin server is an error status code (4xx or  
    5xx)  
    '''  
  
    if int(response['status']) >= 400 and int(response['status']) <= 599:  
        response['status'] = 200  
        response['statusDescription'] = 'OK'  
        response['body'] = 'Body generation example'  
    return response
```

Exemplo: usar um acionador de resposta de origem para atualizar o código do status de erro para 302

Essa função demonstra como você pode atualizar o código de status HTTP para 302, de forma a redirecionar a outro caminho (comportamento de cache) que tem uma origem diferente configurada. Observe o seguinte:

- A função é acionada em uma resposta da origem.
- O status da resposta do servidor de origem é um código de status de erro (4xx ou 5xx).

Node.js

```
'use strict';  
  
exports.handler = (event, context, callback) => {  
    const response = event.Records[0].cf.response;  
    const request = event.Records[0].cf.request;
```

```

/**
 * This function updates the HTTP status code in the response to 302, to
redirect to another
 * path (cache behavior) that has a different origin configured. Note the
following:
 * 1. The function is triggered in an origin response
 * 2. The response status from the origin server is an error status code (4xx or
5xx)
 */

if (response.status >= 400 && response.status <= 599) {
  const redirect_path = `/plan-b/path?${request.querystring}`;

  response.status = 302;
  response.statusDescription = 'Found';

  /* Drop the body, as it is not required for redirects */
  response.body = '';
  response.headers['location'] = [{ key: 'Location', value: redirect_path }];
}

callback(null, response);
};

```

Python

```

def lambda_handler(event, context):
    response = event['Records'][0]['cf']['response']
    request = event['Records'][0]['cf']['request']

    ...

    This function updates the HTTP status code in the response to 302, to redirect
to another
    path (cache behavior) that has a different origin configured. Note the
following:
    1. The function is triggered in an origin response
    2. The response status from the origin server is an error status code (4xx or
5xx)
    ...

    if int(response['status']) >= 400 and int(response['status']) <= 599:
        redirect_path = '/plan-b/path?%s' % request['querystring']

```

```
response['status'] = 302
response['statusDescription'] = 'Found'

# Drop the body as it is not required for redirects
response['body'] = ''
response['headers']['location'] = [{'key': 'Location', 'value':
redirect_path}]

return response
```

Acesso ao corpo da solicitação: exemplos

Os exemplos a seguir mostram como você pode usar o Lambda@Edge para trabalhar com solicitações POST.

Note

Para usar esses exemplos, você deve habilitar a opção `include body` (incluir corpo) na associação da função do Lambda da distribuição. Ele não é habilitado por padrão.

- Para habilitar essa configuração no console do CloudFront, marque a caixa de seleção `Include Body (Incluir corpo)` na `Lambda Function Association` (Associação de função do Lambda).
- Para habilitar essa configuração na API do CloudFront ou com o AWS CloudFormation, defina o campo `IncludeBody` para `true` em `LambdaFunctionAssociation`.

Tópicos

- [Exemplo: usar um acionador de solicitação para ler um formulário HTML](#)
- [Exemplo: usar um acionador de solicitação para modificar um formulário HTML](#)

Exemplo: usar um acionador de solicitação para ler um formulário HTML

Essa função demonstra como você pode processar o corpo de uma solicitação POST gerada por um formulário HTML (formulário da web), como um formulário "entre em contato conosco". Por exemplo, você pode ter um formulário em HTML como o seguinte:

```
<html>
  <form action="https://example.com" method="post">
    Param 1: <input type="text" name="name1"><br>
    Param 2: <input type="text" name="name2"><br>
    input type="submit" value="Submit">
  </form>
</html>
```

No exemplo a seguir, a função deve ser acionada em uma solicitação de origem ou de um visualizador do CloudFront.

Node.js

```
'use strict';

const querystring = require('querystring');

/**
 * This function demonstrates how you can read the body of a POST request
 * generated by an HTML form (web form). The function is triggered in a
 * CloudFront viewer request or origin request event type.
 */

exports.handler = (event, context, callback) => {
  const request = event.Records[0].cf.request;

  if (request.method === 'POST') {
    /* HTTP body is always passed as base64-encoded string. Decode it. */
    const body = Buffer.from(request.body.data, 'base64').toString();

    /* HTML forms send the data in query string format. Parse it. */
    const params = querystring.parse(body);

    /* For demonstration purposes, we only log the form fields here.
     * You can put your custom logic here. For example, you can store the
     * fields in a database, such as Amazon DynamoDB, and generate a response
     * right from your Lambda@Edge function.
     */
    for (let param in params) {
      console.log(`For "${param}" user submitted "${params[param]}".\n`);
    }
  }

  return callback(null, request);
}
```

```
};
```

Python

```
import base64
from urllib.parse import parse_qs

'''
Say there is a POST request body generated by an HTML such as:

<html>
<form action="https://example.com" method="post">
  Param 1: <input type="text" name="name1"><br>
  Param 2: <input type="text" name="name2"><br>
  input type="submit" value="Submit">
</form>
</html>

'''

'''
This function demonstrates how you can read the body of a POST request
generated by an HTML form (web form). The function is triggered in a
CloudFront viewer request or origin request event type.
'''

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    if request['method'] == 'POST':
        # HTTP body is always passed as base64-encoded string. Decode it
        body = base64.b64decode(request['body']['data'])

        # HTML forms send the data in query string format. Parse it
        params = {k: v[0] for k, v in parse_qs(body).items()}

        '''
        For demonstration purposes, we only log the form fields here.
        You can put your custom logic here. For example, you can store the
        fields in a database, such as Amazon DynamoDB, and generate a response
        right from your Lambda@Edge function.
        '''
        for key, value in params.items():
```

```
    print("For %s use submitted %s" % (key, value))

return request
```

Exemplo: usar um acionador de solicitação para modificar um formulário HTML

Essa função demonstra como você pode modificar o corpo de uma solicitação POST gerada por um formulário HTML (formulário da web). A função é acionada em uma solicitação de origem ou de visualizador do CloudFront.

Node.js

```
'use strict';

const querystring = require('querystring');

exports.handler = (event, context, callback) => {
  var request = event.Records[0].cf.request;
  if (request.method === 'POST') {
    /* Request body is being replaced. To do this, update the following
    /* three fields:
    *   1) body.action to 'replace'
    *   2) body.encoding to the encoding of the new data.
    *
    *       Set to one of the following values:
    *
    *       text - denotes that the generated body is in text format.
    *             Lambda@Edge will propagate this as is.
    *       base64 - denotes that the generated body is base64 encoded.
    *             Lambda@Edge will base64 decode the data before sending
    *             it to the origin.
    *   3) body.data to the new body.
    */
    request.body.action = 'replace';
    request.body.encoding = 'text';
    request.body.data = getUpdatedBody(request);
  }
  callback(null, request);
};

function getUpdatedBody(request) {
  /* HTTP body is always passed as base64-encoded string. Decode it. */
```

```
const body = Buffer.from(request.body.data, 'base64').toString();

/* HTML forms send data in query string format. Parse it. */
const params = querystring.parse(body);

/* For demonstration purposes, we're adding one more param.
 *
 * You can put your custom logic here. For example, you can truncate long
 * bodies from malicious requests.
 */
params['new-param-name'] = 'new-param-value';
return querystring.stringify(params);
}
```

Python

```
import base64
from urllib.parse import parse_qs, urlencode

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    if request['method'] == 'POST':
        ...

        Request body is being replaced. To do this, update the following
        three fields:
            1) body.action to 'replace'
            2) body.encoding to the encoding of the new data.

            Set to one of the following values:

                text - denotes that the generated body is in text format.
                    Lambda@Edge will propagate this as is.
                base64 - denotes that the generated body is base64 encoded.
                    Lambda@Edge will base64 decode the data before sending
                    it to the origin.
            3) body.data to the new body.
        ...

        request['body']['action'] = 'replace'
        request['body']['encoding'] = 'text'
        request['body']['data'] = getUpdatedBody(request)
    return request

def getUpdatedBody(request):
```



```
# HTTP body is always passed as base64-encoded string. Decode it
body = base64.b64decode(request['body']['data'])

# HTML forms send data in query string format. Parse it
params = {k: v[0] for k, v in parse_qs(body).items()}

# For demonstration purposes, we're adding one more param

# You can put your custom logic here. For example, you can truncate long
# bodies from malicious requests
params['new-param-name'] = 'new-param-value'
return urlencode(params)
```

Restrições das funções de borda

Os tópicos a seguir descrevem as restrições aplicáveis ao CloudFront Functions e ao Lambda@Edge. Algumas restrições aplicam-se a todas as funções de borda, enquanto outras são válidas apenas para o CloudFront Functions ou o Lambda@Edge.

Cada tópico fornece informações detalhadas sobre as limitações e as restrições a serem levadas em conta no desenvolvimento e na implantação de funções de borda com o CloudFront.

Compreender essas restrições ajuda a garantir que as funções de borda se comportem conforme o esperado e estejam em conformidade com os recursos compatíveis.

Tópicos

- [Restrições de todas as funções de borda](#)
- [Restrições do CloudFront Functions](#)
- [Restrições ao Lambda@Edge](#)

Para obter mais informações sobre cotas (anteriormente chamadas de limites), consulte [Cotas no CloudFront Functions](#) e [Cotas do Lambda@Edge](#).

Restrições de todas as funções de borda

As restrições a seguir aplicam-se a todas as funções de borda, tanto ao CloudFront Functions quanto ao Lambda@Edge.

Tópicos

- [Propriedade da Conta da AWS](#)
- [Combinação do CloudFront Functions ao Lambda@Edge](#)
- [Códigos de status de HTTP](#)
- [Cabeçalhos HTTP](#)
- [Strings de consulta](#)
- [URI](#)
- [Codificação de URI, string de consulta e cabeçalhos](#)
- [Microsoft Smooth Streaming](#)
- [Tags](#)

Propriedade da Conta da AWS

Para associar uma função de borda a uma distribuição do CloudFront, a função e a distribuição devem pertencer à mesma Conta da AWS.

Combinação do CloudFront Functions ao Lambda@Edge

Para um determinado comportamento de cache, as seguintes restrições são aplicáveis:

- Cada tipo de evento (solicitação do visualizador, solicitação de origem, resposta de origem e resposta do visualizador) pode ter apenas uma associação de função de borda.
- Não é possível combinar o CloudFront Functions e o Lambda@Edge em eventos do visualizador (solicitação do visualizador e resposta do visualizador).

Todas as demais combinações de funções de borda são permitidas. A tabela a seguir explica as combinações permitidas.

		Funções do CloudFront	
		Solicitação do visualizador	Resposta do visualizador
Lambda@Edge	Solicitação do visualizador	Não permitido	Não permitido

Solicitação da origem	Permitido	Permitido
Resposta da origem	Permitido	Permitido
Resposta do visualizador	Não permitido	Não permitido

Códigos de status de HTTP

O CloudFront não invocará funções de borda para eventos de resposta do visualizador se a origem retornar um código de status HTTP 400 ou superior.

As funções do Lambda@Edge para eventos de resposta de origem são chamadas para todas as respostas de origem, incluindo quando a origem retorna um código de status HTTP 400 ou superior. Para obter mais informações, consulte [Atualizar respostas de HTTP em acionadores de resposta da origem](#).

Cabeçalhos HTTP

Determinados cabeçalhos HTTP não são permitidos, o que significa que eles não estão expostos a funções de borda e as funções não podem adicioná-los. Outros cabeçalhos são somente leitura, o que significa que as funções podem lê-los, mas não podem adicioná-los, modificá-los nem excluí-los.

Tópicos

- [Cabeçalhos não permitidos](#)
- [Cabeçalhos somente leitura](#)

Cabeçalhos não permitidos

Os cabeçalhos HTTP a seguir não são expostos a funções de borda e as funções não podem adicioná-los. Se sua função adicionar um desses cabeçalhos, o CloudFront não a validará e retornará o código de status HTTP 502 (gateway inválido) para o visualizador.

- Connection
- Expect
- Keep-Alive
- Proxy-Authenticate

- Proxy-Authorization
- Proxy-Connection
- Trailer
- Upgrade
- X-Accel-Buffering
- X-Accel-Charset
- X-Accel-Limit-Rate
- X-Accel-Redirect
- X-Amz-Cf-*
- X-Amzn-Auth
- X-Amzn-Cf-Billing
- X-Amzn-Cf-Id
- X-Amzn-Cf-Xff
- X-Amzn-ErrorType
- X-Amzn-File-Profile
- X-Amzn-Header-Count
- X-Amzn-Header-Order
- X-Amzn-Lambda-Integration-Tag
- X-Amzn-RequestId
- X-Cache
- X-Edge-*
- X-Forwarded-Proto
- X-Real-IP

Cabeçalhos somente leitura

Os cabeçalhos a seguir são somente leitura. Sua função pode lê-los ou usá-los como entrada para a lógica da função, mas não podem alterar os valores. Se sua função adicionar ou editar um cabeçalho somente leitura, a solicitação falhará na validação do CloudFront, o qual retornará o código de status HTTP 502 (gateway inválido) para o visualizador.

Cabeçalhos somente leitura em eventos de solicitação do visualizador

Os cabeçalhos a seguir são somente leitura em eventos de solicitação do visualizador.

- Content-Length
- Host
- Transfer-Encoding
- Via

Cabeçalhos somente leitura em eventos de solicitação de origem (somente Lambda@Edge)

Os seguintes cabeçalhos são somente leitura em eventos de solicitação de origem, os quais existem apenas no Lambda@Edge.

- Accept-Encoding
- Content-Length
- If-Modified-Since
- If-None-Match
- If-Range
- If-Unmodified-Since
- Transfer-Encoding
- Via

Cabeçalhos somente leitura em eventos de resposta de origem (somente Lambda@Edge)

Os seguintes cabeçalhos são somente leitura em eventos de resposta de origem, os quais existem apenas no Lambda@Edge.

- Transfer-Encoding
- Via

Cabeçalhos somente leitura em eventos de resposta do visualizador

Os cabeçalhos a seguir são somente leitura em eventos de resposta do visualizador para o CloudFront Functions e para o Lambda@Edge.

- Warning
- Via

Os cabeçalhos a seguir são somente leitura em eventos de resposta do visualizador para o Lambda@Edge.

- Content-Length
- Content-Encoding
- Transfer-Encoding

Strings de consulta

As restrições a seguir aplicam-se a funções que leem, atualizam ou criam uma string de consulta em um URI de solicitação.

- (Somente Lambda@Edge) Para acessar a string de consulta em uma solicitação de origem ou função de resposta de origem, sua política de cache ou política de solicitação de origem deve ser definida como All (Todas) para Query strings (Strings de consulta).
- Uma função pode criar ou atualizar uma string de consulta para eventos de solicitação do visualizador e solicitação da origem (eventos de solicitação da origem existem apenas no Lambda@Edge).
- Uma função pode ler uma string de consulta, mas não pode criar ou atualizar uma, para eventos de resposta da origem e resposta do visualizador (eventos de resposta da origem existem apenas no Lambda@Edge).
- Se uma função criar ou atualizar uma string de consulta, as seguintes restrições se aplicarão:
 - A string de consulta não pode incluir espaços, caracteres de controle nem o identificador de fragmento (#).
 - O tamanho total do URI, incluindo a string de consulta, deve ser menor que 8.192 caracteres.
 - Recomendamos o uso de codificação percentual para o URI e a string de consulta. Para ter mais informações, consulte [Codificação de URI, string de consulta e cabeçalhos](#).

URI

Se uma função alterar o URI para uma solicitação, o comportamento do cache da solicitação ou a origem para a qual a solicitação é encaminhada não será alterada.

O tamanho total do URI, incluindo a string de consulta, deve ser menor que 8.192 caracteres.

Codificação de URI, string de consulta e cabeçalhos

Os valores de string de consulta e de URI e os cabeçalhos transmitidos para as funções de borda são codificados em UTF-8. A função deve usar codificação UTF-8 para os valores de URI, de string de consulta e de cabeçalhos exibidos. A codificação percentual é compatível com a codificação UTF-8.

A seguinte lista explica como o CloudFront lida com a codificação de URI, string de consulta e cabeçalhos:

- Quando os valores na solicitação são codificados em UTF-8, o CloudFront encaminha os valores para a função sem alterá-los.
- Quando os valores na solicitação são [codificados em ISO-8859-1](#), o CloudFront os converte para a codificação UTF-8 antes de encaminhá-los para sua função.
- Quando os valores na solicitação são codificados usando qualquer outra codificação de caracteres, o CloudFront assume que eles estão codificados em ISO 8859-1 e tenta convertê-los de ISO-8859-1 em UTF-8.

Important

A versão convertida pode ser uma interpretação imprecisa dos valores da solicitação original. Isso pode fazer com que sua função ou origem produzam um resultado indesejado.

Os valores de URI, de string de consulta e de cabeçalhos que o CloudFront encaminhará à sua origem dependerão de uma função alterá-los ou não:

- Se uma função não alterar o URI, a string de consulta ou o cabeçalho, o CloudFront encaminhará à origem os valores recebidos na solicitação.
- Se uma função alterar o URI, a string de consulta ou o cabeçalho, o CloudFront encaminhará os valores codificados em UTF-8.

Microsoft Smooth Streaming

Não é possível usar funções de borda com uma distribuição do CloudFront utilizada em streaming de arquivos de mídia transcodificados no formato Microsoft Smooth Streaming.

Tags

Não é possível adicionar tags a funções de borda. Consulte mais informações sobre marcação no CloudFront em [Marcar uma distribuição](#).

Restrições do CloudFront Functions

As restrições a seguir aplicam-se somente ao CloudFront Functions.

Sumário

- [Logs](#)
- [Corpo da solicitação](#)
- [Usar credenciais temporárias com a API KeyValueCollection do CloudFront](#)
- [Runtime](#)
- [Utilização de recursos de computação](#)

Para ter mais informações sobre cotas (anteriormente chamadas de limites), consulte [Cotas no CloudFront Functions](#).

Logs

Os logs de função no CloudFront Functions são truncados em 10 KB.

Corpo da solicitação

O CloudFront Functions não pode acessar o corpo da solicitação HTTP.

Usar credenciais temporárias com a API KeyValueCollection do CloudFront

É possível usar o AWS Security Token Service (AWS STS) para gerar credenciais de segurança temporárias (também conhecidas como tokens de sessão). Os tokens de sessão permitem assumir temporariamente um perfil do AWS Identity and Access Management (IAM) para que você possa acessar Serviços da AWS.

Para chamar a [API KeyValueCollection do CloudFront](#), use um endpoint Regional no AWS STS para exibir um token de sessão da versão 2. Se você usar o endpoint global para o AWS STS (`sts.amazonaws.com`), o AWS STS vai gerar um token de sessão da versão 1, que não é compatível com o Signature Version 4A (SigV4A). Como consequência, você receberá um erro de autenticação.

Para chamar a API KeyValueCollection do CloudFront, é possível usar as seguintes opções:

AWS CLI e AWS SDKs

É possível configurar a AWS CLI ou um SDK da AWS para usar endpoints regionais do AWS STS. Para ter mais informações, consulte [AWS STS Regionalized endpoints](#) no Guia de referência de ferramentas e SDKs da AWS.

Para ter mais informações sobre endpoints do AWS STS, consulte [Regiões e endpoints](#) no Guia do usuário do IAM.

SAML

É possível configurar o SAML para usar endpoints regionais do AWS STS. Para ter mais informações, consulte a publicação de blog [How to use regional SAML endpoints for failover](#).

API do `SetSecurityTokenServicePreferences`

Em vez de usar um endpoint regional do AWS STS, é possível configurar o endpoint global para o AWS STS para exibir tokens de sessão da versão 2. Use a operação de API [SetSecurityTokenServicePreferences](#) para fazer isso e configurar a Conta da AWS.

Example Exemplo: comando da CLI do IAM

```
aws iam set-security-token-service-preferences --global-endpoint-token-version v2Token
```

Tip

Recomendamos usar endpoints regionais do AWS STS em vez dessa opção. Os endpoints regionais oferecem maior disponibilidade e casos de failover.

Provedor de identidades personalizado

Se você estiver usando um provedor de identidades personalizado que faça a federação e assuma o perfil, use uma das opções anteriores para o sistema de provedor de identidades principal responsável por gerar o token de sessão.

Runtime

O ambiente de runtime do CloudFront Functions não permite a avaliação dinâmica de código e restringe o acesso à rede, ao sistema de arquivos, às variáveis do ambiente e aos temporizadores. Para ter mais informações, consulte [Recursos restritos](#).

Note

Para usar o CloudFront KeyValueCollection, a função do CloudFront deve usar o [runtime 2.0 do JavaScript](#).

Utilização de recursos de computação

O CloudFront Functions tem um limite de tempo para executar, o qual é medido como Utilização de recursos de computação. A utilização de recursos de computação é um número entre 0 e 100 que indica a quantidade de tempo que a função levou para ser executada como um percentual do tempo máximo permitido. Por exemplo, uma utilização de computação de 35 significa que a função foi concluída em 35% do tempo máximo permitido.

Quando você [testa uma função](#), é possível ver o valor de utilização de recursos de computação na saída do evento de teste. Para funções de produção, você pode visualizar a [compute utilization metric](#) (métrica de utilização de recursos de computação) na [Página de monitoramento no console do CloudFront](#) ou no CloudWatch.

Restrições ao Lambda@Edge

As restrições a seguir aplicam-se somente ao Lambda@Edge.

Sumário

- [Resolução do DNS](#)
- [Códigos de status de HTTP](#)

- [Versionamento da função do Lambda](#)
- [Região do Lambda](#)
- [Permissões de função do Lambda](#)
- [Recursos do Lambda](#)
- [Tempos de execução compatíveis](#)
- [Cabeçalhos do CloudFront](#)
- [Restrições do corpo da solicitação com a opção de incluir corpo](#)
- [Tempo limite de resposta e de manutenção \(somente origens personalizadas\)](#)

Para obter informações sobre cotas, consulte [Cotas do Lambda@Edge](#).

Resolução do DNS

O CloudFront executa uma resolução de DNS no nome de domínio da origem antes de executar a solicitação de origem da função do Lambda@Edge. Se o serviço DNS do domínio estiver com problemas e o CloudFront não puder resolver o nome de domínio para obter o endereço IP, a função do Lambda@Edge não será invocada. O CloudFront retornará um [código de status HTTP 502 \(gateway inativo\)](#) ao cliente. Para ter mais informações, consulte [Erro de DNS \(NonS3OriginDnsError\)](#).

Para ter mais informações sobre como gerenciar failover de DNS, consulte [Configurar failover de DNS](#) no Guia do desenvolvedor do Amazon Route 53.

Códigos de status de HTTP

As funções do Lambda@Edge para eventos de resposta do visualizador não podem modificar o código de status HTTP da resposta, independentemente de a resposta ter vindo da origem ou do cache do CloudFront.

Versionamento da função do Lambda

Você deve usar uma versão numerada da função do Lambda, e não \$LATEST nem aliases.

Região do Lambda

A função do Lambda deve estar na região Leste dos EUA (Norte da Virgínia).

Permissões de função do Lambda

A função de execução do IAM associada à função do Lambda deve permitir que os principais de serviço `lambda.amazonaws.com` e `edgelambda.amazonaws.com` assumam a função. Para ter mais informações, consulte [Configurar permissões e perfis do IAM para o Lambda@Edge](#).

Recursos do Lambda

Os seguintes recursos do Lambda não são compatíveis com o Lambda@Edge:

- [Configurações de gerenciamento de runtime do Lambda](#) diferentes de Auto (padrão)
- Configuração de sua função do Lambda para acessar recursos na VPC
- [Filas de mensagens não entregues da função do Lambda](#)
- [Variáveis de ambiente do Lambda](#) (exceto variáveis de ambiente reservadas, que são automaticamente compatíveis)
- Funções do Lambda com [Gerenciar dependências do AWS Lambda com camadas](#)
- [Como usar o AWS X-Ray](#)
- Simultaneidade provisionada do Lambda

Note

As funções do Lambda@Edge têm os mesmos recursos de [Simultaneidade regional](#) que as funções do Lambda. Para ter mais informações, consulte [Cotas do Lambda@Edge](#).

- [Criar uma função do Lambda usando uma imagem de contêiner](#)
- [Funções do Lambda que usam a arquitetura arm64](#)
- Funções do Lambda com mais de 512 MB de armazenamento temporário
- Capturar logs de funções do Lambda no formato estruturado JSON
- Controle de detalhamento em nível de log dos logs da função do Lambda
- Definir para qual grupo de logs do Amazon CloudWatch o Lambda deve enviar logs

Tempos de execução compatíveis

O Lambda@Edge oferece suporte a funções do Lambda com os seguintes tempos de execução:

Node.js	Python
<ul style="list-style-type: none">• Node.js 20• Node.js 18• Node.js 16¹• Node.js 14²• Node.js 12²• Node.js 10²• Node.js 8²• Node.js 6²	<ul style="list-style-type: none">• Python 3.12• Python 3.11• Python 3.10• Python 3.9• Python 3.8• Python 3.7

¹ Essa versão do Node.js chegou ao fim da vida útil e em breve será descontinuada pelo AWS Lambda.

² Essa versão do Node.js chegou ao fim da vida útil e foi totalmente descontinuada pelo AWS Lambda.

Não é possível criar nem atualizar funções com versões descontinuadas do Node.js. Só é possível associar funções existentes a essas versões com distribuições do CloudFront. Funções com essas versões que estão associadas a distribuições continuam a ser executadas. No entanto, recomendamos mover sua função para versões mais recentes do Node.js. Para obter mais informações, consulte [Política de descontinuação de tempo de execução](#) no Guia do desenvolvedor do AWS Lambda e a [Agenda de versões do Node.js](#) no GitHub.

Tip

Como prática recomendada, use as versões mais recentes dos runtimes fornecidos para melhorias de desempenho e novos atributos.

Cabeçalhos do CloudFront

As funções do Lambda@Edge podem ler, editar, remover ou adicionar qualquer um dos cabeçalhos do CloudFront listados em [Adicionar cabeçalhos de solicitação do CloudFront](#).

Observações

- Para que o CloudFront adicione esses cabeçalhos, configure-o para adicioná-los usando uma [política de cache](#) ou [política de solicitação de origem](#).
- O CloudFront adiciona os cabeçalhos após o evento de solicitação do visualizador, o que significa que eles não estão disponíveis para as funções do Lambda@Edge em uma solicitação do visualizador. Os cabeçalhos só estão disponíveis para funções do Lambda@Edge em uma solicitação e resposta de origem.
- Se a solicitação do visualizador incluir cabeçalhos que têm esses nomes e você configurou o CloudFront para adicionar esses cabeçalhos usando uma [política de cache](#) ou [política de solicitação de origem](#), o CloudFront substituirá os valores de cabeçalho que estavam na solicitação do visualizador. As funções voltadas para o visualizador veem o valor do cabeçalho da solicitação do visualizador, enquanto as funções voltadas para a origem veem o valor do cabeçalho adicionado pelo o CloudFront.
- Se uma função de solicitação do visualizador adicionar o cabeçalho `CloudFront-Viewer-Country`, a validação falhará e o CloudFront retornará o código de status HTTP 502 (gateway inválido) para o visualizador.

Restrições do corpo da solicitação com a opção de incluir corpo

Ao escolher a opção Incluir corpo para expor o corpo da solicitação à função do Lambda@Edge, as informações e limites de tamanho a seguir se aplicam às partes do corpo que são expostas ou substituídas.

- O CloudFront sempre codifica em base64 o corpo da solicitação antes de expô-lo ao Lambda@Edge.
- Se o corpo da solicitação for grande, o CloudFront o truncará antes de expô-lo ao Lambda@Edge da seguinte forma:
 - Para eventos de solicitação do visualizador, o corpo é truncado em 40 KB.
 - Para eventos de solicitação da origem, o corpo é truncado em 1 MB.
- Se você acessar o corpo da solicitação como somente leitura, o CloudFront enviará o corpo da solicitação original completo à origem.
- Se a função do Lambda@Edge substituir o corpo da solicitação, os limites de tamanho a seguir se aplicarão ao corpo retornado pela função:

- Se a função do Lambda@Edge retornar o corpo como texto simples:
 - Para eventos de solicitação do visualizador, o limite do corpo é de 40 KB.
 - Para eventos de solicitação da origem, o limite do corpo é de 1 MB.
- Se a função do Lambda@Edge retornar o corpo como texto codificado em base64:
 - Para eventos de solicitação do visualizador, o limite do corpo é de 53,2 KB.
 - Para eventos de solicitação da origem, o limite do corpo é de 1,33 MB.

Note

Se a função do Lambda@Edge retornar um corpo que exceda esses limites, a solicitação falhará com um código de status HTTP 502 ([Erro de validação do Lambda](#)). Recomendamos que você atualize a função do Lambda@Edge para que o corpo não exceda esses limites.

Tempo limite de resposta e de manutenção (somente origens personalizadas)

Se estiver usando as funções do Lambda@Edge para definir o tempo limite de resposta ou de manutenção para as origens da distribuição, verifique se você está especificando um valor compatível com a origem. Para ter mais informações, consulte [Cotas de tempo limite de resposta e manutenção](#).

Relatórios, métricas e logs

O CloudFront oferece várias opções para relatar, monitorar e registrar em log os recursos do CloudFront. É possível concluir as seguintes tarefas:

- Visualizar e baixar relatórios para ver o uso e a atividade das distribuições do CloudFront, inclusive relatórios de faturamento, estatísticas de cache, conteúdo conhecido e principais indicadores.
- Monitorar e rastrear o CloudFront, inclusive as [funções de computação de borda](#), diretamente do console do CloudFront ou usando o Amazon CloudWatch. O CloudFront envia métricas ao CloudWatch para distribuições e funções de borda, tanto do Lambda@Edge quanto do CloudFront Functions.
- Visualizar logs das solicitações do visualizador que as distribuições do CloudFront recebem com logs padrão ou em tempo real. Além dos logs de solicitação do visualizador, você pode usar o CloudWatch Logs para obter logs para suas funções de borda, tanto do Lambda@Edge quanto do CloudFront Functions. Você também pode usar o AWS CloudTrail para obter logs da atividade da API do CloudFront em sua Conta da AWS.
- Monitorar as alterações de configuração nos recursos do CloudFront usando o AWS Config.

Consulte mais informações sobre essas opções nos tópicos a seguir.

Tópicos

- [Relatórios de uso e faturamento da AWS para o CloudFront](#)
- [Visualizar relatórios do CloudFront no console](#)
- [Monitorar métricas do CloudFront com o Amazon CloudWatch](#)
- [Registro em log do CloudFront e de funções de borda](#)
- [Monitorar as alterações de configuração com o AWS Config](#)

Relatórios de uso e faturamento da AWS para o CloudFront

A AWS fornece dois relatórios de uso para o CloudFront:

- O relatório de faturamento da AWS é uma visualização de alto nível de todas as atividades dos Serviços da AWS que você está usando, inclusive o CloudFront.

- O relatório de uso da AWS é um resumo da atividade de um serviço específico, agregado por hora, dia ou mês. Ele também inclui gráficos de uso que fornecem uma representação gráfica do uso que você faz do CloudFront.

Note

Como os outros Serviços da AWS, o CloudFront cobra apenas pelo que você usa. Para ter mais informações, consulte [Definição de preço do CloudFront](#).

Tópicos

- [Visualizar o relatório de faturamento da AWS para o CloudFront](#)
- [Visualizar o relatório de uso da AWS para o CloudFront](#)
- [Interpretar os relatórios de uso e faturamento da AWS para o CloudFront](#)

Visualizar o relatório de faturamento da AWS para o CloudFront

É possível ver um resumo do uso e das cobranças da AWS, indicadas por serviço, na página Faturas no console do AWS Billing and Cost Management.

Como visualizar o relatório de faturamento da AWS

1. Faça login no AWS Management Console e abra o console do AWS Billing em <https://console.aws.amazon.com/billing/>.
2. No painel de navegação, selecione Contas.
3. Escolha um Período de cobrança (por exemplo, agosto de 2023).
4. Na guia Cobranças por serviço, escolha CloudFront e, depois, expanda Global ou o nome Região da AWS.
5. Para baixar um relatório de faturamento detalhado no formato CSV, selecione Fazer download de tudo para CSV.

Para ter mais informações sobre a fatura da AWS, consulte [Viewing your bill](#) no Guia do usuário do AWS Billing.

O relatório de faturamento inclui os seguintes valores que se aplicam ao CloudFront:

- **ProductCode:** AmazonCloudFront.
- **UsageType:** um dos seguintes valores:
 - Um código que identifica o tipo de transferência de dados
 - Invalidations
 - Executions-CloudFrontFunctions
 - KeyValueStore-APIOperations
 - KeyValueStore-EdgeReads
 - RealTimeLog-KinesisDataStream
 - SSL-Cert-Custom
- **ItemDescription:** descrição da taxa de faturamento para o UsageType.
- **UsageStart Date e UsageEndDate:** o dia do uso, no Tempo Universal Coordenado (UTC).
- **UsageQuantity:** um dos seguintes valores:
 - O número de solicitações durante o período especificado
 - A quantidade de dados transferidos, em gigabytes
 - O número de objetos invalidados
 - A soma dos meses pro rata nos quais você teve certificados SSL associados a distribuições habilitadas do CloudFront. Por exemplo, se você tiver um certificado associado a uma distribuição ativada por um mês inteiro e outro certificado associado a uma distribuição ativada metade do mês, esse valor será 1,5.

Visualizar o relatório de uso da AWS para o CloudFront

A AWS fornece um relatório de uso do CloudFront que é mais detalhado do que o relatório de faturamento, mas menos detalhado do que os logs de acesso do CloudFront. O relatório de uso fornece dados de uso agregados por hora, dia ou mês e indica as operações por região e tipo de uso, como dados transferidos da região da Austrália.

Como visualizar o relatório de uso da AWS

1. Faça login no AWS Management Console e abra o console do AWS Billing em <https://console.aws.amazon.com/billing/>.
2. No painel de navegação, selecione Custo e relatórios.
3. Em Relatório de uso da AWS, selecione Criar um relatório de uso.

4. Na página Baixar relatório de uso, em Serviços, escolha Amazon CloudFront
5. Escolha o Tipo de uso.
6. Escolha a guia Operação.
7. Escolha o Período para o relatório. Se você escolher Intervalo de datas personalizado, precisará especificar o Intervalo de datas para o relatório manualmente.
8. Em Granularidade do relatório, escolha Horário, Diário ou Mensal.
9. Escolha Baixar e, em seguida, escolha Relatório XML ou Relatório CSV.

Para ter mais informações sobre o relatório de uso da AWS, consulte [AWS Usage Report](#) no Guia do usuário do Exportações de dados da AWS.

O relatório de uso do CloudFront inclui os seguintes valores:

- Serviço: AmazonCloudFront
- Operation: método HTTP. Os valores incluem DELETE, GET, HEAD, OPTIONS, PATCH, POST e PUT.
- UsageType: um dos seguintes valores:
 - Um código que identifica o tipo de transferência de dados
 - Invalidations
 - Executions-CloudFrontFunctions
 - KeyValueStore-APIOperations
 - KeyValueStore-EdgeReads
 - RealTimeLog-KinesisDataStream
 - SSL-Cert-Custom
- Resource: o ID da distribuição do CloudFront associada ao uso ou de um certificado SSL associado a uma distribuição do CloudFront.
- StartTime/EndTime: o dia aplicável do uso, no Tempo Universal Coordenado (UTC).
- UsageValue: 1) O número de solicitações durante o período especificado ou 2) a quantidade de dados transferidos, em bytes.

Se você estiver usando o Amazon S3 como origem do CloudFront, considere executar o relatório de uso do Amazon S3 também. No entanto, se você usar o Amazon S3 para outros fins além de como uma origem das distribuições do CloudFront, a parte que se aplica ao uso do CloudFront pode não ficar clara.

i Tip

Para obter informações detalhadas sobre cada solicitação recebida pelo CloudFront para seus objetos, ative os logs de acesso do CloudFront para a sua distribuição. Para ter mais informações, consulte [the section called “Usar logs padrão \(logs de acesso\)”](#).

Para ter mais informações sobre noções básicas das cobranças e dos tipos de uso do CloudFront nos relatórios, consulte [the section called “Interpretar os relatórios de uso e faturamento da AWS para o CloudFront”](#).

Interpretar os relatórios de uso e faturamento da AWS para o CloudFront

De posse do [relatório de faturamento](#) e do [relatório de uso](#), você pode usar este tópico para entender como interpretar cada cobrança do CloudFront exibida na fatura e o tipo de uso correspondente de cada cobrança. Este tópico inclui os códigos e as abreviações de Região da AWS que podem aparecer nos dois relatórios.

A maioria dos códigos das duas colunas incluem uma abreviação de duas letras que indica a localização da atividade. Na tabela abaixo, a *região* em um código é substituída em sua fatura e no relatório de uso da AWS por uma das seguintes abreviações de duas letras:

- AP: Hong Kong, Filipinas, Coreia do Sul, Taiwan e Cingapura (Ásia-Pacífico)
- AU: Austrália
- CA: Canadá
- UE: Europa e Israel
- IN: Índia
- JP: Japão
- OM: Oriente Médio
- SA: América do Sul
- US: Estados Unidos
- ZA: África do Sul

Para ter mais informações sobre definição de preço por Região da AWS, consulte [Preço do Amazon CloudFront](#).

Observações

- Esta tabela não inclui cobranças de transferência de objetos de um bucket do Amazon S3 para pontos de presença do CloudFront. Essas cobranças, se for o caso, serão exibidas na seção Transferência de dados da AWS em sua fatura da AWS.
- A primeira coluna indica as cobranças exibidas no relatório de faturamento da AWS e explica o que cada uma significa.
- A segunda coluna indica os itens exibidos no relatório de uso AWS e mostra a correlação entre os itens da fatura e do relatório de uso.

Cobranças do CloudFront na fatura da AWS	Valores da coluna UsageType no relatório de uso da AWS
<p><i>region</i>-DataTransfer-Out-Bytes</p> <p>Total de bytes fornecidos dos locais da borda do CloudFront na <i>região</i> em resposta a solicitações GET e HEAD do usuário.</p>	<p><i>region</i>-Out-Bytes-HTTP-Static:</p> <p>Bytes fornecidos por HTTP para objetos com TTL \geq 3.600 segundos.</p> <p><i>region</i>-Out-Bytes-HTTPS-Static:</p> <p>Bytes fornecidos por HTTP para objetos com TTL \geq 3.600 segundos.</p> <p><i>region</i>-Out-Bytes-HTTP-Dynamic:</p> <p>Bytes fornecidos por HTTP para objetos com TTL $<$ 3.600 segundos.</p> <p><i>region</i>-Out-Bytes-HTTPS-Dynamic:</p> <p>Bytes fornecidos por HTTP para objetos com TTL $<$ 3.600 segundos.</p> <p><i>region</i>-Out-Bytes-HTTP-Proxy:</p>

Cobranças do CloudFront na fatura da AWS	Valores da coluna UsageType no relatório de uso da AWS
	<p>Bytes retornados do CloudFront para os visualizadores em resposta a solicitações DELETE, OPTIONS, PATCH, POST e PUT.</p> <p><i>region</i>-Out-Bytes-HTTPS-Proxy:</p> <p>Bytes retornados do CloudFront para os visualizadores em resposta a solicitações DELETE, OPTIONS, PATCH, POST e PUT.</p>
<p><i>region</i>-DataTransfer-Out-OBytes</p> <p>Total de bytes transferidos de locais da borda do CloudFront para a sua origem ou função de borda em resposta a solicitações DELETE, OPTIONS, PATCH, POST e PUT. As cobranças incluem transferência de dados para dados WebSocket do cliente para o servidor.</p>	<p><i>region</i>-Out-OBytes-HTTP-Proxy</p> <p>Total de bytes transferidos por HTTP de locais da borda do CloudFront para a sua origem ou função de borda em resposta a solicitações DELETE, OPTIONS, PATCH, POST e PUT.</p> <p><i>region</i>-Out-OBytes-HTTPS-Proxy</p> <p>Total de bytes transferidos por HTTP de locais da borda do CloudFront para a sua origem ou função de borda em resposta a solicitações DELETE, OPTIONS, PATCH, POST e PUT.</p>
<p><i>region</i>-Requests-Tier1</p> <p>Número de solicitações HTTP GET e HEAD.</p>	<p><i>region</i>-Requests-HTTP-Static</p> <p>Número de solicitações HTTP GET e HEAD atendidas de objetos com TTL \geq 3.600 segundos.</p> <p><i>region</i>-Requests-HTTP-Dynamic</p> <p>Número de solicitações HTTP GET e HEAD atendidas de objetos com TTL $<$ 3.600 segundos.</p>

Cobranças do CloudFront na fatura da AWS	Valores da coluna UsageType no relatório de uso da AWS
<p><i>region</i>-Requests-Tier2-HTTPS</p> <p>Número de solicitações HTTPS GET e HEAD.</p>	<p><i>region</i>-Requests-HTTPS-Static</p> <p>Número de solicitações HTTPS GET e HEAD atendidas de objetos com TTL \geq 3.600 segundos.</p> <p><i>region</i>-Requests-HTTPS-Dynamic</p> <p>Número de solicitações HTTPS GET e HEAD atendidas de objetos com TTL $<$ 3.600 segundos.</p>
<p><i>region</i>-Requests-HTTP-Proxy</p> <p>Número de solicitações HTTP DELETE, OPTIONS, PATCH, POST e PUT que o CloudFront encaminha para a sua origem ou função de borda.</p> <p>Também inclui o número de solicitações de WebSocket de HTTP (solicitações de GET com o cabeçalho Upgrade: websocket) que o CloudFront encaminha para a sua origem ou função de borda.</p>	<p><i>region</i>-Requests-HTTP-Proxy</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>

Cobranças do CloudFront na fatura da AWS	Valores da coluna UsageType no relatório de uso da AWS
<p><i>region</i>-Requests-HTTPS-Proxy</p> <p>Número de solicitações HTTP DELETE, OPTIONS, PATCH, POST e PUT que o CloudFront encaminha para a sua origem ou função de borda.</p> <p>Também inclui o número de solicitações de WebSocket de HTTPS (solicitações de GET com o cabeçalho Upgrade: websocket) que o CloudFront encaminha para a sua origem ou função de borda.</p>	<p><i>region</i>-Requests-HTTPS-Proxy</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>
<p><i>region</i>-Requests-HTTPS-Proxy-FLE</p> <p>Número de solicitações HTTPS DELETE, OPTIONS, PATCH e POST processadas com a criptografia em nível de campo que o CloudFront encaminha para a sua origem ou função de borda.</p>	<p><i>region</i>-Requests-HTTPS-Proxy-FLE</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>
<p><i>region</i>-Bytes-OriginShield</p> <p>Total de bytes transferidos da origem para qualquer cache de borda regional, incluindo o cache de borda regional que está habilitado como Origin Shield (Shield de origem).</p>	<p><i>region</i>-Bytes-OriginShield</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>
<p><i>region</i>-OBytes-OriginShield</p> <p>Total de bytes transferidos para a origem a partir de qualquer cache de borda regional, incluindo o cache de borda regional que está habilitado como Origin Shield (Shield de origem).</p>	<p><i>region</i>-OBytes-OriginShield</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>

Cobranças do CloudFront na fatura da AWS	Valores da coluna UsageType no relatório de uso da AWS
<p><i>region</i>-Requests-OriginShield</p> <p>Número de solicitações que vão para Origin Shield (Shield de origem) como uma camada incremental. Para solicitações dinâmicas (não armazenáveis em cache) encaminhadas por proxy para a origem, o Origin Shield é sempre uma camada incremental. Para solicitações que podem ser armazenados em cache, o Shield de origem é às vezes uma camada incremental.</p> <p>Para obter mais informações, consulte the section called “Estimar custos do Origin Shield”.</p>	<p><i>region</i>-Requests-OriginShield</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>
<p>Invalidações</p> <p>A cobrança pela invalidação de objetos (remoção de objetos de locais de borda do CloudFront). Para ter mais informações, consulte Pagar pela invalidação de arquivos.</p>	<p>Invalidações</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>
<p>SSL-Cert-Custom</p> <p>A cobrança pelo uso de um certificado SSL com um nome de domínio alternativo do CloudFront, como example.com, ao invés de usar o certificado SSL padrão do CloudFront e o nome de domínio que o CloudFront atribui à sua distribuição.</p>	<p>SSL-Cert-Custom</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>
<p>RealTimeLog-KinesisDataStream</p> <p>A cobrança pelo número de linhas geradas para logs em tempo real.</p>	<p>RealTimeLog-KinesisDataStream</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>

Cobranças do CloudFront na fatura da AWS	Valores da coluna UsageType no relatório de uso da AWS
<p>Executions-CloudFrontFunctions</p> <p>A cobrança pelo número de invocações do CloudFront Functions.</p>	<p>Executions-CloudFrontFunctions</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>
<p>region-Lambda-Edge-Request</p> <p>A cobrança pelo número de invocações da função do Lambda@Edge.</p>	<p>region-Lambda-Edge-Request</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>
<p>region-Lambda-Edge-GB-Second</p> <p>A cobrança pela duração desde o momento em que a função do Lambda@Edge é invocada até o momento em que ela retorna ou é encerrada.</p>	<p>region-Lambda-Edge-GB-Second</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>
<p>KeyValueStore-EdgeReads</p> <p>A cobrança pelo número de chamadas de leitura para os CloudFront KeyValueStore, <code>get()</code>, <code>exists()</code> e <code>meta()</code>. Para ter mais informações, consulte Métodos auxiliares para armazenamentos de chave-valor.</p>	<p>KeyValueStore-EdgeReads</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>
<p>KeyValueStore-APIOperations</p> <p>A cobrança pelo número de chamadas para a API CloudFront KeyValueStore.</p>	<p>KeyValueStore-APIOperations</p> <p>Equivalente ao item correspondente na fatura do CloudFront.</p>

Visualizar relatórios do CloudFront no console

Cada relatório fornece informações e visualizações detalhadas. Desse modo, você pode otimizar a entrega de conteúdo, identificar gargalos de performance e tomar decisões orientadas por dados. Se você precisa monitorar a eficiência do cache, analisar padrões de tráfego ou entender melhor seus visualizadores, use esses relatórios para monitorar e analisar com eficácia as distribuições do CloudFront.

É possível visualizar os seguintes relatórios da atividade do CloudFront no console:

Tópicos

- [Visualizar relatórios de estatísticas de cache do CloudFront](#)
- [Visualizar relatórios de objetos comuns do CloudFront](#)
- [Visualizar relatórios dos principais indicadores do CloudFront](#)
- [Visualizar relatórios de uso do CloudFront](#)
- [Visualizar relatórios de visualizadores do CloudFront](#)

A maioria desses relatórios é baseada nos dados dos logs de acesso do CloudFront que contêm informações detalhadas sobre cada solicitação do usuário recebida por ele. Não é necessário permitir que os logs de acesso visualizem os relatórios. Para ter mais informações, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Visualizar relatórios de estatísticas de cache do CloudFront

O relatório de estatísticas de cache do Amazon CloudFront mostra as seguintes informações:

- Total de solicitações: o número total de solicitações de todos os códigos de status HTTP (por exemplo, 200 ou 404) e de todos os métodos (por exemplo, GET, HEAD ou POST).
- Porcentagem de solicitações do visualizador por tipo de resultado: o número de solicitações atendidas e não atendidas, e erros como uma porcentagem do total de solicitações do visualizador para a distribuição do CloudFront selecionada.
- Bytes transferidos aos visualizadores: o total de bytes e os bytes de solicitações não atendidas.
- Códigos de status HTTP: as solicitações do visualizador por código de status HTTP.
- Porcentagem de solicitações GET cujo download não foi concluído: as solicitações GET do visualizador com download do objeto solicitado não concluído como porcentagem do total de solicitações.

Os dados dessas estatísticas são obtidos da mesma origem que os logs de acesso do CloudFront. No entanto, não é necessário habilitar o [registro em log de acesso](#) para visualizar as estatísticas de cache.

Você pode exibir gráficos para um intervalo de datas nos últimos 60 dias, com pontos de dados por hora ou dia. Normalmente, é possível visualizar os dados das solicitações recebidas pelo CloudFront há uma hora, mas eles podem atrasar até 24 horas.

Tópicos

- [Visualizar relatórios de estatísticas de cache do CloudFront no console](#)
- [Baixar dados no formato CSV](#)
- [Como os gráficos de estatísticas do cache estão relacionados aos dados nos logs padrão \(logs de acesso\) do CloudFront](#)

Visualizar relatórios de estatísticas de cache do CloudFront no console

É possível visualizar o relatório de estatísticas de cache do CloudFront no console.

Como visualizar o relatório de estatísticas de cache do CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Estatísticas do cache.
3. No painel CloudFront Cache Statistics Reports (Relatórios de estatísticas de cache do CloudFront), em Start Date (Data de início) e End Date (Data de término), selecione o intervalo de datas para o qual você deseja exibir gráficos de estatísticas do cache. Os intervalos disponíveis dependem do valor selecionado para Granularity:
 - Daily (Diariamente): para exibir gráficos com um ponto de dados por dia, selecione qualquer intervalo de datas nos últimos 60 dias.
 - Hourly (Por hora): para exibir gráficos com um ponto de dados por hora, selecione qualquer intervalo de datas de até 14 dias nos últimos 60 dias.

As datas e horas estão em Tempo Universal Coordenado (UTC).

4. Em Granularity, especifique se você deseja exibir um ponto de dados por dia ou por hora nos gráficos. Se você especificar um intervalo de datas maior que 14 dias, a opção para especificar um ponto de dados por hora não estará disponível.
5. Em Viewer Location, escolha o continente de origem das solicitações do visualizador ou escolha All Locations. Os gráficos de estatísticas do cache incluem dados de solicitações recebidas pelo CloudFront do local especificado.
6. Na lista Distribution, selecione as distribuições para as quais você deseja exibir os dados nos gráficos de uso:

- An individual distribution (uma distribuição individual): os gráficos exibem os dados da distribuição selecionada do CloudFront. A lista Distribution exibe o ID e os nomes de domínio alternativos (CNAMEs) da distribuição, se houver. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá os nomes de domínio de origem dela.
- Todas as distribuições: os gráficos exibem a soma dos dados de todas as distribuições que estão associadas à Conta da AWS atual, com exceção das distribuições que você excluiu.

7. Selecione Atualizar.

Tip

- Para visualizar dados de um ponto de dados por dia ou por hora em um gráfico, passe o ponteiro do mouse sobre o ponto de dados.
- Nos gráficos que mostram os dados transferidos, é possível alterar a escala vertical para gigabytes, megabytes ou kilobytes.

Baixar dados no formato CSV

Você pode baixar o relatório de estatísticas do cache no formato CSV. Esta seção explica como fazer download do relatório e descreve os valores dele.

Para baixar o relatório de estatísticas do cache no formato CSV

1. No relatório de estatísticas de cache, selecione CSV.
2. Na caixa de diálogo Opening file name, opte por abrir ou salvar o arquivo.

Informações sobre o relatório

As primeiras linhas do relatório incluem as seguintes informações:

Versão

A versão do formato desse arquivo CSV.

Relatório

O nome do relatório.

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou ALL se o relatório tiver sido gerado para todas as distribuições.

StartDateUTC

O início do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

EndDateUTC

O término do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

GeneratedTimeUTC

A data e a hora nas quais você executou o relatório, no Tempo Universal Coordenado (UTC).

Granularity

Se cada linha do relatório representa uma hora ou um dia.

ViewerLocation

O continente de origem das solicitações do visualizador ou ALL se você optar por fazer download do relatório para todos os locais.

Dados do relatório de estatísticas do cache

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou ALL se o relatório tiver sido gerado para todas as distribuições.

FriendlyName

Um nome de domínio alternativo (CNAME), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

ViewerLocation

O continente de origem das solicitações do visualizador ou ALL se você optar por fazer download do relatório para todos os locais.

TimeBucket

A hora ou o dia ao qual os dados se aplicam, no Tempo Universal Coordenado (UTC).

RequestCount

O número total de solicitações de todos os códigos de status do HTTP (por exemplo, 200 ou 404) e de todos os métodos (por exemplo, GET, HEAD ou POST).

HitCount

O número de solicitações do visualizador atendidas pelo objeto de um ponto de presença de caches do CloudFront.

MissCount

O número de solicitações do visualizador para as quais o objeto não está em um ponto de presença de caches, de modo que o CloudFront precise obter o objeto de sua origem.

ErrorCount

O número de solicitações do visualizador que resultaram em erro, fazendo com que o CloudFront não fornecesse o objeto.

IncompleteDownloadCount

O número de solicitações para as quais o visualizador começou, mas não terminou de fazer download do objeto.

HTTP2xx

O número de solicitações do visualizador para as quais o código de status do HTTP é um valor 2xx (realizado).

HTTP3xx

O número de solicitações do visualizador para as quais o código de status do HTTP é um valor 3xx (ação adicional necessária).

HTTP4xx

O número de solicitações do visualizador para as quais o código de status do HTTP é um valor 4xx (erro do cliente).

HTTP5xx

O número de solicitações do visualizador para as quais o código de status do HTTP é um valor 5xx (erro do servidor).

TotalBytes

O número total de bytes enviados para os visualizadores pelo CloudFront em resposta a todas as solicitações de todos os métodos HTTP.

BytesFromMisses

O número de bytes enviados aos visualizadores para objetos que não estavam no ponto de presença de caches no momento da solicitação. Esse valor é uma boa aproximação dos bytes transferidos da origem aos pontos de presença de caches do CloudFront. No entanto, ele não inclui solicitações de objetos que já estão no ponto de presença de cache, mas expiraram.

Como os gráficos de estatísticas do cache estão relacionados aos dados nos logs padrão (logs de acesso) do CloudFront

A tabela a seguir mostra como os gráficos de estatísticas do cache do console do CloudFront correspondem aos valores dos logs de acesso do CloudFront. Para mais informações sobre os logs de acesso do CloudFront, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Total requests

Esse gráfico mostra o número total de solicitações de todos os códigos de status do HTTP (por exemplo, 200 ou 404) e de todos os métodos (por exemplo, GET, HEAD ou POST). O total de solicitações exibido nesse gráfico é igual ao número total de solicitações dos arquivos de log de acesso para o mesmo período.

Percentage of viewer requests by result type

Esse gráfico mostra o número de solicitações atendidas e não atendidas, e os erros como porcentagem do total de solicitações do visualizador para a distribuição selecionada do CloudFront:

- Hit (Atendidas): uma solicitação do visualizador para a qual o objeto é fornecido de um ponto de presença de caches do CloudFront. Nos logs de acesso, o valor de `x-edge-response-result-type` dessas solicitações é `Hit`.
- Miss (Não atendidas): uma solicitação do visualizador para a qual o objeto não está em um ponto de presença de caches, de modo que o CloudFront precise obtê-lo de sua origem. Nos logs de acesso, o valor de `x-edge-response-result-type` dessas solicitações é `Miss`.

- **Error (Erro):** uma solicitação do visualizador que resultou em erro, fazendo com que o CloudFront não fornecesse o objeto. Nos logs de acesso, o valor de `x-edge-response-result-type` dessas solicitações é `Error`, `LimitExceeded` ou `CapacityExceeded`.

O gráfico não inclui solicitações atendidas de atualização (solicitações de objetos que estão no ponto de presença de caches, mas expiraram). Nos logs de acesso, o valor de `x-edge-response-result-type` dessas solicitações é `RefreshHit`.

Bytes transferred to viewers

Esse gráfico mostra dois valores:

- **Total bytes (Total de bytes)** - o número total de bytes enviados para os visualizadores pelo CloudFront em resposta a todas as solicitações de todos os métodos HTTP. Nos logs de acesso do CloudFront, `Total Bytes (Total de bytes)` é a soma dos valores da coluna `sc-bytes` de todas as solicitações durante o mesmo período.
- **Bytes from misses (Bytes de perdas)** - o número de bytes enviados aos visualizadores para objetos que não estavam no ponto de presença de caches no momento da solicitação. Nos logs de acesso do CloudFront, `Bytes from misses (Bytes de perdas)` é a soma dos valores da coluna `sc-bytes` das solicitações para as quais o valor de `x-edge-result-type` é `Miss`. Esse valor é uma boa aproximação dos bytes transferidos da origem aos pontos de presença de caches do CloudFront. No entanto, ele não inclui solicitações de objetos que já estão no ponto de presença de cache, mas expiraram.

Códigos de status de HTTP

Esse gráfico mostra as solicitações do visualizador por código de status do HTTP. Nos logs de acesso do CloudFront, os códigos de status são exibidos na coluna `sc-status`:

- **2xx:** a solicitação foi bem-sucedida.
- **3xx:** ação adicional necessária. Por exemplo, 301 (Movido permanentemente) significa que o objeto solicitado foi movido para um local diferente.
- **4xx:** o cliente aparentemente cometeu um erro. Por exemplo, 404 (Não encontrado) significa que o cliente solicitou um objeto não encontrado.
- **5xx:** o servidor de origem não atendeu a solicitação. Por exemplo, 503 (Serviço indisponível) significa que o servidor de origem está indisponível no momento.

Percentage of GET requests that didn't finish downloading

Esse gráfico mostra as solicitações GET do visualizador que não concluíram o download do objeto solicitado como uma porcentagem do total de solicitações. Normalmente, o download de um

objeto não é concluído pois o visualizador cancelou o download, por exemplo, clicando em um link diferente ou fechando o navegador. Nos logs de acesso do CloudFront, essas solicitações têm valor 200 na coluna `sc-status` e valor `Error` na coluna `x-edge-result-type`.

Visualizar relatórios de objetos comuns do CloudFront

Consulte no relatório de objetos comuns do Amazon CloudFront para ver os 50 objetos mais comuns para uma distribuição durante um intervalo de datas específico nos últimos 60 dias. Também é possível visualizar estatísticas sobre esses objetos, incluindo o seguinte:

- Número de solicitações para o objeto
- Número de acertos e erros
- Hit ratio
- Número de bytes fornecidos para erros
- Total de bytes fornecidos
- Número de downloads incompletos
- Número de solicitações por código de status HTTP (2xx, 3xx, 4xx e 5xx)

Os dados dessas estatísticas são obtidos da mesma origem que os logs de acesso do CloudFront. No entanto, não é necessário habilitar o [registro em log de acesso](#) para visualizar objetos populares.

Tópicos

- [Visualizar relatórios de objetos comuns do CloudFront no console](#)
- [Como o CloudFront calcula estatísticas de objetos comuns](#)
- [Baixar dados no formato CSV](#)
- [Como os dados no relatório de objetos populares estão relacionados aos dados dos logs padrão \(logs de acesso\) do CloudFront](#)

Visualizar relatórios de objetos comuns do CloudFront no console

É possível visualizar o relatório de objetos comuns do CloudFront no console.

Como visualizar objetos comuns para uma distribuição do CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Objetos comuns.
3. No painel CloudFront Popular Objects Report (Relatório de objetos populares do CloudFront), em Start Date (Data de início) e End Date (Data de término), selecione o intervalo de datas para o qual você deseja exibir uma lista de objetos populares. Você pode escolher qualquer intervalo de datas nos últimos 60 dias.

As datas e horas estão em Tempo Universal Coordenado (UTC).

4. Na lista Distribution, selecione a distribuição para a qual você deseja exibir uma lista de objetos populares.
5. Selecione Atualizar.

Como o CloudFront calcula estatísticas de objetos comuns

Para obter uma contagem precisa dos 50 principais objetos na distribuição, o CloudFront soma as solicitações de todos os objetos em intervalos de 10 minutos a partir da meia-noite e mantém um total de execução dos 150 principais objetos pelas próximas 24 horas. (O CloudFront também mantém totais diários dos 150 principais objetos por 60 dias.)

Na parte inferior da lista, os objetos constantemente sobem de posição ou saem da lista. Portanto, os totais desses objetos são aproximados. Os 50 objetos na parte superior da lista de 150 objetos podem subir ou cair de posição, mas raramente saem da lista. Portanto, os totais desses objetos são mais confiáveis.

Quando um objeto sai da lista dos 150 principais objetos e volta para a lista novamente em um dia específico, o CloudFront adiciona um número estimado de solicitações para o período em que o objeto não estava na lista. A estimativa é baseada no número de solicitações recebidas por qualquer objeto que estava na parte inferior da lista durante esse período.

Se o objeto subir para as 50 primeiras posições posteriormente nesse mesmo dia, as estimativas do número de solicitações recebidas pelo CloudFront enquanto o objeto não estava na lista dos primeiros 150 objetos normalmente faz com que o número de solicitações do relatório de objetos populares ultrapasse o número de solicitações exibido nos logs de acesso desse objeto.

Baixar dados no formato CSV

Você pode baixar o relatório de objetos populares no formato CSV. Esta seção explica como fazer download do relatório e descreve os valores dele.

Para baixar o relatório de objetos populares no formato CSV

1. No relatório de objetos comuns, selecione CSV.
2. Na caixa de diálogo Opening file name, opte por abrir ou salvar o arquivo.

Informações sobre o relatório

As primeiras linhas do relatório incluem as seguintes informações:

Versão

A versão do formato desse arquivo CSV.

Relatório

O nome do relatório.

DistributionID

O ID da distribuição para a qual você executou o relatório.

StartDateUTC

O início do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

EndDateUTC

O término do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

GeneratedTimeUTC

A data e a hora nas quais você executou o relatório, no Tempo Universal Coordenado (UTC).

Dados do relatório de objetos populares

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual você executou o relatório.

FriendlyName

Um nome de domínio alternativo (CNAME), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

Objeto

Os últimos 500 caracteres do URL do objeto.

RequestCount

O número total de solicitações desse objeto.

HitCount

O número de solicitações do visualizador atendidas pelo objeto de um ponto de presença de caches do CloudFront.

MissCount

O número de solicitações do visualizador para as quais o objeto não está em um ponto de presença de caches, de modo que o CloudFront precise obter o objeto de sua origem.

HitCountPct

O valor de `HitCount` como porcentagem do valor de `RequestCount`.

BytesFromMisses

O número de bytes enviados para os visualizadores para essa objeto que ele não estava no ponto de presença de caches no momento da solicitação.

TotalBytes

O número total de bytes enviados para os visualizadores pelo CloudFront para esse objeto em resposta a todas as solicitações de todos os métodos HTTP.

IncompleteDownloadCount

O número de solicitações desse objeto para as quais o visualizador começou, mas não terminou de fazer download do objeto.

HTTP2xx

O número de solicitações do visualizador para as quais o código de status do HTTP é um valor 2xx (realizado).

HTTP3xx

O número de solicitações do visualizador para as quais o código de status do HTTP é um valor 3xx (ação adicional necessária).

HTTP4xx

O número de solicitações do visualizador para as quais o código de status do HTTP é um valor 4xx (erro do cliente).

HTTP5xx

O número de solicitações do visualizador para as quais o código de status do HTTP é um valor 5xx (erro do servidor).

Como os dados no relatório de objetos populares estão relacionados aos dados dos logs padrão (logs de acesso) do CloudFront

A lista a seguir mostra como os valores do relatório de objetos populares do console do CloudFront correspondem aos valores dos logs de acesso do CloudFront. Para mais informações sobre os logs de acesso do CloudFront, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

URL

Os últimos 500 caracteres do URL que os visualizadores usam para acessar o objeto.

Solicitações

O número total de solicitações do objeto. Esse valor normalmente corresponde ao número de solicitações GET do objeto no logs de acesso do CloudFront.

Hits

O número de solicitações do visualizador atendidas pelo objeto de um de ponto de presença de caches do CloudFront. Nos logs de acesso, o valor de `x-edge-response-result-type` dessas solicitações é `Hit`.

Misses

O número de solicitações do visualizador para as quais o objeto não estava em um de ponto de presença de caches, fazendo com que o CloudFront recuperasse o objeto de sua origem. Nos logs de acesso, o valor de `x-edge-response-result-type` dessas solicitações é `Miss`.

Hit ratio

O valor da coluna Hits como porcentagem do valor da coluna Requests.

Bytes from misses

O número de bytes enviados aos visualizadores para objetos que não estavam no ponto de presença de caches no momento da solicitação. Nos logs de acesso do CloudFront, Bytes from misses (Bytes de perdas) é a soma dos valores da coluna `sc-bytes` das solicitações para as quais o valor de `x-edge-result-type` é Miss.

Total bytes

O número total de bytes enviados para os visualizadores pelo CloudFront em resposta a todas as solicitações do objeto de todos os métodos HTTP. Nos logs de acesso do CloudFront, Total Bytes (Total de bytes) é a soma dos valores da coluna `sc-bytes` de todas as solicitações durante o mesmo período.

Incomplete downloads

O número de solicitações do visualizador que não concluíram o download do objeto solicitado. Normalmente, um download não é concluído pois o visualizador o cancelou, por exemplo, clicando em um link diferente ou fechando o navegador. Nos logs de acesso do CloudFront, essas solicitações têm valor `200` na coluna `sc-status` e valor `Error` na coluna `x-edge-result-type`.

2xx

O número de solicitações para as quais o código de status do HTTP é `2xx`, `Successful`. Nos logs de acesso do CloudFront, os códigos de status são exibidos na coluna `sc-status`.

3xx

O número de solicitações para as quais o código de status do HTTP é `3xx`, `Redirection`. Os códigos de status `3xx` indicam que uma ação adicional é necessária. Por exemplo, `301` (Movido permanentemente) significa que o objeto solicitado foi movido para um local diferente.

4xx

O número de solicitações para as quais o código de status do HTTP é `4xx`, `Client Error`. Os códigos de status `4xx` indicam que o cliente aparentemente cometeu um erro. Por exemplo, `404` (Não encontrado) significa que o cliente solicitou um objeto não encontrado.

5xx

O número de solicitações para as quais o código de status do HTTP é 5xx, `Server Error`. Os códigos de status 5xx indicam que o servidor de origem não atendeu a solicitação. Por exemplo, 503 (Serviço indisponível) significa que o servidor de origem está indisponível no momento.

Visualizar relatórios dos principais indicadores do CloudFront

O relatório dos principais indicadores do CloudFront inclui o seguinte para qualquer intervalo de datas nos últimos 60 dias:

- Os 25 principais indicadores (domínios de sites que originaram a maioria das solicitações HTTP e HTTPS referentes a objetos distribuídos pelo CloudFront para a distribuição)
- Número de solicitações de um indicador
- Número de solicitações de um indicador como porcentagem do número total de solicitações durante o período especificado

Os dados do relatório de principais indicadores são obtidos da mesma origem que os logs de acesso do CloudFront. No entanto, não é necessário habilitar o [registro em log de acesso](#) para visualizar os principais indicadores.

Os principais indicadores podem ser mecanismos de pesquisa, outros sites com link direto aos seus objetos ou seu próprio site. Por exemplo, se `https://example.com/index.html` tiver links para dez elementos gráficos, `example.com` será o indicador de todos eles.

Note

Se um usuário inserir um URL diretamente na linha de endereço do navegador, não haverá indicador para o objeto solicitado.

Tópicos

- [Visualizar relatórios dos principais indicadores do CloudFront no console](#)
- [Como o CloudFront calcula as estatísticas dos principais indicadores](#)
- [Baixar dados no formato CSV](#)
- [Como os dados do relatório de principais indicadores estão relacionados aos dados dos logs padrão \(logs de acesso\) do CloudFront](#)

Visualizar relatórios dos principais indicadores do CloudFront no console

É possível visualizar o relatório dos principais indicadores do CloudFront no console.

Como visualizar os principais indicadores referentes a uma distribuição do CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Principais indicadores.
3. No painel CloudFront Top Referrers Report (Relatório de principais indicadores do CloudFront), em Start Date (Data de início) e End Date (Data de término), selecione o intervalo de datas para o qual você deseja exibir uma lista dos principais indicadores.

As datas e horas estão em Tempo Universal Coordenado (UTC).

4. Na lista Distribution, selecione a distribuição para a qual você deseja exibir uma lista dos principais indicadores.
5. Selecione Atualizar.

Como o CloudFront calcula as estatísticas dos principais indicadores

Para obter uma contagem precisa dos 25 principais indicadores, o CloudFront soma a solicitações de todos os seus objetos em intervalos de 10 minutos e mantém um total de execução dos 75 principais indicadores. Na parte inferior da lista, os indicadores constantemente sobem de posição ou saem da lista. Portanto, os totais desses indicadores são aproximados.

Os 25 indicadores na parte superior da lista de 75 indicadores podem subir ou cair de posição, mas raramente saem da lista. Portanto, os totais desses indicadores costumam ser mais confiáveis.

Baixar dados no formato CSV

Você pode baixar o relatório de principais indicadores no formato CSV. Esta seção explica como fazer download do relatório e descreve os valores dele.

Para baixar o relatório de principais indicadores no formato CSV

1. No relatório dos principais indicadores, selecione CSV.
2. Na caixa de diálogo Opening file name, opte por abrir ou salvar o arquivo.

Informações sobre o relatório

As primeiras linhas do relatório incluem as seguintes informações:

Versão

A versão do formato desse arquivo CSV.

Relatório

O nome do relatório.

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou ALL se o relatório tiver sido gerado para todas as distribuições.

StartDateUTC

O início do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

EndDateUTC

O término do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

GeneratedTimeUTC

A data e a hora nas quais você executou o relatório, no Tempo Universal Coordenado (UTC).

Dados do relatório de principais indicadores

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou ALL se o relatório tiver sido gerado para todas as distribuições.

FriendlyName

Um nome de domínio alternativo (CNAME), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

Referrer

O nome de domínio do indicador.

RequestCount

O número total de solicitações do nome de domínio na coluna `Referrer`.

RequestsPct

O número de solicitações enviadas por um indicador como porcentagem do número total de solicitações durante o período especificado.

Como os dados do relatório de principais indicadores estão relacionados aos dados dos logs padrão (logs de acesso) do CloudFront

A lista a seguir mostra como os valores do relatório de principais indicadores do console do CloudFront correspondem aos valores dos logs de acesso do CloudFront. Para mais informações sobre os logs de acesso do CloudFront, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Referrer

O nome de domínio do indicador. Nos logs de acesso, os indicadores estão indicados na coluna `cs(Referrer)`.

Request count

O número total de solicitações do nome de domínio na coluna `Referrer`. Esse valor normalmente corresponde ao número de solicitações GET do indicador no logs de acesso do CloudFront.

Solicitação %

O número de solicitações enviadas por um indicador como porcentagem do número total de solicitações durante o período especificado. Se você tiver mais de 25 indicadores, não será possível calcular a % de solicitações com base nos dados dessa tabela, porque a coluna `Request count` (Contagem de solicitações) não inclui todas as solicitações durante o período especificado.

Visualizar relatórios de uso do CloudFront

Os relatórios de uso do CloudFront incluem as seguintes informações:

- **Number of requests 0** (Número de solicitações) mostra o número total de solicitações atendidas pelo CloudFront de locais de borda na região selecionada durante cada intervalo de tempo para a distribuição do CloudFront especificada.

- Data transferred by protocol (Dados transferidos por protocolo) e Data transferred by destination (Dados transferidos por destino) - ambos mostram a quantidade total de dados transferidos de locais de borda do CloudFront na região selecionada durante cada intervalo de tempo para a distribuição do CloudFront especificada. Eles separam os dados de forma diferente, da seguinte maneira:
 - By protocol (Por protocolo) - separa os dados por protocolo: HTTP ou HTTPS.
 - Por destino: separa os dados por destino: para os visualizadores ou para a origem.

O relatório de uso do CloudFront é baseado no relatório de uso da AWS para o CloudFront. Esse relatório não exige configuração adicional. Para ter mais informações, consulte [Visualizar o relatório de uso da AWS para o CloudFront](#).

É possível visualizar relatórios referentes a um intervalo de datas nos últimos 60 dias, com pontos de dados por hora ou dia. Normalmente, é possível visualizar os dados das solicitações recebidas pelo CloudFront há quatro horas, mas eles podem atrasar até 24 horas.

Para ter mais informações, consulte [Como os gráficos de uso estão relacionados aos dados do relatório de uso do CloudFront](#).

Tópicos

- [Visualizar relatórios de uso do CloudFront no console](#)
- [Baixar dados no formato CSV](#)
- [Como os gráficos de uso estão relacionados aos dados do relatório de uso do CloudFront](#)

Visualizar relatórios de uso do CloudFront no console

É possível visualizar o relatório de uso do CloudFront no console.

Como visualizar relatórios de uso do CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Usage Reports (Relatórios de uso).
3. No painel CloudFront Usage Reports (Relatórios de uso do CloudFront), em Start Date (Data de início) e End Date (Data de término), selecione o intervalo de datas para o qual você deseja exibir gráficos de uso. Os intervalos disponíveis dependem do valor selecionado para Granularity:

- **Daily (Diariamente):** para exibir gráficos com um ponto de dados por dia, selecione qualquer intervalo de datas nos últimos 60 dias.
- **Hourly (Por hora):** para exibir gráficos com um ponto de dados por hora, selecione qualquer intervalo de datas de até 14 dias nos últimos 60 dias.

As datas e horas estão em Tempo Universal Coordenado (UTC).

4. Em **Granularity**, especifique se você deseja exibir um ponto de dados por dia ou por hora nos gráficos. Se você especificar um intervalo de datas maior que 14 dias, a opção para especificar um ponto de dados por hora não estará disponível.
5. Em **Billing Region (Região de faturamento)**, escolha a região de faturamento do CloudFront com os dados que você deseja visualizar ou escolha **All Regions (Todas as regiões)**. Os gráficos de uso incluem dados de solicitações processados pelo CloudFront em pontos de presença na região especificada. A região na qual o CloudFront processa solicitações pode ou não corresponder à localização dos visualizadores.

Selecione apenas regiões incluídas na classe de preço da sua distribuição. Caso contrário, os gráficos de uso provavelmente não conterão dados. Por exemplo, se você escolher a classe de preço 200 para sua distribuição, as regiões de faturamento América do Sul e Austrália não serão incluídas, portanto, o CloudFront não processará as solicitações dessas regiões. Para ter mais informações sobre as classes de preços, consulte [Definição de preço do CloudFront](#).

6. Na lista **Distribution**, selecione as distribuições para as quais você deseja exibir os dados nos gráficos de uso:
 - **An individual distribution (uma distribuição individual):** os gráficos exibem os dados da distribuição selecionada do CloudFront. A lista **Distribution** exibe o ID e os nomes de domínio alternativos (CNAMEs) da distribuição, se houver. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá os nomes de domínio de origem dela.
 - **All distributions (excludes deleted) (Todas as distribuições (exceto excluídas)):** os gráficos exibem a soma dos dados de todas as distribuições que estão associadas à conta atual da AWS, com exceção das distribuições que você excluiu.
 - **All Deleted Distributions (Todas as distribuições excluídas):** os gráficos exibem a soma dos dados de todas as distribuições que estão associadas à conta atual da AWS e que foram excluídas nos últimos 60 dias.
7. Escolha **Atualizar grafos**.

 Tip

- Para visualizar dados de um ponto de dados por dia ou por hora em um gráfico, passe o ponteiro do mouse sobre o ponto de dados.
- Nos gráficos que mostram os dados transferidos, é possível alterar a escala vertical para gigabytes, megabytes ou kilobytes.

Baixar dados no formato CSV

É possível baixar o relatório de uso no formato CSV. Esta seção explica como fazer download do relatório e descreve os valores dele.

Para baixar o relatório de uso no formato CSV

1. No relatório de uso, selecione CSV.
2. Na caixa de diálogo Opening file name, opte por abrir ou salvar o arquivo.

Informações sobre o relatório

As primeiras linhas do relatório incluem as seguintes informações:

Versão

A versão do formato desse arquivo CSV.

Relatório

O nome do relatório.

DistributionID

O ID da distribuição para a qual o relatório foi gerado, ALL se o relatório tiver sido gerado para todas as distribuições ou ALL_DELETED se o relatório tiver sido gerado para todas as distribuições excluídas.

StartDateUTC

O início do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

EndDateUTC

O término do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

GeneratedTimeUTC

A data e a hora nas quais você executou o relatório, no Tempo Universal Coordenado (UTC).

Granularity

Se cada linha do relatório representa uma hora ou um dia.

BillingRegion

O continente de origem das solicitações do visualizador ou ALL se você optar por fazer download do relatório para todas as regiões de faturamento.

Dados do relatório de uso

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado, ALL se o relatório tiver sido gerado para todas as distribuições ou ALL_DELETED se o relatório tiver sido gerado para todas as distribuições excluídas.

FriendlyName

Um nome de domínio alternativo (CNAME), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

BillingRegion

A região de faturamento do CloudFront executada no relatório ou ALL.

TimeBucket

A hora ou o dia ao qual os dados se aplicam, no Tempo Universal Coordenado (UTC).

HTTP

O número de solicitações HTTP atendidas pelo CloudFront de pontos de presença na região selecionada durante cada intervalo de tempo para a distribuição do CloudFront específica. Os valores incluem:

- O número de solicitações GET e HEAD, que fazem com que o CloudFront transfira dados para os visualizadores
- O número de solicitações DELETE, OPTIONS, PATCH, POST e PUT que fazem com que o CloudFront transfira dados para sua origem

HTTPS

O número de solicitações HTTPS atendidas pelo CloudFront de pontos de presença na região selecionada durante cada intervalo de tempo para a distribuição do CloudFront específica. Os valores incluem:

- O número de solicitações GET e HEAD, que fazem com que o CloudFront transfira dados para os visualizadores
- O número de solicitações DELETE, OPTIONS, PATCH, POST e PUT que fazem com que o CloudFront transfira dados para sua origem

HTTPBytes

A quantidade total de dados transferidos por HTTP de pontos de presença do CloudFront na região de faturamento selecionada durante o período para a distribuição específica do CloudFront. Os valores incluem:

- Os dados transferidos do CloudFront para os visualizadores em resposta às solicitações GET e HEAD
- Dados transferidos dos visualizadores ao CloudFront para solicitações DELETE, OPTIONS, PATCH, POST e PUT
- Os dados transferidos do CloudFront para os visualizadores em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT.

HTTPSBytes

A quantidade total de dados transferidos por HTTPS de pontos de presença do CloudFront na região de faturamento selecionada durante o período para a distribuição específica do CloudFront. Os valores incluem:

- Os dados transferidos do CloudFront para os visualizadores em resposta às solicitações GET e HEAD
- Dados transferidos dos visualizadores ao CloudFront para solicitações DELETE, OPTIONS, PATCH, POST e PUT
- Os dados transferidos do CloudFront para os visualizadores em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT.

BytesIn

A quantidade total de dados transferidos do CloudFront para sua origem para as solicitações DELETE, OPTIONS, PATCH, POST e PUT na região selecionada durante cada intervalo de tempo para a distribuição específica do CloudFront.

BytesOut

O volume total de dados transferidos por HTTP e HTTPS do CloudFront para os visualizadores na região selecionada durante cada intervalo de tempo para a distribuição específica do CloudFront. Os valores incluem:

- Os dados transferidos do CloudFront para os visualizadores em resposta às solicitações GET e HEAD
- Os dados transferidos do CloudFront para os visualizadores em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT.

Como os gráficos de uso estão relacionados aos dados do relatório de uso do CloudFront

A lista a seguir mostra como o uso gráficos no console do CloudFront correspondem aos valores da coluna Usage Type (Tipo de uso) no relatório de uso do CloudFront.

Tópicos

- [Number of requests](#)
- [Data transferred by protocol](#)
- [Data transferred by destination](#)

Number of requests

Esse gráfico mostra o número total de solicitações atendidas pelo CloudFront de pontos de presença na região selecionada durante cada intervalo de tempo para a distribuição do CloudFront especificada, separadas por protocolo (HTTP ou HTTPS) e por tipo (estáticas, dinâmicas ou de proxy).

Number of HTTP requests

- *region*-Requests-HTTP-Static: número de solicitações GET e HEAD HTTP atendidas para objetos com TTL \geq 3600 segundos

- *region*-Requests-HTTP-Dynamic: número de solicitações GET e HEAD HTTP atendidas para objetos com TTL < 3.600 segundos
- *region*-Requests-HTTP-Proxy: número de solicitações DELETE, OPTIONS, PATCH, POST e HTTP PUT encaminhado pelo CloudFront à sua origem

Number of HTTPS requests

- *region*-Requests-HTTPS-Static: número de solicitações GET and HEAD HTTPS atendidas para objetos com TTL \geq 3600 segundos
- *region*-Requests-HTTPS-Dynamic: Número de solicitações GET e HEAD HTTPS atendidas para objetos com TTL < 3.600 segundos
- *region*-Requests-HTTPS-Proxy: número de solicitações DELETE, OPTIONS, PATCH, POST e PUT HTTPS que o CloudFront encaminha à sua origem

Data transferred by protocol

Esse grafo mostra o volume total de dados transferidos dos locais da borda do CloudFront na região selecionada durante cada intervalo de tempo para a distribuição especificada do CloudFront, separadas por protocolo (HTTP ou HTTPS), por tipo (estáticas, dinâmicas ou de proxy) e por destino (visualizadores ou origem).

Data transferred over HTTP

- *region*-Out-Bytes-HTTP-Static: bytes fornecidos por HTTP para objetos com TTL \geq 3600 segundos
- *region*-Out-Bytes-HTTP-Dynamic: bytes fornecidos por HTTP para objetos com TTL < 3.600 segundos
- *region*-Out-Bytes-HTTP-Proxy: bytes retornados por HTTP do CloudFront para os visualizadores em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT
- *region*-Out-Bytes-HTTP-Proxy: total de bytes transferidos por HTTP de pontos de presença do CloudFront para sua origem em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT

Data transferred over HTTPS

- *region*-Out-Bytes-HTTPS-Static: bytes fornecidos por HTTPS para objetos com TTL \geq 3600 segundos
- *region*-Out-Bytes-HTTPS-Dynamic: Bytes fornecidos por HTTPS para objetos com TTL < 3.600 segundos

- *region*-Out-Bytes-HTTPS-Proxy: bytes retornados por HTTPS do CloudFront para os visualizadores em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT
- *region*-Out-OBytes-HTTPS-Proxy: total de bytes transferidos por HTTPS de pontos de presença do CloudFront para sua origem em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT

Data transferred by destination

Esse grafo mostra o volume total de dados transferidos dos locais da borda do CloudFront na região selecionada durante cada intervalo de tempo para a distribuição especificada do CloudFront, separadas por destino (visualizadores ou origem), protocolo (HTTP ou HTTPS) e tipo (estáticas, dinâmicas ou de proxy).

Dados transferidos do CloudFront para os visualizadores

- *region*-Out-Bytes-HTTP-Static: bytes fornecidos por HTTP para objetos com TTL \geq 3600 segundos
- *region*-Out-Bytes-HTTPS-Static: bytes fornecidos por HTTPS para objetos com TTL \geq 3600 segundos
- *region*-Out-Bytes-HTTP-Dynamic: bytes fornecidos por HTTP para objetos com TTL $<$ 3.600 segundos
- *region*-Out-Bytes-HTTPS-Dynamic: Bytes fornecidos por HTTPS para objetos com TTL $<$ 3.600 segundos
- *region*-Out-Bytes-HTTP-Proxy: bytes retornados por HTTP do CloudFront para os visualizadores em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT
- *region*-Out-Bytes-HTTPS-Proxy: bytes retornados por HTTPS do CloudFront para os visualizadores em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT

Dados transferidos do CloudFront para a sua origem

- *region*-Out-OBytes-HTTP-Proxy: total de bytes transferidos por HTTP de pontos de presença do CloudFront para sua origem em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT
- *region*-Out-OBytes-HTTPS-Proxy: total de bytes transferidos por HTTPS de pontos de presença do CloudFront para sua origem em resposta às solicitações DELETE, OPTIONS, PATCH, POST e PUT

Visualizar relatórios de visualizadores do CloudFront

Os relatórios de visualizadores do CloudFront incluem as seguintes informações para qualquer intervalo de datas nos últimos 60 dias:

- **Dispositivos:** os tipos de dispositivos usados com maior frequência para acessar seu conteúdo (como desktop ou dispositivo móvel).
- **Navegadores:** os dez navegadores mais usados para acessar seu conteúdo (como Chrome ou Firefox).
- **Sistemas operacionais:** os dez sistemas operacionais mais usados ao acessar seu conteúdo (como Linux, macOS ou Windows).
- **Locais:** os 50 principais locais (países ou estados/territórios dos EUA) dos visualizadores que acessam seu conteúdo com maior frequência.
 - Também é possível visualizar locais com pontos de dados por hora de qualquer intervalo de datas de até 14 dias nos últimos 60 dias.

Note

Não é necessário habilitar o [registro em log de acesso](#) para ver os gráficos e relatórios de visualizadores.

Tópicos

- [Visualizar gráficos e relatórios de visualizadores no console](#)
- [Baixar dados no formato CSV](#)
- [Dados incluídos nos relatórios de visualizadores](#)
- [Como os dados do relatório de locais estão relacionados aos dados dos logs padrão \(logs de acesso\) do CloudFront](#)

Visualizar gráficos e relatórios de visualizadores no console

É possível visualizar gráficos e relatórios de visualizadores do CloudFront no console.

Como visualizar gráficos e relatórios de visualizadores do CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Visualizadores.
3. No painel CloudFront Viewers (Visualizadores do CloudFront) em Start Date (Data de início) e End Date (Data de término), selecione o intervalo de datas para o qual você deseja exibir gráficos e relatórios de visualizadores.

No gráfico de locais, os intervalos disponíveis dependem do valor selecionado para Granularity:

- Daily (Diariamente): para exibir gráficos com um ponto de dados por dia, selecione qualquer intervalo de datas nos últimos 60 dias.
- Hourly (Por hora): para exibir gráficos com um ponto de dados por hora, selecione qualquer intervalo de datas de até 14 dias nos últimos 60 dias.

As datas e horas estão em Tempo Universal Coordenado (UTC).

4. (Somente nos gráficos de navegadores e sistemas operacionais) Em Grouping, especifique se você deseja agrupar os navegadores e sistemas operacionais por nome (Chrome, Firefox) ou por nome e versão (Chrome 40.0, Firefox 35.0).
5. (Somente no gráfico de locais) Em Granularity, especifique se você deseja exibir um ponto de dados por dia ou por hora nos gráficos. Se você especificar um intervalo de datas maior que 14 dias, a opção para especificar um ponto de dados por hora não estará disponível.
6. (Somente no gráfico de locais) Em Details, especifique se você deseja exibir os principais locais por país ou estado dos EUA.
7. Na lista Distribution, selecione a distribuição para a qual você deseja exibir os dados nos gráficos de uso:
 - An individual distribution (uma distribuição individual): os gráficos exibem os dados da distribuição selecionada do CloudFront. A lista Distribution exibe o ID e um nome de domínio alternativo (CNAME) da distribuição, se houver. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.
 - All distributions (excludes deleted) (Todas as distribuições (exceto excluídas)): os gráficos exibem a soma dos dados de todas as distribuições que estão associadas à conta atual da AWS, com exceção das distribuições que você excluiu.
8. Selecione Atualizar.

Para visualizar dados de um ponto de dados por dia ou por hora em um gráfico, passe o ponteiro do mouse sobre o ponto de dados.

Baixar dados no formato CSV

Você pode baixar cada um dos relatórios de visualizadores no formato CSV. Esta seção explica como fazer download dos relatórios e descreve os valores dele.

Para baixar o relatório de visualizadores no formato CSV

1. No relatório de visualizadores, selecione CSV.
2. Escolha os dados que você deseja baixar, por exemplo, Devices ou Devices Trends.
3. Na caixa de diálogo Opening file name, opte por abrir ou salvar o arquivo.

Dados incluídos nos relatórios de visualizadores

As primeiras linhas de cada relatório incluem as seguintes informações:

Version (Versão)

A versão do formato desse arquivo CSV.

Relatório

O nome do relatório.

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou ALL se o relatório tiver sido gerado para todas as distribuições.

StartDateUTC

O início do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

EndDateUTC

O término do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

GeneratedTimeUTC

A data e a hora nas quais você executou o relatório, no Tempo Universal Coordenado (UTC).

Agrupamento (somente relatórios de navegadores e sistemas operacionais)

Se os dados são agrupados por nome ou por nome e versão do navegador ou sistema operacional.

Granularity

Se cada linha do relatório representa uma hora ou um dia.

Detalhes (somente relatório de locais)

Se as solicitações são indicadas por país ou estado dos EUA.

Os tópicos a seguir descrevem as informações nos diferentes relatórios de visualizadores.

Tópicos

- [Relatório de dispositivos](#)
- [Relatório de tendências do dispositivo](#)
- [Relatório de navegadores](#)
- [Relatório de tendências do navegador](#)
- [Relatório de sistemas operacionais](#)
- [Relatório de tendências do sistema operacional](#)
- [Relatório de locais](#)
- [Relatório de tendências do local](#)

Relatório de dispositivos

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou ALL se o relatório tiver sido gerado para todas as distribuições.

FriendlyName

Um nome de domínio alternativo (CNAME), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

Solicitações

O número de solicitações recebidas pelo CloudFront por cada tipo de dispositivo.

RequestsPct

O número de solicitações recebidas pelo CloudFront de cada tipo de dispositivo como porcentagem do número total de solicitações recebidas pelo CloudFront de todos os dispositivos.

Personalizar

As solicitações para as quais o valor do cabeçalho `User-Agent HTTP` não foi associado a um dos tipos de dispositivo padrão, por exemplo, `Desktop` ou `Mobile`.

Relatório de tendências do dispositivo

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou `ALL` se o relatório tiver sido gerado para todas as distribuições.

FriendlyName

Um nome de domínio alternativo (`CNAME`), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

TimeBucket

A hora ou o dia ao qual os dados se aplicam, no Tempo Universal Coordenado (UTC).

Desktop

O número de solicitações recebidas pelo CloudFront de desktops durante o período.

Dispositivos móveis

O número de solicitações recebidas pelo CloudFront de dispositivos móveis durante o período. Dispositivos móveis podem incluir tablets e celulares. Se o CloudFront não conseguir determinar se uma solicitação foi feita de um celular ou tablet, ela é inserida na coluna `Mobile`.

Smart-TV

O número de solicitações recebidas pelo CloudFront de Smart TVs durante o período.

Tablet

O número de solicitações recebidas pelo CloudFront de tablets durante o período. Se o CloudFront não conseguir determinar se uma solicitação foi feita de um celular ou tablet, ela é inserida na coluna `Mobile`.

Desconhecido

As solicitações para as quais o valor do cabeçalho `User-Agent HTTP` não foi associado a um dos tipos de dispositivo padrão, por exemplo, `Desktop` ou `Mobile`.

Empty

O número de solicitações recebidas pelo CloudFront que não incluem um valor no cabeçalho `User-Agent HTTP` durante o período.

Relatório de navegadores

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou `ALL` se o relatório tiver sido gerado para todas as distribuições.

FriendlyName

Um nome de domínio alternativo (`CNAME`), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

Grupo

O navegador ou o navegador e a versão do qual o CloudFront recebeu solicitações, dependendo do valor de `Grouping`. Além dos nomes dos navegadores, os valores possíveis incluem:

- `Bot/Crawler`: solicitações principalmente de mecanismos de pesquisa que estiverem indexando seu conteúdo.
- `Empty (Vazio)`: solicitações para as quais o valor do cabeçalho `User-Agent HTTP` está vazio.
- `Other (Outro)`: navegadores identificados pelo CloudFront, mas que não estão entre os mais populares. Se `Bot/Crawler`, `Empty` e/ou `Unknown` não for exibido entre os nove primeiros valores, eles também serão incluídos em `Other`.

- **Unknown (Desconhecido):** solicitações para as quais o valor do cabeçalho `User-Agent HTTP` não foi associado a um navegador padrão. A maioria das solicitações dessa categoria são provenientes de aplicativos ou scripts personalizados.

Solicitações

O número de solicitações recebidas pelo CloudFront por cada tipo de navegador.

RequestsPct

O número de solicitações recebidas pelo CloudFront de cada tipo de navegador como porcentagem do número total de solicitações recebidas pelo CloudFront durante o período especificado.

Relatório de tendências do navegador

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou `ALL` se o relatório tiver sido gerado para todas as distribuições.

FriendlyName

Um nome de domínio alternativo (`CNAME`), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

TimeBucket

A hora ou o dia ao qual os dados se aplicam, no Tempo Universal Coordenado (UTC).

(Navegadores)

As demais colunas do relatório indicam os navegadores ou os navegadores e suas versões, dependendo do valor de `Grouping`. Além dos nomes dos navegadores, os valores possíveis incluem:

- **Bot/Crawler:** solicitações principalmente de mecanismos de pesquisa que estiverem indexando seu conteúdo.
- **Empty (Vazio):** solicitações para as quais o valor do cabeçalho `User-Agent HTTP` está vazio.
- **Other (Outro):** navegadores identificados pelo CloudFront, mas que não estão entre os mais populares. Se `Bot/Crawler`, `Empty` e/ou `Unknown` não for exibido entre os nove primeiros valores, eles também serão incluídos em `Other`.

- **Unknown (Desconhecido):** solicitações para as quais o valor do cabeçalho `User-Agent HTTP` não foi associado a um navegador padrão. A maioria das solicitações dessa categoria são provenientes de aplicativos ou scripts personalizados.

Relatório de sistemas operacionais

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou `ALL` se o relatório tiver sido gerado para todas as distribuições.

FriendlyName

Um nome de domínio alternativo (`CNAME`), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

Grupo

O sistema operacional ou o sistema operacional e a versão do qual o CloudFront recebeu solicitações, dependendo do valor de `Grouping`. Além dos nomes dos sistemas operacionais, os valores possíveis incluem:

- **Bot/Crawler:** solicitações principalmente de mecanismos de pesquisa que estiverem indexando seu conteúdo.
- **Empty (Vazio):** solicitações para as quais o valor do cabeçalho `User-Agent HTTP` está vazio.
- **Other (Outro):** sistemas operacionais identificados pelo CloudFront, mas que não estão entre os mais populares. Se `Bot/Crawler`, `Empty` e/ou `Unknown` não for exibido entre os nove primeiros valores, eles também serão incluídos em `Other`.
- **Unknown (Desconhecido):** solicitações para as quais o valor do cabeçalho `User-Agent HTTP` não foi associado a um navegador padrão. A maioria das solicitações dessa categoria são provenientes de aplicativos ou scripts personalizados.

Solicitações

O número de solicitações recebidas pelo CloudFront por cada tipo de sistema operacional.

RequestsPct

O número de solicitações recebidas pelo CloudFront de cada tipo de sistema operacional, como porcentagem do número total de solicitações recebidas pelo CloudFront durante o período especificado.

Relatório de tendências do sistema operacional

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou ALL se o relatório tiver sido gerado para todas as distribuições.

FriendlyName

Um nome de domínio alternativo (CNAME), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

TimeBucket

A hora ou o dia ao qual os dados se aplicam, no Tempo Universal Coordenado (UTC).

(Sistemas operacionais)

As demais colunas do relatório indicam os sistemas operacionais ou os sistemas operacionais e suas versões, dependendo do valor de `Grouping`. Além dos nomes dos sistemas operacionais, os valores possíveis incluem:

- `Bot/Crawler`: solicitações principalmente de mecanismos de pesquisa que estiverem indexando seu conteúdo.
- `Empty (Vazio)`: solicitações para as quais o valor do cabeçalho `User-Agent HTTP` está vazio.
- `Other (Outro)`: sistemas operacionais identificados pelo CloudFront, mas que não estão entre os mais populares. Se `Bot/Crawler`, `Empty` e/ou `Unknown` não for exibido entre os nove primeiros valores, eles também serão incluídos em `Other`.
- `Unknown`: solicitações para as quais o sistema operacional não é especificado no cabeçalho `User-Agent HTTP`.

Relatório de locais

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou ALL se o relatório tiver sido gerado para todas as distribuições.

FriendlyName

Um nome de domínio alternativo (CNAME), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

LocationCode

A abreviação do local do qual o CloudFront recebeu solicitações. Para obter mais informações sobre os possíveis valores, veja a descrição de local em [Como os dados do relatório de locais estão relacionados aos dados dos logs padrão \(logs de acesso\) do CloudFront](#).

LocationName

O nome do local do qual o CloudFront recebeu solicitações.

Solicitações

O número de solicitações recebidas pelo CloudFront em cada local.

RequestsPct

O número de solicitações recebidas pelo CloudFront de cada local como porcentagem do número total de solicitações recebidas pelo CloudFront de todos os locais durante o período especificado.

TotalBytes

O número de bytes fornecido pelo CloudFront aos visualizadores nesse país ou estado para a distribuição e o período especificados.

Relatório de tendências do local

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual o relatório foi gerado ou ALL se o relatório tiver sido gerado para todas as distribuições.

FriendlyName

Um nome de domínio alternativo (CNAME), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

TimeBucket

A hora ou o dia ao qual os dados se aplicam, no Tempo Universal Coordenado (UTC).

(Locais)

As demais colunas do relatório indicam os locais dos quais o CloudFront recebeu solicitações. Para obter mais informações sobre os possíveis valores, veja a descrição de local em [Como os dados do relatório de locais estão relacionados aos dados dos logs padrão \(logs de acesso\) do CloudFront](#).

Como os dados do relatório de locais estão relacionados aos dados dos logs padrão (logs de acesso) do CloudFront

A lista a seguir mostra como os dados do relatório de locais do console do CloudFront correspondem aos valores dos logs de acesso do CloudFront. Para mais informações sobre os logs de acesso do CloudFront, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).

Local

O país ou estado dos EUA no qual o visualizador está localizado. Nos logs de acesso, a coluna `c-ip` contém o endereço IP do dispositivo de origem do visualizador. Usamos dados de geolocalização para identificar a localização geográfica do dispositivo com base no endereço IP.

Se você estiver exibindo o relatório Localizações por país, a lista de países é baseada no [ISO 3166-2, Códigos para a representação dos nomes dos países e das suas subdivisões – Parte 2: Código de subdivisão do país](#). A lista de países inclui os seguintes valores adicionais:

- Anonymous Proxy (Proxy anônimo): a solicitação de origem de um proxy anônimo.
- Satellite Provider (Provedor de satélite): a solicitação de origem de um provedor de satélite que fornece serviços de Internet para vários países. Os visualizadores podem estar em países com alto risco de fraude.
- Europe (Unknown) (Europa (desconhecido)): a solicitação de origem de um IP em um bloco usado por vários países europeus. O país de origem da solicitação não pode ser determinado. O CloudFront usa Europe (Unknown) (Europa (desconhecido)) como padrão.
- Asia/Pacific (Unknown) (Ásia/Pacífico (desconhecido)): a solicitação de origem de um IP em um bloco usado por vários países na região Ásia/Pacífico. O país de origem da solicitação não pode ser determinado. O CloudFront usa Asia/Pacific (Unknown) (Ásia/Pacífico (desconhecido)) como padrão.

Se você estiver exibindo o relatório Locations por estado dos EUA, observe que ele pode incluir os territórios e as regiões das Forças Armadas dos EUA.

Note

Se o CloudFront não puder determinar a localização de um usuário, o local aparecerá como "Unknown" nos relatórios do visualizador.

Request Count

O número total de solicitações do país ou estado dos EUA no qual o visualizador está localizado, para a distribuição e o período especificados. Esse valor normalmente corresponde ao número de solicitações GET dos endereços IP desse país ou estado nos logs de acesso do CloudFront.

Solicitação %

Um dos seguintes, dependendo do valor selecionado para Details:

- Countries (Países): as solicitações desse país como porcentagem do número total de solicitações.
- U.S. States (Estados dos EUA): as solicitações desse estado como porcentagem do número total de solicitações dos Estados Unidos.

Se as solicitações forem provenientes de mais de 50 países, não será possível calcular a % de solicitações com base nos dados dessa tabela, porque a coluna Request Count não inclui todas as solicitações durante o período especificado.

Bytes

O número de bytes fornecido pelo CloudFront aos visualizadores nesse país ou estado para a distribuição e o período especificados. Para alterar a exibição dos dados dessa coluna para KB, MB ou GB, selecione o link no cabeçalho da coluna.

Monitorar métricas do CloudFront com o Amazon CloudWatch

O Amazon CloudFront é integrado ao Amazon CloudWatch e publica automaticamente métricas operacionais para distribuições e funções de borda (do [Lambda@Edge e CloudFront Functions](#)). É possível usar essas métricas para solucionar problemas, monitorar e depurar problemas. Muitas dessas métricas são exibidas em um conjunto de grafos no console do CloudFront e também podem ser acessadas usando a CLI ou a API do CloudFront. Todas essas métricas estão disponíveis no [console do CloudWatch](#) ou por meio da CLI ou da API do CloudWatch. As métricas do CloudFront

não são contabilizadas em relação às [cotas do CloudWatch \(anteriormente conhecidas como limites\)](#) e não geram custo adicional.

Além das métricas padrão para distribuições do CloudFront, é possível ativar outras métricas por um custo adicional. As métricas adicionais se aplicam às distribuições do CloudFront e devem ser ativadas para cada distribuição separadamente. Para obter mais informações sobre o custo, consulte [the section called “Estimar o custo para as métricas adicionais do CloudFront”](#).

Também é possível definir alarmes com base nessas métricas no console do CloudFront ou na API, na CLI ou no console do CloudWatch. Por exemplo, é possível definir um alarme com base na métrica `5xxErrorRate`, que representa a porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta está no intervalo de 500 a 599. Quando a taxa de erro atinge determinado valor por certo período, por exemplo, 5% das solicitações por cinco minutos contínuos, o alarme é acionado. Especifique o valor do alarme e a respectiva unidade de tempo ao criar o alarme.

Observações

- Ao criar um alarme do CloudWatch no console do CloudFront, ele cria um para você na região Leste dos EUA (Norte da Virgínia) (`us-east-1`). Se você criar um alarme no console do CloudWatch, deverá usar a mesma região. Como o CloudFront é um serviço global, as métricas do serviço são enviadas para o Leste dos EUA (Norte da Virgínia).
- Ao criar alarmes, aplica-se o [preço padrão do CloudWatch](#).

Tópicos

- [Visualizar métricas do CloudFront e de funções de borda](#)
- [Criar alarmes do para métricas do](#)
- [Baixar dados de métricas no formato CSV](#)
- [Obter métricas usando a API do Amazon CloudWatch](#)

Visualizar métricas do CloudFront e de funções de borda

É possível visualizar métricas operacionais sobre as distribuições do CloudFront e as [funções de borda](#) no console do CloudFront.

Como visualizar métricas do CloudFront e de funções de borda no CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Monitoring (Monitoramento).
3. Para visualizar grafos sobre a atividade de uma distribuição do CloudFront ou função de borda específica, selecione uma e, depois, escolha View distribution metrics (Visualizar métricas de distribuição) ou View metrics (Visualizar métricas).
4. É possível personalizar os gráficos fazendo o seguinte:
 - a. Para alterar o intervalo de tempo das informações exibidas nos gráficos, escolha 1h (uma hora), 3h (três horas) ou outro intervalo, ou especifique um intervalo personalizado.
 - b. Para alterar a frequência com que o CloudFront atualiza as informações no gráfico, clique na seta para baixo ao lado do ícone de atualização e escolha uma taxa de atualização. A taxa de atualização padrão é de um minuto, mas é possível escolher outras opções.
5. Para exibir gráficos do CloudFront no console do CloudWatch, escolha Add to dashboard (Adicionar ao painel). É necessário usar a região Leste dos EUA (N. da Virgínia) para visualizar os grafos no console do CloudWatch.

Tópicos

- [Métricas de distribuição padrão do CloudFront](#)
- [Ativar métricas adicionais de distribuição do CloudFront](#)
- [Métricas de função padrão do Lambda@Edge](#)
- [Métricas padrão do CloudFront Functions](#)

Métricas de distribuição padrão do CloudFront

As seguintes métricas padrão são incluídas para todas as distribuições do CloudFront, sem custo adicional:

Solicitações

O número total de solicitações de visualizador recebidas pelo CloudFront, para todos os métodos HTTP e para solicitações HTTP e HTTPS.

Bytes baixados

O número de bytes obtidos por download por visualizadores para solicitações GET, HEAD e OPTIONS.

Bytes carregados

O número total de bytes dos quais os visualizadores fizeram upload para o CloudFront usando as solicitações POST e PUT.

Taxa de erros 4xx

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 4xx.

Taxa de erros 5xx

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 5xx.

Taxa de erros total

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 4xx ou 5xx.

Essas métricas são mostradas em grafos para cada distribuição do CloudFront na página Monitoramento do console do CloudFront. Em cada gráfico, os totais são exibidos na granularidade de 1 minuto. Além de visualizar os gráficos, também é possível [fazer download de relatórios de métricas como arquivos CSV](#).

Ativar métricas adicionais de distribuição do CloudFront

Além das métricas padrão, é possível ativar outras métricas por um custo adicional. Para obter mais informações sobre o custo, consulte [the section called “Estimar o custo para as métricas adicionais do CloudFront”](#).

Essas métricas adicionais devem ser ativadas para cada distribuição separadamente:

Taxa de acertos do cache

A porcentagem de todas as solicitações armazenáveis em cache para as quais o CloudFront forneceu o conteúdo do cache. Solicitações HTTP POST e PUT e erros não são considerados solicitações armazenáveis em cache.

Latência de origem

O tempo total gasto de quando o CloudFront recebe uma solicitação até quando começa a fornecer uma resposta à rede (não ao visualizador), para solicitações que são atendidas da origem, não do cache do CloudFront. Isso também é conhecido como latência de primeiro byte ou tempo até o primeiro byte.

Taxa de erro por código de status

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é um código específico no intervalo 4xx ou 5xx. Essa métrica está disponível para todos os seguintes códigos de erro: 401, 403, 404, 502, 503 e 504.

Você pode ativar métricas adicionais no console do CloudFront com o AWS CloudFormation, com a AWS Command Line Interface (AWS CLI) ou com a API do CloudFront.

Console

Como ativar métricas adicionais

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Monitoring (Monitoramento).
3. Selecione a distribuição para a qual ativar métricas adicionais e selecione View distribution metrics (Visualizar métricas de distribuição).
4. Selecione Manage additional metrics (Gerenciar métricas adicionais).
5. Na janela Manage additional metrics (Gerenciar métricas adicionais), ative Enabled (Habilitado). Após a ativação das métricas adicionais, é possível fechar a janela Manage additional metrics (Gerenciar métricas adicionais).

Após a ativação das métricas adicionais, elas são mostradas em gráficos. Em cada gráfico, os totais são exibidos na granularidade de 1 minuto. Além de visualizar os gráficos, também é possível [fazer download de relatórios de métricas como arquivos CSV](#).

CloudFormation

Para ativar métricas adicionais com o CloudFormation, use o tipo de recurso `AWS::CloudFront::MonitoringSubscription`. O exemplo a seguir mostra a sintaxe do modelo do AWS CloudFormation no formato YAML, para habilitar métricas adicionais.

```
Type: AWS::CloudFront::MonitoringSubscription
Properties:
  DistributionId: EDFDVBD6EXAMPLE
  MonitoringSubscription:
    RealtimeMetricsSubscriptionConfig:
      RealtimeMetricsSubscriptionStatus: Enabled
```

CLI

Para gerenciar métricas adicionais com a AWS Command Line Interface (AWS CLI), use um dos seguintes comandos:

Como ativar métricas adicionais para uma distribuição

- Use o comando `create-monitoring-subscription` como no exemplo a seguir. Substitua **EDFDVBD6EXAMPLE** pelo ID da distribuição para a qual você está habilitando métricas adicionais.

```
aws cloudfront create-monitoring-subscription --
distribution-id EDFDVBD6EXAMPLE --monitoring-subscription
RealtimeMetricsSubscriptionConfig={RealtimeMetricsSubscriptionStatus=Enabled}
```

Como verificar se as métricas adicionais estão ativadas para uma distribuição

- Use o comando `get-monitoring-subscription` como no exemplo a seguir. Substitua **EDFDVBD6EXAMPLE** pelo ID da distribuição que você está verificando.

```
aws cloudfront get-monitoring-subscription --distribution-id EDFDVBD6EXAMPLE
```

Como desativar métricas adicionais para uma distribuição

- Use o comando `delete-monitoring-subscription` como no exemplo a seguir. Substitua **EDFDVBD6EXAMPLE** pelo ID da distribuição para a qual você está desativando métricas adicionais.

```
aws cloudfront delete-monitoring-subscription --distribution-id EDFDVBD6EXAMPLE
```

API

Para gerenciar métricas adicionais com a API do CloudFront, use uma das seguintes operações de API.

- Para ativar métricas adicionais para uma distribuição, use [CreateMonitoringSubscription](#).
- Para ver se as métricas adicionais estão ativadas para uma distribuição, use [GetMonitoringSubscription](#).
- Para desativar métricas adicionais para uma distribuição, use [DeleteMonitoringSubscription](#).

Consulte mais informações sobre essas operações de API na documentação de referência da API do AWS SDK ou de outro cliente de API.

Estimar o custo para as métricas adicionais do CloudFront

Quando você ativa métricas adicionais para uma distribuição, o CloudFront envia até oito métricas para o CloudWatch na região Leste dos EUA (N. da Virgínia). O CloudWatch cobra uma taxa baixa e fixa para cada métrica. Essa taxa é cobrada apenas uma vez por mês, por métrica (até 8 métricas por distribuição). É uma taxa fixa, portanto, o custo permanece o mesmo, independentemente do número de solicitações ou respostas que a distribuição do CloudFront recebe ou envia. Para obter a taxa por métrica, consulte a [página de definição de preço do Amazon CloudWatch](#) e a [Calculadora de definição de preço do CloudWatch](#). Taxas de API adicionais se aplicam ao recuperar as métricas com a API do CloudWatch.

Métricas de função padrão do Lambda@Edge

É possível usar métricas do CloudWatch para monitorar, em tempo real, problemas nas funções do Lambda@Edge. Não há cobrança adicional por essas métricas.

Quando você associa uma função do Lambda@Edge a um comportamento de cache em uma distribuição do CloudFront, o Lambda começa a enviar métricas para o CloudWatch automaticamente. As métricas estão disponíveis para todas as regiões do Lambda, mas, para visualizar métricas no console do CloudWatch ou obter os dados de métricas de API do CloudWatch, use a região Leste dos EUA (N. da Virgínia) (us-east-1). O nome do grupo de métricas é formatado como: AWS/CloudFront/*distribution-ID*, em que *distribution-ID* é o ID da

distribuição do CloudFront à qual a função do Lambda@Edge está associada. Para obter mais informações sobre as métricas do CloudWatch, consulte o [Guia do usuário do Amazon CloudWatch](#).

As seguintes métricas padrão são exibidas em grafos para cada função do Lambda@Edge na página Monitoramento do console do CloudFront:

- 5xxTaxa de erros do Lambda@Edge
- Erros de execução do Lambda
- Respostas inválidas do Lambda
- Aceleradores do Lambda

Os gráficos incluem o número de invocações, erros, limitações e muito mais. Em cada gráfico, os totais são exibidos na granularidade de 1 minuto, agrupados por região da AWS.

Se você vir um pico nos erros que deseja investigar, será possível escolher uma função e visualizar os arquivos de log por região da AWS, até determinar qual função está causando problemas e em qual região da AWS. Para obter mais informações sobre a resolução de erros do Lambda@Edge, consulte:

- [the section called “Como determinar o tipo de falha”](#)
- [Quatro etapas para depurar sua entrega de conteúdo na AWS](#)

Métricas padrão do CloudFront Functions

O CloudFront Functions envia métricas operacionais para o Amazon CloudWatch para que você possa monitorar suas funções. A visualização dessas métricas pode ajudar a solucionar problemas, rastrear e depurar problemas. O CloudFront Functions publica as seguintes métricas no CloudWatch:

- **Invocations (Invocações) (FunctionInvocations):** o número de vezes que a função foi iniciada (invocada) em um determinado período de tempo.
- **Validation errors (Erros de validação) (FunctionValidationErrors):** o número de erros de validação produzidos pela função em um determinado período de tempo. Os erros de validação ocorrem quando a função é executada com êxito, mas retorna dados inválidos (um [objeto de evento](#) inválido).
- **Execution errors (Erros de execução) (FunctionExecutionErrors):** o número de erros de execução que ocorreram em um determinado período de tempo. Erros de execução ocorrem quando a função falha ao concluir com êxito.

- **Compute utilization (Utilização de computação) `FunctionComputeUtilization`:** a quantidade de tempo que a função levou para ser executada como um percentual do tempo máximo permitido. Por exemplo, um valor de 35 significa que a função foi concluída em 35% do tempo máximo permitido. Esta métrica é um número entre 0 e 100.

Se esse valor atingir ou estiver próximo de 100, a função usou ou está perto de usar o tempo de execução permitido e as solicitações subseqüentes poderão ser submetidas a controle de utilização. Se sua função estiver sendo executada com 80% ou mais de utilização, recomendamos que você revise sua função para reduzir o tempo de execução e melhorar a utilização. Por exemplo, talvez você queira registrar somente erros em log, simplificar qualquer expressão regex complexa ou remover a análise desnecessária de objetos JSON complexos.

- **Limitações(`FunctionThrottles`):** o número de vezes que a função foi limitada em determinado período. As funções podem ser limitadas pelos seguintes motivos:
 - A função excede continuamente o tempo máximo permitido para execução.
 - A função causa erros de compilação.
 - Há um número excepcionalmente alto de solicitações por segundo.

O `KeyValueStore` do CloudFront também envia as seguintes métricas operacionais para o Amazon CloudWatch:

- **Read requests (Solicitações de leitura) (`KvsReadRequests`):** o número de vezes que a função foi lida com êxito do armazenamento de chave-valor em determinado período.
- **Read errors (Erros de leitura) (`KvsReadErrors`):** o número de vezes que a função falhou ao ler o armazenamento de chave-valor em determinado período.

Todas essas métricas são publicadas no CloudWatch na região Leste dos EUA (Norte da Virgínia) (`us-east-1`), no namespace do CloudFront. Você também pode visualizar essas métricas no console do CloudWatch. No console do CloudWatch, você pode visualizar as métricas por função ou por função por distribuição.

Você também pode usar o CloudWatch para definir alarmes com base nessas métricas. Por exemplo, você pode definir um alarme com base na métrica de tempo de execução (`FunctionComputeUtilization`), que representa a porcentagem do tempo disponível que a função levou para executar. Quando o tempo de execução atinge determinado valor por certo período. Por exemplo, se você escolher mais de 70% do tempo disponível por 15 minutos contínuos,

o alarme será acionado. Especifique o valor do alarme e a respectiva unidade de tempo ao criar o alarme.

Note

O CloudFront Functions envia métricas ao CloudWatch apenas para funções na fase LIVE que são executadas em resposta a solicitações e respostas de produção. Quando você [testa uma função](#), o CloudFront não envia métricas para o CloudWatch. A saída de teste contém informações sobre erros, utilização de computação e logs de funções (instruções `console.log()`), mas essas informações não são enviadas para o CloudWatch.

Para obter informações sobre como obter essas métricas com a API do CloudWatch, consulte [the section called “Obter métricas usando a API do CloudWatch”](#).

Criar alarmes do para métricas do

No console do CloudFront, é possível definir alarmes para notificá-lo pelo Amazon Simple Notification Service (Amazon SNS) com base em métricas específicas do CloudFront.

Como criar alarmes para métricas

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, selecione Alarmes.
3. Selecione Create alarm (Criar alarme).
4. Em Detalhes, especifique o seguinte:
 - a. Nome do alarme: um nome para o alarme.
 - b. Distribuição: a distribuição do CloudFront para a qual você está criando o alarme.
5. Em Condição, especifique o seguinte:
 - a. Métrica: a métrica para a qual você está criando o alarme.
 - b. “SE” <condition>: o limite de quando o CloudWatch deve acionar um alarme e enviar uma notificação para o tópico do Amazon SNS. Por exemplo, para receber uma notificação quando a taxa de erro 5xx exceder 1%, especifique o seguinte:

Taxa de erros 5xx > 1

- c. “POR” períodos consecutivos: o período em que a condição deve ser atendida antes de acionar um alarme. Ao escolher um valor, vise um equilíbrio adequado entre um valor que não acione o alarme para problemas temporários, mas que acione o alarme para problemas persistentes ou reais.
 - d. (Opcional) Notificar: o tópico do Amazon SNS para o qual enviar a notificação se essa métrica acionar um alarme.
6. Selecione Criar alarme.

Observações

- Ao inserir os valores da condição, use números inteiros sem pontuação. Por exemplo, para especificar mil, insira **1000**.
- Para as taxas de erro 4xx, 5xx e total, o valor especificado é uma porcentagem.
- Para solicitações, bytes obtidos por download e bytes enviados por upload, o valor especificado é em unidades. Por exemplo, 1073742000 bytes.

Consulte mais informações sobre como criar tópicos do Amazon SNS em [Creating an Amazon SNS topic](#) no Guia do desenvolvedor do Amazon Simple Notification Service.

Baixar dados de métricas no formato CSV

É possível fazer download dos dados de métricas do CloudWatch para uma distribuição do CloudFront no formato CSV.

Como baixar dados de métricas no formato CSV

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. No painel de navegação, escolha Monitoring (Monitoramento).
3. Selecione a distribuição e escolha Visualizar métricas de distribuição.
4. Selecione Baixar CSV e escolha o período (por exemplo, Para o último dia (período de 1 hora)).
5. Depois que o arquivo for baixado, abra-o para visualizar as informações a seguir.

Tópicos

- [Informações sobre o relatório](#)
- [Dados no relatório de métricas](#)

Informações sobre o relatório

As primeiras linhas do relatório incluem as seguintes informações:

Versão

A versão dos relatórios do CloudFront.

Relatório

O nome do relatório.

DistributionID

O ID da distribuição para a qual você executou o relatório.

StartDateUTC

O início do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

EndDateUTC

O término do intervalo de datas para o qual você executou o relatório, no Tempo Universal Coordenado (UTC).

GeneratedTimeUTC

A data e a hora nas quais você executou o relatório, no Tempo Universal Coordenado (UTC).

Granularity

O período de cada linha do relatório, por exemplo, ONE_MINUTE.

Dados no relatório de métricas

O relatório inclui os seguintes valores:

DistributionID

O ID da distribuição para a qual você executou o relatório.

FriendlyName

Um nome de domínio alternativo (CNAME), se houver, para a distribuição. Se uma distribuição não tiver nomes de domínio alternativos, a lista incluirá um nome de domínio de origem dela.

TimeBucket

A hora ou o dia ao qual os dados se aplicam, no Tempo Universal Coordenado (UTC).

Solicitações

O número total de solicitações de todos os códigos de status do HTTP (por exemplo, 200 ou 404) e de todos os métodos (por exemplo, GET, HEAD ou POST) durante o período.

BytesDownloaded

O número de bytes baixados pelos visualizadores para a distribuição especificada durante o período.

BytesUploaded

O número de bytes que visualizadores enviaram por upload para a distribuição especificada durante o período.

TotalErrorRatePct

A porcentagem de solicitações para as quais o código de status HTTP foi um erro 4xx ou 5xx para a distribuição especificada durante o período.

4xxErrorRatePct

A porcentagem de solicitações para as quais o código de status HTTP foi um erro 4xx para a distribuição especificada durante o período.

5xxErrorRatePct

A porcentagem de solicitações para as quais o código de status HTTP foi um erro 5xx para a distribuição especificada durante o período.

Se você tiver [ativado métricas adicionais](#) para a distribuição, o relatório também incluirá os seguintes valores adicionais:

401ErrorRatePct

A porcentagem de solicitações para as quais o código de status HTTP foi um erro 401 para a distribuição especificada durante o período.

403ErrorRatePct

A porcentagem de solicitações para as quais o código de status HTTP foi um erro 403 para a distribuição especificada durante o período.

404ErrorRatePct

A porcentagem de solicitações para as quais o código de status HTTP foi um erro 404 para a distribuição especificada durante o período.

502ErrorRatePct

A porcentagem de solicitações para as quais o código de status HTTP foi um erro 502 para a distribuição especificada durante o período.

503ErrorRatePct

A porcentagem de solicitações para as quais o código de status HTTP foi um erro 503 para a distribuição especificada durante o período.

504ErrorRatePct

A porcentagem de solicitações para as quais o código de status HTTP foi um erro 504 para a distribuição especificada durante o período.

OriginLatency

O tempo total gasto, em milissegundos, desde o momento em que o CloudFront recebeu uma solicitação até o instante em que começou a fornecer uma resposta à rede (não ao visualizador), para solicitações que foram fornecidas da origem, não do cache do CloudFront. Isso também é conhecido como latência de primeiro byte ou tempo até o primeiro byte.

CacheHitRate

A porcentagem de todas as solicitações armazenáveis em cache para as quais o CloudFront forneceu o conteúdo do cache. Solicitações HTTP POST e PUT e erros não são considerados solicitações armazenáveis em cache.

Obter métricas usando a API do Amazon CloudWatch

É possível usar a API ou a AWS Command Line Interface (AWS CLI) do CloudWatch para obter as métricas do CloudFront em programas ou aplicações que você cria. É possível usar os

dados brutos para criar seus próprios painéis personalizados, suas próprias ferramentas de alarmes e muito mais.

Consulte mais informações em [get-metric-data](#) na Referência de comandos da AWS CLI ou na operação de API [GetMetricData](#) na Referência de API do Amazon CloudWatch.

Tópicos

- [Valores para todas as métricas do CloudFront](#)
- [Valores para métricas de distribuição do CloudFront](#)
- [Valores para métricas de função do CloudFront](#)

Note

Para obter as métricas do CloudFront da API do CloudWatch, use a região Leste dos EUA (Norte da Virgínia) (us-east-1). Você também precisa conhecer alguns valores e tipos para cada métrica.

Valores para todas as métricas do CloudFront

Os valores a seguir se aplicam a todas as métricas do CloudFront:

Namespace

O valor para Namespace é sempre `AWS/CloudFront`.

Dimensões

Cada métrica do CloudFront tem as duas dimensões a seguir:

DistributionId

O ID da distribuição do CloudFront para o qual você deseja obter métricas.

FunctionName

O nome da função (no CloudFront Functions) para a qual você deseja obter métricas.

Essa dimensão se aplica apenas a funções.

Region

O valor de Region é sempre `Global`, pois o CloudFront é um serviço global.

Valores para métricas de distribuição do CloudFront

Use as informações da lista a seguir para obter detalhes sobre métricas específicas de distribuição do CloudFront da API do CloudWatch. Algumas dessas métricas ficam disponíveis somente quando você ativa métricas adicionais para a distribuição.

Note

Apenas uma estatística, `Average` ou `Sum`, é aplicável para cada métrica. A lista a seguir especifica qual estatística é aplicável a essa métrica.

Taxa de erros 4xx

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 4xx.

- Nome da métrica: `4xxErrorRate`
- Estatística válida: `Average`
- Unidade: `Percent`

Taxa de erro 401

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 401. Para obter essa métrica, primeiro é necessário [ativar métricas adicionais](#).

- Nome da métrica: `401ErrorRate`
- Estatística válida: `Average`
- Unidade: `Percent`

Taxa de erro 403

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 403. Para obter essa métrica, primeiro é necessário [ativar métricas adicionais](#).

- Nome da métrica: `403ErrorRate`
- Estatística válida: `Average`
- Unidade: `Percent`

Taxa de erro 404

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 404. Para obter essa métrica, primeiro é necessário [ativar métricas adicionais](#).

- Nome da métrica: `404ErrorRate`
- Estatística válida: `Average`
- Unidade: `Percent`

Taxa de erros 5xx

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 5xx.

- Nome da métrica: `5xxErrorRate`
- Estatística válida: `Average`
- Unidade: `Percent`

Taxa de erro 502

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 502. Para obter essa métrica, primeiro é necessário [ativar métricas adicionais](#).

- Nome da métrica: `502ErrorRate`
- Estatística válida: `Average`
- Unidade: `Percent`

Taxa de erro 503

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 503. Para obter essa métrica, primeiro é necessário [ativar métricas adicionais](#).

- Nome da métrica: `503ErrorRate`
- Estatística válida: `Average`
- Unidade: `Percent`

Taxa de erro 504

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 504. Para obter essa métrica, primeiro é necessário [ativar métricas adicionais](#).

- Nome da métrica: `504ErrorRate`
- Estatística válida: `Average`
- Unidade: `Percent`

Bytes baixados

O número de bytes obtidos por download por visualizadores para solicitações GET, HEAD e OPTIONS.

- Nome da métrica: BytesDownloaded
- Estatística válida: Sum
- Unidade: None

Bytes carregados

O número total de bytes que os visualizadores fizeram upload para a origem com o CloudFront usando POST e PUT.

- Nome da métrica: BytesUploaded
- Estatística válida: Sum
- Unidade: None

Taxa de acertos do cache

A porcentagem de todas as solicitações armazenáveis em cache para as quais o CloudFront forneceu o conteúdo do cache. Solicitações HTTP POST e PUT e erros não são considerados solicitações armazenáveis em cache. Para obter essa métrica, primeiro é necessário [ativar métricas adicionais](#).

- Nome da métrica: CacheHitRate
- Estatística válida: Average
- Unidade: Percent

Latência de origem

O tempo total gasto, em milissegundos, de quando o CloudFront recebe uma solicitação até quando começa a fornecer uma resposta à rede (não ao visualizador), para solicitações que são fornecidas pela origem, não pelo cache do CloudFront. Isso também é conhecido como latência de primeiro byte ou tempo até o primeiro byte. Para obter essa métrica, primeiro é necessário [ativar métricas adicionais](#).

- Nome da métrica: OriginLatency
- Estatística válida: Percentile
- Unidade: Milliseconds

Note

Para obter uma estatística Percentile da API do CloudWatch, use o parâmetro `ExtendedStatistics`, não `Statistics`. Para obter mais informações, consulte

[GetMetricStatistics](#) na Referência da API do Amazon CloudWatch ou a documentação de referência dos [AWS SDKs](#).

Solicitações

O número total de solicitações de visualizador recebidas pelo CloudFront, para todos os métodos HTTP e para solicitações HTTP e HTTPS.

- Nome da métrica: `Requests`
- Estatística válida: `Sum`
- Unidade: `None`

Taxa de erros total

A porcentagem de todas as solicitações do visualizador para as quais o código de status HTTP da resposta é 4xx ou 5xx.

- Nome da métrica: `TotalErrorRate`
- Estatística válida: `Average`
- Unidade: `Percent`

Valores para métricas de função do CloudFront

Use as informações da lista a seguir para obter detalhes sobre métricas específicas de função do CloudFront da API do CloudWatch.

Note

Apenas uma estatística, `Average` ou `Sum`, é aplicável para cada métrica. A lista a seguir especifica qual estatística é aplicável a essa métrica.

Invocações

O número de vezes que a função foi iniciada (invocada) em um determinado período de tempo.

- Nome da métrica: `FunctionInvocations`
- Estatística válida: `Sum`
- Unidade: `None`

Erros de validação

O número de erros de validação produzidos pela função em um determinado período de tempo. Os erros de validação ocorrem quando a função é executada com êxito, mas retorna dados inválidos (um objeto de evento inválido).

- Nome da métrica: `FunctionValidationErrors`
- Estatística válida: `Sum`
- Unidade: `None`

Erros de execução

O número de erros de execução que ocorreram em um determinado período de tempo. Erros de execução ocorrem quando a função falha ao concluir com êxito.

- Nome da métrica: `FunctionExecutionErrors`
- Estatística válida: `Sum`
- Unidade: `None`

Utilização de recursos de computação

A quantidade de tempo (0 a 100) que a função levou para ser executada como uma porcentagem do tempo máximo permitido. Por exemplo, um valor de 35 significa que a função foi concluída em 35% do tempo máximo permitido.

- Nome da métrica: `FunctionComputeUtilization`
- Estatística válida: `Average`
- Unidade: `Percent`

Controles de utilização

O número de vezes que a função foi limitada em determinado período.

- Nome da métrica: `FunctionThrottles`
- Estatística válida: `Sum`
- Unidade: `None`

Registro em log do CloudFront e de funções de borda

O Amazon CloudFront fornece diferentes tipos de registro em log. É possível registrar em log as solicitações do visualizador recebidas pelas distribuições do CloudFront ou registrar as atividades do

serviço do CloudFront (atividade de API) em sua conta da AWS. Também é possível obter logs das funções do CloudFront Functions e do Lambda@Edge.

Registrar solicitações em log

O CloudFront fornece as seguintes maneiras de registrar em log as solicitações recebidas por suas distribuições.

Registros padrão (logs de acesso)

Os logs padrão do CloudFront fornecem registros detalhados sobre cada solicitação feita para uma distribuição. Esses logs são úteis para muitos cenários, incluindo auditorias de segurança e acesso.

Os logs padrão do CloudFront são entregues ao bucket do Amazon S3 de sua escolha. O CloudFront não cobra pelos logs padrão, embora você gere cobranças do Amazon S3 por armazenar e acessar os arquivos de log.

Para ter mais informações, consulte [Usar logs padrão \(logs de acesso\)](#).

Logs em tempo real

Os logs em tempo real do CloudFront fornecem informações sobre solicitações feitas para uma distribuição, em tempo real (os registros de log são entregues em segundos após o recebimento das solicitações). Você pode escolher a taxa de amostragem para seus logs em tempo real, ou seja, a porcentagem de solicitações para as quais deseja receber registros de log em tempo real. Também é possível escolher os campos específicos que deseja receber nos registros de log.

Os logs em tempo real do CloudFront são entregues ao stream de dados de sua escolha no Amazon Kinesis Data Streams. O CloudFront cobra por logs em tempo real, além das cobranças que você incorre pelo uso do Kinesis Data Streams.

Para ter mais informações, consulte [Usar logs em tempo real](#).

Registrar em log funções de borda

É possível usar o Amazon CloudWatch Logs para obter logs para as funções de borda, tanto do Lambda@Edge quanto do CloudFront Functions. É possível acessar os logs usando o console do CloudWatch ou a API do CloudWatch Logs. Para ter mais informações, consulte [the section called “Logs de funções de borda”](#).

Registrar em log as atividades do serviço

Você pode usar o AWS CloudTrail para registrar a atividade do serviço do CloudFront (atividade de API) em sua conta da AWS. O CloudTrail fornece um registro de ações de API realizadas por um usuário, uma função ou um serviço da AWS no CloudFront. Usando as informações coletadas pelo CloudTrail, é possível determinar a solicitação de API realizada para o CloudFront, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita, além de detalhes adicionais.

Para ter mais informações, consulte [Registrar em log chamadas de API do Amazon CloudFront usando o AWS CloudTrail](#).

Consulte mais informações sobre o registro em log nos seguintes tópicos:

Tópicos

- [Configurar e usar logs padrão \(logs de acesso\)](#)
- [Usar logs em tempo real](#)
- [Logs de funções de borda](#)
- [Registrar em log chamadas de API do Amazon CloudFront usando o AWS CloudTrail](#)

Configurar e usar logs padrão (logs de acesso)

Você pode configurar o CloudFront para criar arquivos de log que contenham informações detalhadas sobre todas as solicitações dos usuários que ele recebe. Esses são chamados de logs padrão, também conhecidos como logs de acesso. Se você habilitar os logs padrão, também será possível especificar o bucket do Amazon S3 no qual você deseja que o CloudFront salve arquivos.

É possível habilitar os logs padrão ao criar ou atualizar uma distribuição. Para ter mais informações, consulte [Registro em log](#).

O CloudFront também oferece logs em tempo real, que fornecem informações sobre solicitações feitas em uma distribuição em tempo real (os logs são entregues em segundos após o recebimento das solicitações). É possível usar os logs em tempo real para monitorar, analisar e tomar ações com base na performance da entrega de conteúdo. Para obter mais informações, consulte [Usar logs em tempo real](#).

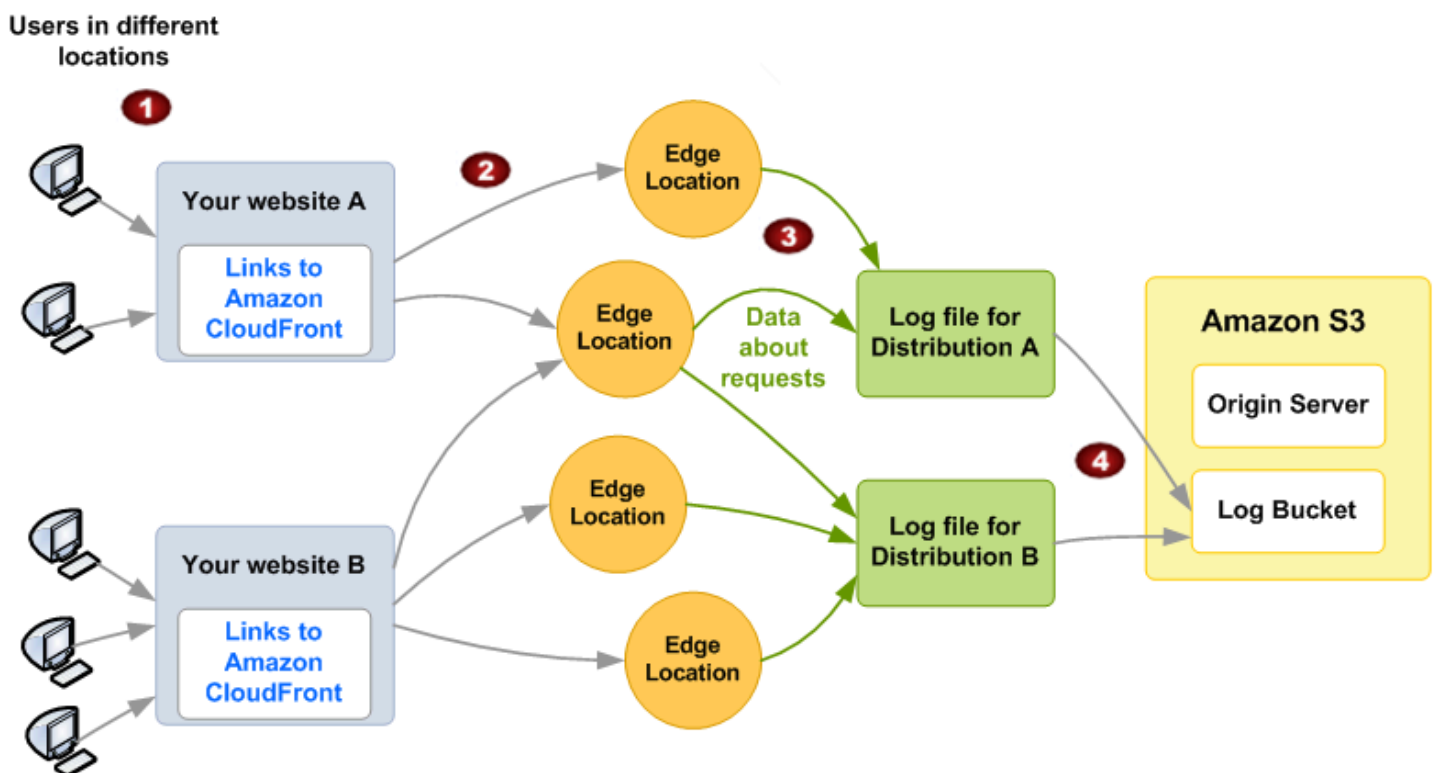
Tópicos

- [Como funciona o registro em log padrão](#)

- [Escolher um bucket do Amazon S3 para logs padrão](#)
- [Permissões necessárias para configurar o registro em log padrão e acessar os arquivos de log](#)
- [Política de chaves obrigatória para buckets do SSE-KMS](#)
- [Formato do nome do arquivo](#)
- [Tempo de entrega do arquivo de log padrão](#)
- [Como as solicitações são registradas em log quando os cabeçalhos ou o URL de solicitação excedem o tamanho máximo](#)
- [Analisar logs padrão](#)
- [Editar as configurações do registro em log padrão](#)
- [Excluir arquivos de log padrão de um bucket do Amazon S3](#)
- [Formato de arquivo de log padrão](#)
- [Cobranças de logs padrão](#)

Como funciona o registro em log padrão

O diagrama a seguir mostra como o CloudFront registra informações sobre as solicitações de seus objetos.



As considerações a seguir explicam como o CloudFront registra informações sobre as solicitações de seus objetos, conforme ilustrado no diagrama anterior.

1. Neste diagrama, você tem dois sites, A e B, e duas distribuições do CloudFront correspondentes. Os usuários solicitam seus objetos usando URLs associados a suas distribuições.
2. O CloudFront encaminha cada solicitação para o ponto de presença adequado.
3. O CloudFront grava dados sobre cada solicitação em um arquivo de log específico a essa distribuição. Neste exemplo, informações sobre as solicitações relacionadas à Distribuição A são registradas em um arquivo de log exclusivo para ela. Da mesma forma, informações sobre as solicitações relacionadas à Distribuição B são registradas em um arquivo de log exclusivo para ela.
4. O CloudFront periodicamente salva o arquivo de log de uma distribuição no bucket do Amazon S3 especificado ao habilitar o registro. Depois, o CloudFront começa salvando as informações de solicitações subsequentes em um novo arquivo de log para a distribuição.

Se nenhum usuário acessar seu conteúdo em uma hora específica, você não receberá arquivos de log referentes a essa hora.

Cada entrada em um arquivo de log fornece detalhes sobre uma única solicitação. Para obter mais informações sobre o formato do arquivo de log, consulte [Formato de arquivo de log padrão](#).

Note

Recomendamos que você use os logs para compreender a natureza das solicitações do seu conteúdo, não como uma contabilidade completa de todas as solicitações. O CloudFront entrega logs de acesso com base no melhor esforço. A entrada do log de uma solicitação específica pode ser entregue muito depois do processamento da solicitação e, raramente, nunca ser entregue. Quando uma entrada de log for omitida dos logs de acesso, o número de entradas nos logs não corresponderá ao uso exibido nos relatórios de uso e faturamento da AWS.

Escolher um bucket do Amazon S3 para logs padrão

Ao habilitar o registro de uma distribuição, você especifica o bucket do Amazon S3 no qual deseja que o CloudFront armazene os arquivos de log. Se você estiver usando o Amazon S3 como origem, recomendamos usar um bucket separado para os arquivos de log.

Você pode armazenar os arquivos de log de várias distribuições no mesmo bucket. Ao habilitar o registro, você pode especificar um prefixo opcional para os nomes dos arquivos, a fim de diferenciar quais arquivos de log são associados a quais distribuições.

Como escolher um bucket do S3

- O bucket deve ter a lista de controle de acesso (ACL) habilitada. Se você escolher um bucket sem a ACL habilitada no console do CloudFront, será exibida uma mensagem de erro. Consulte [Permissões necessárias para configurar o registro em log padrão e acessar os arquivos de log](#).
- Não escolha um bucket do Amazon S3 com [Propriedade do objeto do S3](#) configurado como imposto pelo proprietário do bucket. Essa configuração desativa as ACLs para o bucket e os objetos nele contidos, o que impede que o CloudFront entregue arquivos de log para o bucket.
- Não escolha um bucket do Amazon S3 nas Regiões da AWS a seguir. O CloudFront não entrega logs padrão para buckets nestas regiões:
 - África (Cape Town)
 - Ásia-Pacífico (Hong Kong)
 - Ásia-Pacífico (Hyderabad)
 - Ásia-Pacífico (Jacarta)
 - Ásia-Pacífico (Melbourne)
 - Oeste do Canadá (Calgary)
 - Europa (Milão)
 - Europa (Espanha)
 - Europa (Zurique)
 - Israel (Tel Aviv)
 - Oriente Médio (Barém)
 - Oriente Médio (Emirados Árabes Unidos)

Permissões necessárias para configurar o registro em log padrão e acessar os arquivos de log

Important

A partir de abril de 2023, será necessário habilitar as ACLs do S3 para novos buckets do S3 usados para os logs padrão do CloudFront. É possível habilitar ACLs ao [criar um bucket](#) ou habilitar ACLs para um [bucket existente](#).

Para obter mais informações sobre as alterações, consulte [Perguntas frequentes sobre configurações padrão para novos buckets do S3](#) no Guia do usuário do Amazon Simple Storage Service e em [Alerta: Alterações na segurança do Amazon S3 chegarão em abril de 2023](#) no blog de notícias da AWS.

Sua Conta da AWS deve ter as seguintes permissões para o bucket especificado para os arquivos de log:

- A ACL do bucket deve conceder FULL_CONTROL a você. Se você for o proprietário do bucket, sua conta terá essa permissão por padrão. Em caso negativo, o proprietário do bucket deverá atualizar a ACL do bucket.
- s3:GetBucketAcl
- s3:PutBucketAcl

ACL do bucket

Ao criar ou atualizar uma distribuição e habilitar o registro em log, o CloudFront usa essas permissões para atualizar a ACL do bucket a fim de fornecer à conta `awslogsdelivery` a permissão FULL_CONTROL. A conta `awslogsdelivery` grava arquivos de log no bucket. Caso sua conta não tenha as permissões necessárias para atualizar a ACL, ocorrerá uma falha na criação ou atualização da distribuição.

Em alguns casos, se você enviar, de forma programática, uma solicitação para criar um bucket, mas já existir um bucket com o nome especificado, o S3 redefinirá as permissões do bucket para o valor padrão. Se você configurou o CloudFront para salvar os logs de acesso em um bucket do S3 e interromper o registro nesse bucket, verifique as permissões dele para garantir que o CloudFront tenha as permissões necessárias.

Restauração da ACL do bucket

Se você remover as permissões da conta `awslogsdelivery`, o CloudFront não poderá salvar os logs no bucket do S3. Para permitir que o CloudFront comece a salvar logs para sua distribuição novamente, restaure a permissão da ACL de uma das seguintes maneiras:

- Desabilite o registro em log de sua distribuição no CloudFront e habilite-o novamente. Para ter mais informações, consulte [Registro em log](#).
- Adicione a permissão da ACL para `awslogsdelivery` manualmente navegando até o bucket do S3 no console do Amazon S3 e adicionando a permissão. Para adicionar a ACL para `awslogsdelivery`, você deve fornecer o ID canônico da conta, que é o seguinte:

```
c4c1ede66af53448b93c283ce9448c4ba468c9432aa01d700d3878632f77d2d0
```

Consulte mais informações sobre como adicionar ACLs a buckets do S3 em [Configurar ACLs](#) no Guia do usuário do Amazon Simple Storage Service.

ACL para cada arquivo de log

Além da ACL no bucket, há uma ACL em cada arquivo de log. O proprietário do bucket tem permissão `FULL_CONTROL` em cada arquivo de log, o proprietário da distribuição (se não for o proprietário do bucket) não tem permissão e a conta `awslogsdelivery` tem permissões de leitura e de gravação.

Como desabilitar o registro

Se você desabilitar o registro, o CloudFront não excluirá as ACLs do bucket nem dos arquivos de log. Você pode excluir as ACLs, se necessário.

Política de chaves obrigatória para buckets do SSE-KMS

Se o bucket do S3 para seus logs padrão usar criptografia no lado do servidor com AWS KMS keys (SSE-KMS) usando uma chave gerenciada pelo cliente, você deverá adicionar a seguinte instrução à política de chaves para sua chave gerenciada pelo cliente. Isso permite que o CloudFront grave arquivos de log no bucket. Não é possível usar a SSE-KMS com a Chave gerenciada pela AWS porque o CloudFront não poderá gravar arquivos de log no bucket.

```
{  
  "Sid": "Allow CloudFront to use the key to deliver logs",  
  "Effect": "Allow",  
  "Principal": {
```

```

    "Service": "delivery.logs.amazonaws.com"
  },
  "Action": "kms:GenerateDataKey*",
  "Resource": "*"
}

```

Se o bucket do S3 para seus logs padrão usar SSE-KMS com uma [Chave de bucket do S3](#), você também precisará adicionar a permissão `kms:Decrypt` à instrução de política. Nesse caso, a declaração de política completa se parece com a seguinte.

```

{
  "Sid": "Allow CloudFront to use the key to deliver logs",
  "Effect": "Allow",
  "Principal": {
    "Service": "delivery.logs.amazonaws.com"
  },
  "Action": [
    "kms:GenerateDataKey*",
    "kms:Decrypt"
  ],
  "Resource": "*"
}

```

Formato do nome do arquivo

O nome de cada arquivo de log salvo pelo CloudFront no seu bucket do Amazon S3 usa o seguinte formato do nome do arquivo:

<optional prefix>/<distribution ID>.YYYY-MM-DD-HH.unique-ID.gz

A data e a hora estão em Tempo Universal Coordenado (UTC).

Por exemplo, se você usar `example-prefix` como prefixo e seu ID de distribuição for `EMLARXS9EXAMPLE`, seus nomes de arquivo serão semelhantes a este:

`example-prefix/EMLARXS9EXAMPLE.2019-11-14-20.RT4KCN4SGK9.gz`

Ao habilitar o registro para uma distribuição, você pode especificar um prefixo opcional para os nomes dos arquivos, a fim de diferenciar quais arquivos de log são associados a quais distribuições. Se você incluir um valor para o prefixo do arquivo de log e seu prefixo não terminar com uma barra (/), o CloudFront acrescentará uma automaticamente. Se o seu prefixo terminar com uma barra, o CloudFront não adicionará outra.

O `.gz` no fim do nome do arquivo indica que o CloudFront compactou o arquivo de log usando gzip.

Tempo de entrega do arquivo de log padrão

O CloudFront oferece logs padrão para uma distribuição até várias vezes por hora. Em geral, um arquivo de log contém informações sobre as solicitações recebidas pelo CloudFront durante um período específico. O CloudFront geralmente entrega o arquivo de log desse período no seu bucket do Amazon S3 em até uma hora após os eventos exibidos no log. No entanto, observe que algumas ou todas as entradas do arquivo de log referentes a um período podem demorar até 24 horas.

Quando entradas de log atrasam, o CloudFront as salva em um arquivo de log no qual o nome do arquivo inclui a data e a hora do período de ocorrência das solicitações, não de entrega do arquivo.

Ao criar um arquivo de log, o CloudFront consolida as informações da sua distribuição de todos os pontos de presença que receberam solicitações de seus objetos durante o período de cobertura do arquivo de log.

O CloudFront pode salvar mais de um arquivo por período, dependendo da quantidade de solicitações recebidas pelo CloudFront dos objetos associados a uma distribuição.

O CloudFront começa a entregar os logs de acesso cerca de quatro horas depois de você habilitar o registro. É possível que você receba alguns logs de acesso antes disso.

Note

Se nenhum usuário solicitar seus objetos nesse período, você não receberá arquivos de log referentes a ele.

O CloudFront também oferece logs em tempo real, que fornecem informações sobre solicitações feitas em uma distribuição em tempo real (os logs são entregues em segundos após o recebimento das solicitações). É possível usar os logs em tempo real para monitorar, analisar e tomar ações com base na performance da entrega de conteúdo. Para obter mais informações, consulte [Usar logs em tempo real](#).

Como as solicitações são registradas em log quando os cabeçalhos ou o URL de solicitação excedem o tamanho máximo

Se o tamanho total de todos os cabeçalhos de solicitação, incluindo cookies, exceder 20 KB, ou se o URL exceder 8192 bytes, o CloudFront não poderá analisar a solicitação completamente e não

poderá registrá-la. Como a solicitação não está conectada, você não verá no log os arquivos que o código de status de erro HTTP retornou.

Se o corpo da solicitação exceder o tamanho máximo, a solicitação será registrada, incluindo o código de status de erro HTTP.

Analisar logs padrão

Como você pode receber vários logs de acesso por hora, recomendamos que combine todos os arquivos de log recebidos em um determinado período em um único arquivo. Assim você poderá analisar os dados desse período de forma mais precisa e completa.

Uma forma de analisar seus logs de acesso é usar o [Amazon Athena](#). O Athena é um serviço de consulta interativo que pode ajudar você a analisar dados de serviços da AWS, incluindo o CloudFront. Para saber mais, consulte [Consultar logs do Amazon CloudFront](#) no Guia do usuário do Amazon Athena.

Além disso, as seguintes postagens de blog da AWS discutem algumas maneiras de analisar os logs de acesso.

- [Registro em log de solicitações do Amazon CloudFront](#) (para conteúdo distribuído via HTTP)
- [Logs avançados do CloudFront, agora com strings de consulta](#)

Important

Recomendamos que você use os logs para compreender a natureza das solicitações do seu conteúdo, não como uma contabilidade completa de todas as solicitações. O CloudFront entrega logs de acesso com base no melhor esforço. A entrada do log de uma solicitação específica pode ser entregue muito depois do processamento da solicitação e, raramente, nunca ser entregue. Quando uma entrada de log é omitida dos logs de acesso, o número de entradas nos logs não corresponderá ao uso exibido nos relatórios de uso e faturamento da AWS.

Editar as configurações do registro em log padrão

Você pode habilitar ou desabilitar o registro, alterar o bucket do Amazon S3 no qual os logs são armazenados e alterar o prefixo dos arquivos de log usando o console ou a API do [CloudFront](#). As alterações feitas nas configurações de registro são aplicadas em até 12 horas.

Para obter mais informações, consulte os tópicos a seguir:

- Para atualizar uma distribuição usando o console do CloudFront, consulte [Atualizar uma distribuição](#).
- Para atualizar uma distribuição usando a API do CloudFront, consulte [UpdateDistribution](#) na Referência de API do Amazon CloudFront.

Excluir arquivos de log padrão de um bucket do Amazon S3

O CloudFront não exclui automaticamente arquivos de log do seu bucket do Amazon S3. Para obter informações sobre como excluir arquivos de log de um bucket do Amazon S3, consulte os seguintes tópicos:

- Com o console do Amazon S3: [Excluir objetos](#) no Guia do usuário do console do Amazon Simple Storage Service.
- Usando a API REST: [DeleteObject](#) na Referência da API do Amazon Simple Storage Service.

Formato de arquivo de log padrão

Cada entrada em um arquivo de log fornece detalhes sobre uma única solicitação do visualizador. Os arquivos de log têm as seguintes características:

- Use o [formato do arquivo de log estendido do W3C](#).
- Contêm valores separados por tabulação.
- Contêm registros não necessariamente em ordem cronológica.
- Contém duas linhas de cabeçalho: uma com a versão do formato do arquivo e outra que relaciona os campos do W3C incluídos em cada registro.
- Contém equivalentes codificados em URL para espaços e alguns outros caracteres em valores de campo.

Equivalentes codificados em URL são usados para os seguintes caracteres:

- Códigos de caracteres ASCII 0 a 32, inclusive
- Códigos de caracteres ASCII 127 e posterior
- Todos os caracteres na tabela a seguir

O padrão de codificação de URL é definido no [RFC 1738](#).

Valor codificado em URL	Caractere
%3C	<
%3E	>
%22	"
%23	#
%25	%
%7B	{
%7D	}
%7C	
%5C	\
%5E	^
%7E	~
%5B	[
%5D]
%60	`
%27	'
%20	espaço

Campos padrão de arquivo de log

O arquivo de log para uma distribuição contém 33 campos. A lista a seguir contém cada nome de campo, em ordem, juntamente com uma descrição das informações nesse campo.

1. **date**

A data em que o evento ocorreu no formato YYYY-MM-DD. Por exemplo, 2019-06-30. A data e a hora estão em Tempo Universal Coordenado (UTC). Para conexões WebSockets, esta é a data em que a conexão foi encerrada.

2. **time**

A hora em que o servidor do CloudFront terminou de responder à solicitação (em UTC), por exemplo, 01:42:39. Para conexões WebSockets, este é o momento em que a conexão é fechada.

3. **x-edge-location**

O ponto de presença que atendeu à solicitação. Cada ponto de presença é identificado por um código de três letras e um número atribuído arbitrariamente (por exemplo, DFW3). O código de três letras normalmente corresponde ao código da Associação Internacional de Transportes Aéreos (IATA) de um aeroporto perto da localização geográfica do local da borda. (Essas abreviações podem mudar no futuro.)

4. **sc-bytes**

O número total de bytes enviados pelo servidor para o visualizador em resposta à solicitação, inclusive os cabeçalhos. Para conexões WebSockets, este é o número total de bytes enviados do servidor para o cliente por meio da conexão.

5. **c-ip**

O endereço IP do visualizador que fez a solicitação, por exemplo, 192.0.2.183 ou 2001:0db8:85a3::8a2e:0370:7334. Se o visualizador usar um proxy HTTP ou um load balancer para enviar a solicitação, o valor desse campo será o endereço IP do proxy ou do load balancer. Veja também o campo `x-forwarded-for`.

6. **cs-method**

O método de solicitação HTTP recebido do visualizador.

7. **cs(Host)**

O nome de domínio da distribuição do CloudFront (por exemplo, d111111abcdef8.cloudfront.net).

8. **cs-uri-stem**

A parte do URL da solicitação que identifica o caminho e o objeto (por exemplo, /images/cat.jpg). Os pontos de interrogação (?) em URLs e strings de consulta não são incluídos no log.

9. **sc-status**

Contém um dos seguintes valores:

- O código de status HTTP da resposta do servidor (por exemplo, 200).
- 000, que indica que o visualizador fechou a conexão antes que o servidor pudesse responder à solicitação. Se o visualizador fecha a conexão após o servidor começar a enviar a resposta, esse campo contém o código de status HTTP da resposta que o servidor começou a enviar.

10.**cs(Referer)**

O valor do cabeçalho `Referer` na solicitação. Esse é o nome do domínio que originou a solicitação. Indicadores comuns incluem: mecanismos de pesquisa, outros sites vinculados diretamente aos seus objetos e seu próprio site.

11.**cs(User-Agent)**

O valor do cabeçalho `User-Agent` na solicitação. O cabeçalho `User-Agent` identifica a origem da solicitação, como o tipo de dispositivo e o navegador que enviou a solicitação e, se a solicitação for proveniente de um mecanismo de pesquisa, o mecanismo de pesquisa.

12.**cs-uri-query**

A parte da string de consulta do URL da solicitação, se houver.

Quando um URL não contém uma string de consulta, o valor desse campo é um hífen (-). Para ter mais informações, consulte [Conteúdo em cache com base em parâmetros de string de consulta](#).

13.**cs(Cookie)**

O cabeçalho `Cookie` na solicitação, incluindo pares de nome-valor e os atributos associados.

Se você habilitar o registro de cookies, o CloudFront os registrará em todas as solicitações, independentemente de quais você optar por encaminhar para a origem. Quando uma solicitação não inclui um cabeçalho de cookie, o valor desse campo é um hífen (-). Para obter mais informações sobre cookies, consulte [Conteúdo em cache com base em cookies](#).

14.**x-edge-result-type**

Como o servidor classificou a resposta após o último byte sair do servidor. Em alguns casos, o tipo de resultado pode mudar entre a hora em que o servidor está pronto para enviar a resposta e a hora em que ele conclui o envio. Veja também o campo `x-edge-response-result-type`.

Por exemplo, em streaming HTTP, suponha que o servidor encontre um segmento do stream no cache. Nesse cenário, o valor desse campo normalmente seria `Hit`. No entanto, se o visualizador encerrar a conexão antes de o servidor entregar o segmento inteiro, o tipo do resultado final (e, portanto, o valor desse campo) será `Error`.

As conexões WebSocket terão um valor de `Miss` para esse campo porque o conteúdo não é armazenável em cache e é enviado diretamente de volta ao servidor de origem.

Os possíveis valores incluem:

- `Hit`: o servidor forneceu o objeto do cache ao visualizador.
- `RefreshHit`: o servidor encontrou o objeto no cache, mas o objeto expirou, portanto, o servidor entrou em contato com a origem para verificar se o cache tinha a versão mais recente do objeto.
- `Miss`: não foi possível atender à solicitação por um objeto no cache e, portanto, o servidor a encaminhou ao servidor de origem e retornou o resultado ao visualizador.
- `LimitExceeded`: a solicitação foi negada porque uma cota do CloudFront (anteriormente conhecida como limite) foi excedida.
- `CapacityExceeded`: o servidor retornou um código de status HTTP 503 porque não tinha capacidade suficiente no momento da solicitação para fornecer o objeto.
- `Error`: normalmente, isso significa que a solicitação resultou em um erro de cliente (o valor do campo `sc-status` está no intervalo 4xx) ou em um erro de servidor (o valor do campo `sc-status` está no intervalo 5xx). Se o valor do campo `sc-status` for 200 ou se o valor desse campo for `Error` e o valor do campo `x-edge-response-result-type` não for `Error`, isso significa que a solicitação HTTP foi bem-sucedida, mas o cliente desconectou antes de receber todos os bytes.
- `Redirect`: o servidor redirecionou o visualizador de HTTP para HTTPS de acordo com as configurações de distribuição.

15x-edge-request-id

Uma string opaca que identifica exclusivamente uma solicitação. O CloudFront também envia essa string no cabeçalho de resposta `x-amz-cf-id`.

16x-host-header

O valor incluído pelo visualizador no cabeçalho `Host` da solicitação. Se você estiver usando o nome de domínio do CloudFront nos URLs de objetos (como `d111111abcdef8.cloudfront.net`),

esse campo conterá esse nome de domínio. Se você estiver usando nomes de domínio alternativos (CNAMES) nos URLs de objetos (como `www.example.com`), esse campo conterá o nome de domínio alternativo.

Se você estiver usando nomes de domínio alternativos, consulte `cs(Host)` no campo 7 do nome de domínio associado a sua distribuição.

17.**cs-protocol**

O protocolo da solicitação do visualizador (`http`, `https`, `ws` ou `wss`).

18.**cs-bytes**

O número de bytes de dados que o visualizador adicionou à solicitação, incluindo cabeçalhos. Para conexões WebSockets, este é o número total de bytes enviados do cliente para o servidor na conexão.

19.**time-taken**

O número de segundos (até o milésimo de segundo, por exemplo, `0,082`) de quando o servidor recebe a solicitação do visualizador até quando o servidor grava o último byte da resposta na fila de saída, conforme medido no servidor. Da perspectiva do visualizador, o tempo total para obter o objeto completo será mais longo que esse valor devido à latência da rede e o armazenamento em buffer do TCP.

20.**x-forwarded-for**

Se o visualizador usar um proxy HTTP ou um load balancer para enviar a solicitação, o valor do campo `c-ip` será o endereço IP do proxy ou do load balancer. Nesse caso, esse campo é o endereço IP do visualizador que originou a solicitação. Esse campo pode conter vários endereços IP separados por vírgula. Cada endereço IP pode ser um endereço IPv4 (por exemplo, `192.0.2.183`) ou um endereço IPv6 (por exemplo, `2001:0db8:85a3::8a2e:0370:7334`).

Se o visualizador não tiver usado um proxy HTTP ou um load balancer, o valor deste campo será um hífen (-).

21.**ssl-protocol**

Quando a solicitação usa HTTPS, esse campo contém o protocolo SSL/TLS que o visualizador e o servidor negociaram para transmitir a solicitação e a resposta. Para obter uma lista de valores possíveis, consulte os protocolos SSL/TLS compatíveis em [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

Quando `cs-protocol` no campo 17 for `http`, o valor desse campo será um hífen (-).

22`ssl-cipher`

Quando a solicitação usa HTTPS, esse campo contém a cifra SSL/TLS que o visualizador e o servidor negociaram para criptografar a solicitação e a resposta. Para obter uma lista de valores possíveis, consulte as cifras SSL/TLS compatíveis em [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

Quando `cs-protocol` no campo 17 for `http`, o valor desse campo será um hífen (-).

23`x-edge-response-result-type`

Como o servidor classificou a resposta logo antes de devolvê-la para o visualizador. Veja também o campo `x-edge-result-type`. Os possíveis valores incluem:

- `Hit`: o servidor forneceu o objeto do cache ao visualizador.
- `RefreshHit`: o servidor encontrou o objeto no cache, mas o objeto expirou, portanto, o servidor entrou em contato com a origem para verificar se o cache tinha a versão mais recente do objeto.
- `Miss`: não foi possível atender à solicitação por um objeto no cache, portanto, o servidor a encaminhou ao servidor de origem e retornou o resultado ao visualizador.
- `LimitExceeded`: a solicitação foi negada porque uma cota do CloudFront (anteriormente conhecida como limite) foi excedida.
- `CapacityExceeded`: o servidor retornou um erro 503 porque não tinha capacidade suficiente no momento da solicitação para fornecer o objeto.
- `Error`: normalmente, isso significa que a solicitação resultou em um erro de cliente (o valor do campo `sc-status` está no intervalo 4xx) ou em um erro de servidor (o valor do campo `sc-status` está no intervalo 5xx).

Se o valor do campo `x-edge-result-type` for `Error` e o valor desse campo não for `Error`, o cliente desconectou antes de concluir o download.

- `Redirect`: o servidor redirecionou o visualizador de HTTP para HTTPS de acordo com as configurações de distribuição.

24`cs-protocol-version`

A versão HTTP especificada pelo visualizador na solicitação. Os valores possíveis incluem `HTTP/0.9`, `HTTP/1.0`, `HTTP/1.1`, `HTTP/2.0` e `HTTP/3.0`.

25.file-status

Quando a [criptografia em nível de campo](#) é configurada para uma distribuição, esse campo contém um código que indica se o corpo da solicitação foi processado com êxito. Quando o servidor processa o corpo da solicitação, criptografa os valores nos campos especificados e encaminha a solicitação para a origem com êxito, o valor desse campo é Processed. Nesse caso, o valor de `x-edge-result-type` pode indicar um erro do lado do cliente ou do lado do servidor.

Os valores possíveis para esse campo incluem:

- `ForwardedByContentType`: o servidor encaminhou a solicitação para a origem sem análise nem criptografia, pois nenhum tipo de conteúdo foi configurado.
- `ForwardedByQueryArgs`: o servidor encaminhou a solicitação para a origem sem análise nem criptografia, pois a solicitação contém um argumento de consulta que não foi configurado para a criptografia em nível de campo.
- `ForwardedDueToNoProfile`: o servidor encaminhou a solicitação para a origem sem análise nem criptografia, pois nenhum perfil foi especificado na configuração da criptografia em nível de campo.
- `MalformedContentTypeClientError`: o servidor rejeitou a solicitação e retornou o código de status HTTP 400 para o visualizador, pois o valor do cabeçalho `Content-Type` estava em um formato inválido.
- `MalformedInputClientError`: o servidor rejeitou a solicitação e retornou o código de status HTTP 400 para o visualizador, pois o corpo da solicitação estava em um formato inválido.
- `MalformedQueryArgsClientError`: o servidor rejeitou a solicitação e retornou o código de status HTTP 400 para o visualizador, pois um argumento de consulta estava vazio ou em um formato inválido.
- `RejectedByContentType`: o servidor rejeitou a solicitação e retornou o código de status HTTP 400 para o visualizador, pois nenhum tipo de conteúdo foi especificado na configuração para criptografia em nível de campo.
- `RejectedByQueryArgs`: o servidor rejeitou a solicitação e retornou o código de status HTTP 400 para o visualizador, pois nenhum argumento de consulta foi especificado na configuração para criptografia em nível de campo.
- `ServerError`: o servidor de origem retornou um erro.

Se a solicitação exceder uma cota de criptografia em nível de campo (anteriormente conhecida como limite), esse campo conterá um dos seguintes códigos de erro, e o servidor retornará o

código de status HTTP 400 ao visualizador. Para obter uma lista das cotas atuais de criptografia no nível de campo, consulte [Cotas para criptografia no nível de campo](#).

- `FieldLengthLimitClientError`: um campo configurado para ser criptografado excedeu o tamanho máximo permitido.
- `FieldNumberLimitClientError`: uma solicitação que a distribuição está configurada para criptografar contém o número de campos maior do que o permitido.
- `RequestLengthLimitClientError`: o tamanho do corpo da solicitação excedeu o tamanho máximo permitido quando a criptografia no nível de campo foi configurada.

Se a criptografia no nível de campo não estiver configurada para a distribuição, o valor desse campo será um hífen (-).

26.fle-encrypted-fields

O número de campos de [criptografia em nível de campo](#) que o servidor de borda criptografou e encaminhou para a origem. Os servidores do CloudFront fazem streaming da solicitação processada para a origem à medida que criptografam dados, portanto, esse campo pode ter um valor, mesmo que o valor de `fle-status` seja um erro.

Se a criptografia no nível de campo não estiver configurada para a distribuição, o valor desse campo será um hífen (-).

27.c-port

O número da porta da solicitação do visualizador.

28.time-to-first-byte

O número de segundos entre o recebimento da solicitação e a gravação do primeiro byte da resposta, conforme medido no servidor.

29x-edge-detailed-result-type

Esse campo conterá o mesmo valor que o campo `x-edge-result-type`, exceto nos seguintes casos:

- Quando o objeto for enviado ao visualizador do cache da camada [Origin Shield](#), esse campo conterá `OriginShieldHit`.
- Quando o objeto não estiver no cache do CloudFront e a resposta for gerada por uma [função Lambda@Edge de solicitação de origem](#), esse campo conterá `MissGeneratedResponse`.

- Quando o valor do campo `x-edge-result-type` for `Error`, esse campo conterá um dos seguintes valores com mais informações sobre o erro:
 - `AbortedOrigin`: o servidor encontrou um problema com a origem.
 - `ClientCommError`: a resposta ao visualizador foi interrompida devido a um problema de comunicação entre o servidor e o visualizador.
 - `ClientGeoBlocked`: a distribuição é configurada para recusar solicitações da localização geográfica do visualizador.
 - `ClientHungUpRequest` – o visualizador parou prematuramente ao enviar a solicitação.
 - `Error`: ocorreu um erro para o qual o tipo de erro não se encaixa em nenhuma das outras categorias. Esse tipo de erro pode ocorrer quando o servidor fornece uma resposta de erro do cache.
 - `InvalidRequest`: o servidor recebeu uma solicitação inválida do visualizador.
 - `InvalidRequestBlocked` – o acesso ao recurso solicitado é bloqueado.
 - `InvalidRequestCertificate`: a distribuição não corresponde ao certificado SSL/TLS para o qual a conexão HTTPS foi estabelecida.
 - `InvalidRequestHeader`: a solicitação continha um cabeçalho inválido.
 - `InvalidRequestMethod` – a distribuição não está configurada para lidar com o método de solicitação HTTP que foi usado. Isso pode acontecer quando a distribuição oferece suporte somente a solicitações armazenáveis em cache.
 - `OriginCommError`: a solicitação expirou durante a conexão à origem ou a leitura de dados da origem.
 - `OriginConnectError`: o servidor não pôde se conectar à origem.
 - `OriginContentRangeLengthError`: o cabeçalho `Content-Length` na resposta da origem não corresponde ao tamanho no cabeçalho `Content-Range`.
 - `OriginDnsError`: o servidor não pôde resolver o nome de domínio da origem.
 - `OriginError` – a origem retornou uma resposta incorreta.
 - `OriginHeaderTooBigError`: um cabeçalho retornado pela origem é muito grande para o processamento pelo servidor de borda.
 - `OriginInvalidResponseError` – a origem retornou uma resposta inválida.
 - `OriginReadError`: o servidor não pôde ler na origem.
 - `OriginWriteError`: o servidor não pôde gravar na origem.

- `OriginZeroSizeObjectError` – um objeto de tamanho zero enviado da origem resultou em um erro.
- `SlowReaderOriginError` – o visualizador ficou lento ao ler a mensagem que causou o erro de origem.

30 `sc-content-type`

O valor do cabeçalho do HTTP Content-Type da resposta.

31 `sc-content-len`

O valor do cabeçalho do HTTP Content-Length da resposta.

32 `sc-range-start`

Quando a resposta contém o cabeçalho do HTTP Content-Range, esse campo contém o valor inicial do intervalo.

33 `sc-range-end`

Quando a resposta contém o cabeçalho do HTTP Content-Range, esse campo contém o valor final do intervalo.

Veja abaixo um exemplo de arquivo de log para uma distribuição:

```
#Version: 1.0
#Fields: date time x-edge-location sc-bytes c-ip cs-method cs(Host) cs-uri-stem sc-
status cs(Referer) cs(User-Agent) cs-uri-query cs(Cookie) x-edge-result-type x-edge-
request-id x-host-header cs-protocol cs-bytes time-taken x-forwarded-for ssl-protocol
ssl-cipher x-edge-response-result-type cs-protocol-version fle-status fle-encrypted-
fields c-port time-to-first-byte x-edge-detailed-result-type sc-content-type sc-
content-len sc-range-start sc-range-end
2019-12-04 21:02:31 LAX1 392 192.0.2.100 GET d111111abcdef8.cloudfront.net /
index.html 200 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;
%20x64)%20AppleWebKit/537.36%20(KHTML,%20like
%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Hit
SOX4xwn4XV6Q4rgb7XiVG0Hms_BGLTAC4KyHmureZmBNrjGdRLiNIQ== d111111abcdef8.cloudfront.net
https 23 0.001 - TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 Hit HTTP/2.0 - - 11040 0.001 Hit
text/html 78 - -
2019-12-04 21:02:31 LAX1 392 192.0.2.100 GET d111111abcdef8.cloudfront.net /
index.html 200 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;
%20x64)%20AppleWebKit/537.36%20(KHTML,%20like
%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Hit
```

```
k6WGMNkEzR5BEM_SaF47gjtX9zBD02m3490Y2an0QPEaUum1Z0Lrow== d111111abcdef8.cloudfront.net
https 23 0.000 - TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 Hit HTTP/2.0 - - 11040 0.000 Hit
text/html 78 - -
2019-12-04 21:02:31 LAX1 392 192.0.2.100 GET d111111abcdef8.cloudfront.net /
index.html 200 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;
%20x64)%20AppleWebKit/537.36%20(KHTML,%20like
%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Hit
f37nTMVvnKvV2ZSvEsivup_c2kZ7VXzYdjC-GUQZ5qNs-89BlWazbw== d111111abcdef8.cloudfront.net
https 23 0.001 - TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 Hit HTTP/2.0 - - 11040 0.001 Hit
text/html 78 - -
2019-12-13 22:36:27 SEA19-C1 900 192.0.2.200 GET d111111abcdef8.cloudfront.net /
favicon.ico 502 http://www.example.com/ Mozilla/5.0%20(Windows
%20NT%2010.0;%20Win64;%20x64)%20AppleWebKit/537.36%20(KHTML,
%20like%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Error
1pkpNfBQ39sYMNjjUQjmH2w1wdJnbHYTbag21o_30fcQgPzdL2RSSQ== www.example.com http 675
0.102 - - - Error HTTP/1.1 - - 25260 0.102 OriginDnsError text/html 507 - -
2019-12-13 22:36:26 SEA19-C1 900 192.0.2.200 GET d111111abcdef8.cloudfront.net / 502
- Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;%20x64)%20AppleWebKit/537.36%20(KHTML,
%20like%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Error
3AqrZGcNf_g0-5K0vfA7c9XLcf4YGvMFSeFdIetR1N_2y8jSis8Zxg== www.example.com http 735
0.107 - - - Error HTTP/1.1 - - 3802 0.107 OriginDnsError text/html 507 - -
2019-12-13 22:37:02 SEA19-C2 900 192.0.2.200 GET d111111abcdef8.cloudfront.net / 502
- curl/7.55.1 - - Error kBkDzGnceVtWHqSCqBUqtA_cEs2T3tFUBbnBNkB9E1_uVRhHgcZfcw==
www.example.com http 387 0.103 - - - Error HTTP/1.1 - - 12644 0.103 OriginDnsError
text/html 507 - -
```

Cobranças de logs padrão

O registro em log padrão é um recurso opcional do CloudFront. Não há cobrança adicional para a habilitação do registro em log padrão. No entanto, você acumula as cobranças normais do Amazon S3 para armazenar e acessar os arquivos nele (é possível excluí-los a qualquer momento).

Para mais informações sobre a definição de preço do Amazon S3, consulte [Definição de preço do Amazon S3](#).

Para mais informações sobre preços do CloudFront, consulte [Definição de preço do CloudFront](#).

Usar logs em tempo real

Com os logs em tempo real do CloudFront, é possível obter informações sobre solicitações feitas para uma distribuição em tempo real (os logs são entregues em segundos após o recebimento das

solicitações). É possível usar os logs em tempo real para monitorar, analisar e tomar ações com base na performance da entrega de conteúdo.

Os logs em tempo real do CloudFront são configuráveis. É possível escolher:

- A taxa de amostragem dos logs em tempo real, ou seja, a porcentagem de solicitações para as quais deseja receber registros de log em tempo real.
- Os campos específicos que você deseja receber nos registros de log.
- Os comportamentos de cache específicos (padrões de caminho) dos quais você deseja receber logs em tempo real.

Os logs em tempo real do CloudFront são entregues ao stream de dados de sua escolha no Amazon Kinesis Data Streams. É possível criar seu próprio [consumidor de fluxo de dados do Kinesis](#) ou usar o Amazon Data Firehose para enviar dados de log para o Amazon Simple Storage Service (Amazon S3), Amazon Redshift, Amazon OpenSearch Service (OpenSearch Service) ou um serviço de processamento de log de terceiros.

O CloudFront cobra por logs em tempo real, além das cobranças que você incorre pelo uso do Kinesis Data Streams. Para obter mais informações sobre definição de preço, consulte [Definição de preço do Amazon CloudFront](#) e [Definição de preço do Amazon Kinesis Data Streams](#).

Important

Recomendamos que você use os logs para compreender a natureza das solicitações do seu conteúdo, não como uma contabilidade completa de todas as solicitações. O CloudFront entrega logs em tempo real com base no melhor esforço. A entrada do log de uma solicitação específica pode ser entregue muito depois do processamento da solicitação e, raramente, nunca ser entregue. Quando uma entrada de log for omitida dos logs em tempo real, o número de entradas nos logs não corresponderá ao uso exibido nos relatórios de uso e faturamento da AWS.

Tópicos

- [Criar e usar configurações de log em tempo real](#)
- [Noções básicas sobre as configurações de log em tempo real](#)
- [Criar um consumidor do Kinesis Data Streams](#)
- [Solução de problemas de logs em tempo real](#)

Criar e usar configurações de log em tempo real

Para ter informações sobre solicitações feitas a uma distribuição em tempo real, use as configurações de log em tempo real. Os logs são entregues em segundos após o recebimento das solicitações. É possível criar uma configuração de log em tempo real no console do CloudFront com a AWS Command Line Interface (AWS CLI) ou a API do CloudFront.

Para usar uma configuração de log em tempo real, anexe-a a um ou mais comportamentos de cache em uma distribuição do CloudFront.

Console

Como criar uma configuração de log em tempo real

1. Faça login no AWS Management Console e abra a página Logs no console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home?#/logs>.
2. Selecione a guia Real-time log configurations (Configurações de log em tempo real).
3. Escolha Criar configuração.
4. Em Name (Nome), insira um nome para a configuração.
5. Em Sampling rate (Taxa de amostragem), insira a porcentagem de solicitações para as quais deseja receber registros de log.
6. Em Fields (Campos), escolha os campos a serem recebidos nos registros de log em tempo real.
 - Para incluir todos os [campos CMCD](#) em seus registros de log, selecione CMCD all keys (todas as chaves CMCD).
7. Em Endpoint, selecione um ou mais fluxos de dados do Kinesis para receber registros de log em tempo real.

Note

Os registros de logs em tempo real do CloudFront são entregues ao fluxo de dados especificado no Kinesis Data Streams. Para ler e analisar os registros de logs em tempo real, é possível criar seu próprio consumidor de fluxo de dados do Kinesis. Também é possível usar o Firehose para enviar dados de log para o Amazon S3, Amazon Redshift, Amazon OpenSearch Service ou um serviço de processamento de log de terceiros.

8. Em IAM role (Perfil do IAM), selecione Create new service role (Criar novo perfil de serviço) ou selecione um perfil existente. Você deve ter permissão para criar funções do IAM.
9. (Opcional) Em Distribution (Distribuição), selecione um comportamento de cache e uma distribuição do CloudFront a serem anexados à configuração de log em tempo real.
10. Escolha Criar configuração.

Se tiver êxito, o console mostrará os detalhes da configuração de log em tempo real que você acabou de criar.

Para ter mais informações, consulte [Noções básicas sobre as configurações de log em tempo real](#).

AWS CLI

Para criar uma configuração de log em tempo real com a AWS CLI, use o comando `aws cloudfront create-realtime-log-config`. É possível usar um arquivo de entrada para fornecer os parâmetros de entrada do comando, em vez de especificar cada parâmetro individual como entrada na linha de comando.

Como criar uma configuração de log em tempo real (CLI com arquivo de entrada)

1. Use o comando a seguir para criar um arquivo chamado `rtl-config.yaml` que contém todos os parâmetros de entrada para o comando `create-realtime-log-config`.

```
aws cloudfront create-realtime-log-config --generate-cli-skeleton yaml-input >
rtl-config.yaml
```

2. Abra o arquivo chamado `rtl-config.yaml` que você acabou de criar. Edite o arquivo para especificar as configurações de log em tempo real desejadas e salve o arquivo. Observe o seguinte:

- Para `StreamType`, o único valor válido é `Kinesis`.

Para obter mais informações sobre as configurações de log em tempo real, consulte [Noções básicas sobre as configurações de log em tempo real](#).

3. Use o comando a seguir para criar a configuração de log em tempo real usando parâmetros de entrada do arquivo `rtl-config.yaml`.

```
aws cloudfront create-realtime-log-config --cli-input-yaml file://rtl-
config.yaml
```

Se tiver êxito, a saída do comando mostrará os detalhes da configuração de log em tempo real que você acabou de criar.

Como anexar uma configuração de log em tempo real a uma distribuição existente (CLI com arquivo de entrada)

1. Use o comando a seguir para salvar a configuração da distribuição do CloudFront que você deseja atualizar. Substitua *distribution_ID* pelo ID da distribuição.

```
aws cloudfront get-distribution-config --id distribution_ID --output yaml >
dist-config.yaml
```

2. Abra o arquivo chamado `dist-config.yaml` que você acabou de criar. Edite o arquivo, fazendo as seguintes alterações em cada comportamento de cache que você está atualizando para usar uma configuração de log em tempo real.
 - No comportamento de cache, adicione um campo chamado `RealtimeLogConfigArn`. Para o valor do campo, use o ARN da configuração de log em tempo real que você deseja anexar a esse comportamento de cache.
 - Renomeie o campo `Etag` para `IfMatch`, mas não altere o valor do campo.

Ao concluir, salve o arquivo.

3. Use o comando a seguir para atualizar a distribuição para usar a configuração de log em tempo real. Substitua *distribution_ID* pelo ID da distribuição.

```
aws cloudfront update-distribution --id distribution_ID --cli-input-yaml file://
dist-config.yaml
```

Se tiver êxito, a saída do comando mostrará os detalhes da distribuição que você acabou de atualizar.

API

Para criar uma configuração de log em tempo real com a API do CloudFront, use a operação de API [CreateRealtimeLogConfig](#). Para obter mais informações sobre os parâmetros especificados nessa chamada de API, consulte [Noções básicas sobre as configurações de log em tempo real](#) e a documentação de referência da API do seu SDK da AWS ou de outro cliente de API.

Depois de criar uma configuração de log em tempo real, é possível anexá-la a um comportamento de cache, usando uma das seguintes operações de API:

- Para anexá-la a um comportamento de cache em uma distribuição existente, use [UpdateDistribution](#).
- Para anexá-la a um comportamento de cache em uma nova distribuição, use [CreateDistribution](#).

Para ambas as operações de API, forneça o ARN da configuração de log em tempo real no campo `RealtimeLogConfigArn`, dentro de um comportamento de cache. Para mais informações sobre os outros campos especificados nessas chamadas de API, consulte [Referência de configurações da distribuição](#) e a documentação de referência da API do AWS SDK ou de outro cliente de API.

Noções básicas sobre as configurações de log em tempo real

Para usar os logs em tempo real do CloudFront, você começa criando uma configuração de log em tempo real. A configuração de log em tempo real contém informações sobre quais campos de log você deseja receber, a taxa de amostragem de registros de log e o stream de dados do Kinesis para o qual você deseja entregar os logs.

Especificamente, uma configuração de log em tempo real contém as seguintes configurações:

Sumário

- [Nome](#)
- [Taxa de amostragem](#)
- [Campos](#)
- [Endpoint \(Kinesis Data Streams\)](#)
- [IAM role \(Função do IAM\)](#)

Nome

Um nome para identificar a configuração do log em tempo real.

Taxa de amostragem

A taxa de amostragem é um número inteiro entre 1 e 100 (inclusive) que determina a porcentagem de solicitações de visualizador enviadas ao Kinesis Data Streams como registros de log em tempo real. Para incluir cada solicitação de visualizador em seus logs em tempo real, especifique 100 para a taxa de amostragem. É possível escolher uma taxa de amostragem mais baixa para reduzir custos enquanto ainda recebe uma amostra representativa de dados de solicitação em seus logs em tempo real.

Campos

Uma lista de campos incluídos em cada registro de log em tempo real. Cada registro em log pode conter até 40 campos, e é possível optar por receber todos os campos disponíveis ou apenas os campos necessários para monitorar e analisar a performance.

A lista a seguir contém cada nome de campo e uma descrição das informações nesse campo. Os campos são listados na ordem em que aparecem nos registros de log que são entregues ao Kinesis Data Streams.

Os campos 46–63 são [Common Media Client Data \(CMCD\)](#) que os clientes do media player podem enviar às CDNs com cada solicitação. É possível usar esses dados para entender cada solicitação, como o tipo de mídia (áudio, vídeo), a taxa de reprodução e a duração da transmissão. Esses campos só aparecerão em seus registros em log em tempo real se forem enviados para o CloudFront.

1. **timestamp**

A data e a hora em que o servidor de borda concluiu a resposta à solicitação.

2. **c-ip**

O endereço IP do visualizador que fez a solicitação, por exemplo, 192.0.2.183 ou 2001:0db8:85a3::8a2e:0370:7334. Se o visualizador usar um proxy HTTP ou um load balancer para enviar a solicitação, o valor desse campo será o endereço IP do proxy ou do load balancer. Veja também o campo `x-forwarded-for`.

3. **s-ip**

O endereço IP do servidor do CloudFront que atendeu à solicitação; por exemplo, 192.0.2.183 ou 2001:0db8:85a3::8a2e:0370:7334.

4. **time-to-first-byte**

O número de segundos entre o recebimento da solicitação e a gravação do primeiro byte da resposta, conforme medido no servidor.

5. **sc-status**

O código de status HTTP da resposta do servidor (por exemplo, 200).

6. **sc-bytes**

O número total de bytes enviados pelo servidor para o visualizador em resposta à solicitação, inclusive os cabeçalhos. Para conexões WebSockets, este é o número total de bytes enviados do servidor para o cliente por meio da conexão.

7. **cs-method**

O método de solicitação HTTP recebido do visualizador.

8. **cs-protocol**

O protocolo da solicitação do visualizador (http, https, ws ou wss).

9. **cs-host**

O valor incluído pelo visualizador no cabeçalho Host da solicitação. Se você estiver usando o nome de domínio do CloudFront nos URLs de objetos (como d111111abcdef8.cloudfront.net), esse campo conterá esse nome de domínio. Se você estiver usando nomes de domínio alternativos (CNAMES) nos URLs de objetos (como www.example.com), esse campo conterá o nome de domínio alternativo.

10. **cs-uri-stem**

Todo o URL da solicitação, inclusive a string de consulta (se houver), mas sem o nome do domínio. Por exemplo, /images/cat.jpg?mobile=true.

Note

Em [logs padrão](#), o valor de `cs-uri-stem` não inclui a string de consulta.

11. **cs-bytes**

O número de bytes de dados que o visualizador adicionou à solicitação, incluindo cabeçalhos. Para conexões WebSockets, este é o número total de bytes enviados do cliente para o servidor na conexão.

12x-edge-location

O ponto de presença que atendeu à solicitação. Cada ponto de presença é identificado por um código de três letras e um número atribuído arbitrariamente (por exemplo, DFW3). O código de três letras normalmente corresponde ao código da Associação Internacional de Transportes Aéreos (IATA) de um aeroporto perto da localização geográfica do local da borda. (Essas abreviações podem mudar no futuro.)

13x-edge-request-id

Uma string opaca que identifica exclusivamente uma solicitação. O CloudFront também envia essa string no cabeçalho de resposta `x-amz-cf-id`.

14x-host-header

O nome de domínio da distribuição do CloudFront (por exemplo, `d111111abcdef8.cloudfront.net`).

15time-taken

O número de segundos (até o milésimo de segundo, por exemplo, `0,082`) de quando o servidor recebe a solicitação do visualizador até quando o servidor grava o último byte da resposta na fila de saída, conforme medido no servidor. Da perspectiva do visualizador, o tempo total para obter o objeto completo será mais longo que esse valor devido à latência da rede e o armazenamento em buffer do TCP.

16cs-protocol-version

A versão HTTP especificada pelo visualizador na solicitação. Os valores possíveis incluem `HTTP/0.9`, `HTTP/1.0`, `HTTP/1.1`, `HTTP/2.0` e `HTTP/3.0`.

17c-ip-version

A versão IP da solicitação (IPv4 ou IPv6).

18cs-user-agent

O valor do cabeçalho `User-Agent` na solicitação. O cabeçalho `User-Agent` identifica a origem da solicitação, como o tipo de dispositivo e o navegador que enviou a solicitação e, se a solicitação for proveniente de um mecanismo de pesquisa, o mecanismo de pesquisa.

19.cs-referer

O valor do cabeçalho `Referer` na solicitação. Esse é o nome do domínio que originou a solicitação. Indicadores comuns incluem: mecanismos de pesquisa, outros sites vinculados diretamente aos seus objetos e seu próprio site.

20.cs-cookie

O cabeçalho `Cookie` na solicitação, incluindo pares de nome-valor e os atributos associados.

Note

Este campo é truncado em 800 bytes.

21.cs-uri-query

A parte da string de consulta do URL da solicitação, se houver.

22x-edge-response-result-type

Como o servidor classificou a resposta logo antes de devolvê-la para o visualizador. Veja também o campo `x-edge-result-type`. Os possíveis valores incluem:

- `Hit`: o servidor forneceu o objeto do cache ao visualizador.
- `RefreshHit`: o servidor encontrou o objeto no cache, mas o objeto expirou, portanto, o servidor entrou em contato com a origem para verificar se o cache tinha a versão mais recente do objeto.
- `Miss`: não foi possível atender à solicitação por um objeto no cache, portanto, o servidor a encaminhou ao servidor de origem e retornou o resultado ao visualizador.
- `LimitExceeded`: a solicitação foi negada porque uma cota do CloudFront (anteriormente conhecida como limite) foi excedida.
- `CapacityExceeded`: o servidor retornou um erro 503 porque não tinha capacidade suficiente no momento da solicitação para fornecer o objeto.
- `Error`: normalmente, isso significa que a solicitação resultou em um erro de cliente (o valor do campo `sc-status` está no intervalo 4xx) ou em um erro de servidor (o valor do campo `sc-status` está no intervalo 5xx).

Se o valor do campo `x-edge-result-type` for `Error` e o valor desse campo não for `Error`, o cliente desconectou antes de concluir o download.

- `Redirect`: o servidor redirecionou o visualizador de HTTP para HTTPS de acordo com as configurações de distribuição.

23x-forwarded-for

Se o visualizador usar um proxy HTTP ou um load balancer para enviar a solicitação, o valor do campo `c-ip` será o endereço IP do proxy ou do load balancer. Nesse caso, esse campo é o endereço IP do visualizador que originou a solicitação. Esse campo pode conter vários endereços IP separados por vírgula. Cada endereço IP pode ser um endereço IPv4 (por exemplo, `192.0.2.183`) ou um endereço IPv6 (por exemplo, `2001:0db8:85a3::8a2e:0370:7334`).

24ssl-protocol

Quando a solicitação usa HTTPS, esse campo contém o protocolo SSL/TLS que o visualizador e o servidor negociaram para transmitir a solicitação e a resposta. Para obter uma lista de valores possíveis, consulte os protocolos SSL/TLS compatíveis em [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

25ssl-cipher

Quando a solicitação usa HTTPS, esse campo contém a cifra SSL/TLS que o visualizador e o servidor negociaram para criptografar a solicitação e a resposta. Para obter uma lista de valores possíveis, consulte as cifras SSL/TLS compatíveis em [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

26x-edge-result-type

Como o servidor classificou a resposta após o último byte sair do servidor. Em alguns casos, o tipo de resultado pode mudar entre a hora em que o servidor está pronto para enviar a resposta e a hora em que ele conclui o envio. Veja também o campo `x-edge-response-result-type`.

Por exemplo, em streaming HTTP, suponha que o servidor encontre um segmento do stream no cache. Nesse cenário, o valor desse campo normalmente seria `Hit`. No entanto, se o visualizador encerrar a conexão antes de o servidor entregar o segmento inteiro, o tipo do resultado final (e, portanto, o valor desse campo) será `Error`.

As conexões WebSocket terão um valor de `Miss` para esse campo porque o conteúdo não é armazenável em cache e é enviado diretamente de volta ao servidor de origem.

Os possíveis valores incluem:

- `Hit`: o servidor forneceu o objeto do cache ao visualizador.

- **RefreshHit**: o servidor encontrou o objeto no cache, mas o objeto expirou, portanto, o servidor entrou em contato com a origem para verificar se o cache tinha a versão mais recente do objeto.
- **Miss**: não foi possível atender à solicitação por um objeto no cache e, portanto, o servidor a encaminhou ao servidor de origem e retornou o resultado ao visualizador.
- **LimitExceeded**: a solicitação foi negada porque uma cota do CloudFront (anteriormente conhecida como limite) foi excedida.
- **CapacityExceeded**: o servidor retornou um código de status HTTP 503 porque não tinha capacidade suficiente no momento da solicitação para fornecer o objeto.
- **Error**: normalmente, isso significa que a solicitação resultou em um erro de cliente (o valor do campo `sc-status` está no intervalo 4xx) ou em um erro de servidor (o valor do campo `sc-status` está no intervalo 5xx). Se o valor do campo `sc-status` for 200 ou se o valor desse campo for `Error` e o valor do campo `x-edge-response-result-type` não for `Error`, isso significa que a solicitação HTTP foi bem-sucedida, mas o cliente desconectou antes de receber todos os bytes.
- **Redirect**: o servidor redirecionou o visualizador de HTTP para HTTPS de acordo com as configurações de distribuição.

27.fle-encrypted-fields

O número de campos de [criptografia em nível de campo](#) que o servidor de borda criptografou e encaminhou para a origem. Os servidores do CloudFront fazem streaming da solicitação processada para a origem à medida que criptografam dados, portanto, esse campo pode ter um valor, mesmo que o valor de `fle-status` seja um erro.

28.fle-status

Quando a [criptografia em nível de campo](#) é configurada para uma distribuição, esse campo contém um código que indica se o corpo da solicitação foi processado com êxito. Quando o servidor processa o corpo da solicitação, criptografa os valores nos campos especificados e encaminha a solicitação para a origem com êxito, o valor desse campo é `Processed`. Nesse caso, o valor de `x-edge-result-type` pode indicar um erro do lado do cliente ou do lado do servidor.

Os valores possíveis para esse campo incluem:

- **ForwardedByContentType**: o servidor encaminhou a solicitação para a origem sem análise nem criptografia, pois nenhum tipo de conteúdo foi configurado.

- `ForwardedByQueryArgs`: o servidor encaminhou a solicitação para a origem sem análise nem criptografia, pois a solicitação contém um argumento de consulta que não foi configurado para a criptografia em nível de campo.
- `ForwardedDueToNoProfile`: o servidor encaminhou a solicitação para a origem sem análise nem criptografia, pois nenhum perfil foi especificado na configuração da criptografia em nível de campo.
- `MalformedContentTypeClientError`: o servidor rejeitou a solicitação e retornou o código de status HTTP 400 para o visualizador, pois o valor do cabeçalho `Content-Type` estava em um formato inválido.
- `MalformedInputClientError`: o servidor rejeitou a solicitação e retornou o código de status HTTP 400 para o visualizador, pois o corpo da solicitação estava em um formato inválido.
- `MalformedQueryArgsClientError`: o servidor rejeitou a solicitação e retornou o código de status HTTP 400 para o visualizador, pois um argumento de consulta estava vazio ou em um formato inválido.
- `RejectedByContentType`: o servidor rejeitou a solicitação e retornou o código de status HTTP 400 para o visualizador, pois nenhum tipo de conteúdo foi especificado na configuração para criptografia em nível de campo.
- `RejectedByQueryArgs`: o servidor rejeitou a solicitação e retornou o código de status HTTP 400 para o visualizador, pois nenhum argumento de consulta foi especificado na configuração para criptografia em nível de campo.
- `ServerError`: o servidor de origem retornou um erro.

Se a solicitação exceder uma cota de criptografia em nível de campo (anteriormente conhecida como limite), esse campo conterá um dos seguintes códigos de erro, e o servidor retornará o código de status HTTP 400 ao visualizador. Para obter uma lista das cotas atuais de criptografia no nível de campo, consulte [Cotas para criptografia no nível de campo](#).

- `FieldLengthLimitClientError`: um campo configurado para ser criptografado excedeu o tamanho máximo permitido.
- `FieldNumberLimitClientError`: uma solicitação que a distribuição está configurada para criptografar contém o número de campos maior do que o permitido.
- `RequestLengthLimitClientError`: o tamanho do corpo da solicitação excedeu o tamanho máximo permitido quando a criptografia no nível de campo foi configurada.

29sc-content-type

O valor do cabeçalho do HTTP `Content-Type` da resposta.

30 **sc-content-len**

O valor do cabeçalho do HTTP Content-Length da resposta.

31 **sc-range-start**

Quando a resposta contém o cabeçalho do HTTP Content-Range, esse campo contém o valor inicial do intervalo.

32 **sc-range-end**

Quando a resposta contém o cabeçalho do HTTP Content-Range, esse campo contém o valor final do intervalo.

33 **c-port**

O número da porta da solicitação do visualizador.

34 **x-edge-detailed-result-type**

Esse campo conterá o mesmo valor que o campo `x-edge-result-type`, exceto nos seguintes casos:

- Quando o objeto for enviado ao visualizador do cache da camada [Origin Shield](#), esse campo conterá `OriginShieldHit`.
- Quando o objeto não estiver no cache do CloudFront e a resposta for gerada por uma [função Lambda@Edge de solicitação de origem](#), esse campo conterá `MissGeneratedResponse`.
- Quando o valor do campo `x-edge-result-type` for `Error`, esse campo conterá um dos seguintes valores com mais informações sobre o erro:
 - `AbortedOrigin`: o servidor encontrou um problema com a origem.
 - `ClientCommError`: a resposta ao visualizador foi interrompida devido a um problema de comunicação entre o servidor e o visualizador.
 - `ClientGeoBlocked`: a distribuição é configurada para recusar solicitações da localização geográfica do visualizador.
 - `ClientHungUpRequest` – o visualizador parou prematuramente ao enviar a solicitação.
 - `Error`: ocorreu um erro para o qual o tipo de erro não se encaixa em nenhuma das outras categorias. Esse tipo de erro pode ocorrer quando o servidor fornece uma resposta de erro do cache.
 - `InvalidRequest`: o servidor recebeu uma solicitação inválida do visualizador.

- `InvalidRequestCertificate`: a distribuição não corresponde ao certificado SSL/TLS para o qual a conexão HTTPS foi estabelecida.
- `InvalidRequestHeader`: a solicitação continha um cabeçalho inválido.
- `InvalidRequestMethod` – a distribuição não está configurada para lidar com o método de solicitação HTTP que foi usado. Isso pode acontecer quando a distribuição oferece suporte somente a solicitações armazenáveis em cache.
- `OriginCommError`: a solicitação expirou durante a conexão à origem ou a leitura de dados da origem.
- `OriginConnectError`: o servidor não pôde se conectar à origem.
- `OriginContentRangeLengthError`: o cabeçalho `Content-Length` na resposta da origem não corresponde ao tamanho no cabeçalho `Content-Range`.
- `OriginDnsError`: o servidor não pôde resolver o nome de domínio da origem.
- `OriginError` – a origem retornou uma resposta incorreta.
- `OriginHeaderTooBigError`: um cabeçalho retornado pela origem é muito grande para o processamento pelo servidor de borda.
- `OriginInvalidResponseError` – a origem retornou uma resposta inválida.
- `OriginReadError`: o servidor não pôde ler na origem.
- `OriginWriteError`: o servidor não pôde gravar na origem.
- `OriginZeroSizeObjectError` – um objeto de tamanho zero enviado da origem resultou em um erro.
- `SlowReaderOriginError` – o visualizador ficou lento ao ler a mensagem que causou o erro de origem.

35.c-country

Um código de país que representa a localização geográfica do visualizador, conforme determinado pelo endereço IP do visualizador. Para obter uma lista de códigos de país, consulte [ISO 3166-1 alfa-2](#).

36.cs-accept-encoding

O valor do cabeçalho `Accept-Encoding` na solicitação do visualizador.

37.cs-accept

O valor do cabeçalho `Accept` na solicitação do visualizador.

38.cache-behavior-path-pattern

O padrão do caminho que identifica o comportamento de cache que correspondeu à solicitação do visualizador.

39.cs-headers

Os cabeçalhos HTTP (nomes e valores) na solicitação do visualizador.

Note

Este campo é truncado em 800 bytes.

40.cs-header-names

Os nomes dos cabeçalhos HTTP (não valores) na solicitação do visualizador.

Note

Este campo é truncado em 800 bytes.

41.cs-headers-count

O número de cabeçalhos HTTP na solicitação do visualizador.

42.origin-fbl

O número de segundos de latência de primeiro byte entre o CloudFront e a origem.

43.origin-lbl

O número de segundos de latência de último byte entre o CloudFront e a origem.

44.asn

O número de sistema autônomo (ASN) do visualizador.

45.primary-distribution-id

Quando a implantação contínua está habilitada, essa ID identifica qual distribuição é a principal na distribuição atual.

46.primary-distribution-dns-name

Quando a implantação contínua está habilitada, esse valor mostra o nome de domínio principal relacionado à distribuição atual do CloudFront (por exemplo, d111111abcdef8.cloudfront.net).

 Campos de CMCD em logs em tempo real

Para ter mais informações sobre esses campos, consulte o documento [Ecosistema em vídeo de aplicativo da web de especificação CTA: Common Media Client Data CTA-5004](#).

47.cmcd-encoded-bitrate

A taxa de bits codificada do objeto de áudio ou vídeo solicitado.

48.cmcd-buffer-length

O tamanho do buffer do objeto de mídia solicitado.

49.cmcd-buffer-starvation

Se o buffer ficou inativo em algum momento entre a solicitação anterior e a solicitação do objeto. Isso pode fazer com que o player fique em um estado de rearmazenamento em buffer, o que pode interromper a reprodução de vídeo ou áudio.

50.cmcd-content-id

Uma string exclusiva que identifica o conteúdo atual.

51.cmcd-object-duration

A duração da reprodução do objeto solicitado (em milissegundos).

52.cmcd-deadline

O prazo a partir do momento da solicitação em que a primeira amostra desse objeto deve estar disponível, para que um estado de buffer inoperante ou outros problemas de reprodução sejam evitados.

53.cmcd-measured-throughput

O throughput entre o cliente e o servidor, conforme medido pelo cliente.

54.cmcd-next-object-request

O caminho relativo do próximo objeto solicitado.

55.cmcd-next-range-request

Se a próxima solicitação for uma solicitação de objeto parcial, essa string indica o intervalo de bytes a ser solicitado.

56.**cmcd-object-type**

O tipo de mídia do objeto atual que está sendo solicitado.

57.**cmcd-playback-rate**

1 se estiver em tempo real, 2 se estiver em velocidade dupla, 0 se não estiver sendo reproduzido.

58.**cmcd-requested-maximum-throughput**

O throughput máximo solicitado que o cliente considera suficiente para a entrega de ativos.

59.**cmcd-streaming-format**

O formato de streaming que define a solicitação atual.

60.**cmcd-session-id**

Um GUID que identifica a sessão de reprodução atual.

61.**cmcd-stream-type**

Token que identifica a disponibilidade do segmento. v = todos os segmentos estão disponíveis. l = os segmentos ficam disponíveis ao longo do tempo.

62.**cmcd-startup**

A chave é incluída sem um valor se o objeto for necessário com urgência durante o startup, busca ou recuperação após um evento de esvaziamento do buffer.

63.**cmcd-top-bitrate**

A representação de taxa de bits mais alta que o cliente pode reproduzir.

64.**cmcd-version**

A versão dessa especificação usada para interpretar os nomes e chave-valor definidos. Se essa chave for omitida, o cliente e o servidor devem interpretar os valores como definidos pela versão 1.

Endpoint (Kinesis Data Streams)

O endpoint contém informações sobre o Kinesis Data Streams para o qual você quer enviar logs em tempo real. Forneça o nome de recurso da Amazon (ARN) do stream de dados.

Consulte mais informações sobre como criar um Kinesis Data Streams nos tópicos a seguir no Guia do desenvolvedor do Amazon Kinesis Data Streams.

- [Criar e gerenciar fluxos](#)
- [Executar operações básicas do Kinesis Data Streams usando a AWS CLI](#)
- [Criar um fluxo](#) (usa o AWS SDK for Java)

Ao criar um stream de dados, você precisa especificar o número de fragmentos. Use as seguintes informações para ajudá-lo a estimar o número de fragmentos necessários.

Como estimar o número de fragmentos para o stream de dados do Kinesis

1. Calcule (ou estime) o número de solicitações por segundo recebidas pela sua distribuição do CloudFront.

Você pode usar os [relatórios de uso do CloudFront](#) (no console do CloudFront) e as [métricas do CloudFront](#) (nos consoles do CloudFront e do Amazon CloudWatch) para ajudar você a calcular as solicitações por segundo.

2. Determine o tamanho típico de um único registro de log em tempo real.

Em geral, um único registro de log tem cerca de 500 bytes. Um grande registro que inclui todos os campos disponíveis costuma ter cerca de 1 KB.

Caso não tenha certeza do tamanho do registro de log, você poderá habilitar logs em tempo real com uma taxa de amostragem baixa (por exemplo, 1%), depois calcular o tamanho médio do registro usando dados de monitoramento no Kinesis Data Streams (total de bytes de entrada dividido pelo número total de registros).

3. Na [página de preços do Amazon Kinesis Data Streams](#), em AWS Pricing Calculator, selecione Crie já sua estimativa personalizada.
 - Na calculadora, insira o número de solicitações (registros) por segundo.
 - Insira o tamanho médio de um único registro de log.
 - Selecione Mostrar cálculos.

A calculadora de preços exibe o número de fragmentos de que você precisa e o custo estimado.

IAM role (Função do IAM)

A função do AWS Identity and Access Management (IAM) que concede ao CloudFront permissão para entregar logs em tempo real para o Kinesis Data Streams.

Ao criar uma configuração de log em tempo real com o console do CloudFront, é possível escolher **Create new service role** (Criar nova função de serviço) para permitir que o console crie a função do IAM para você.

Ao criar uma configuração de log em tempo real com o AWS CloudFormation ou a API do CloudFront (AWS CLI ou SDK), será necessário criar a função do IAM e fornecer o ARN da função. Para criar a função do IAM você mesmo, use as políticas a seguir.

Política de confiança da função do IAM

Para usar a seguinte política de confiança de função do IAM, substitua **111122223333** pelo número de sua Conta da AWS. O elemento `Condition` nessa política ajuda a evitar o [problema confused deputy](#) porque o CloudFront só pode assumir essa função em nome de uma distribuição em sua Conta da AWS.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        }
      }
    }
  ]
}
```

Política de permissões da função do IAM para um stream de dados não criptografado

Para usar a política a seguir, substitua **arn:aws:kinesis:us-east-2:123456789012:stream/StreamName** pelo ARN do seu stream de dados do Kinesis.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kinesis:DescribeStreamSummary",
        "kinesis:DescribeStream",
        "kinesis:PutRecord",
        "kinesis:PutRecords"
      ],
      "Resource": [
        "arn:aws:kinesis:us-east-2:123456789012:stream/StreamName"
      ]
    }
  ]
}
```

Política de permissões da função do IAM para um stream de dados criptografado

Para usar a política a seguir, substitua *arn:aws:kinesis:us-east-2:123456789012:stream/StreamName* pelo ARN do seu stream de dados do Kinesis e *arn:aws:kms:us-east-2:123456789012:key/e58a3d0b-fe4f-4047-a495-ae03cc73d486* pelo ARN da sua AWS KMS key.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kinesis:DescribeStreamSummary",
        "kinesis:DescribeStream",
        "kinesis:PutRecord",
        "kinesis:PutRecords"
      ],
      "Resource": [
        "arn:aws:kinesis:us-east-2:123456789012:stream/StreamName"
      ]
    },
    {
      "Effect": "Allow",
```

```
    "Action": [
      "kms:GenerateDataKey"
    ],
    "Resource": [
      "arn:aws:kms:us-east-2:123456789012:key/e58a3d0b-fe4f-4047-a495-ae03cc73d486"
    ]
  }
]
```

Criar um consumidor do Kinesis Data Streams

Para ler e analisar os logs em tempo real, crie ou use um consumidor do Kinesis Data Streams. Ao criar um consumidor para logs em tempo real do CloudFront, é importante saber que os campos em cada registro de log em tempo real são sempre entregues na mesma ordem, conforme listado na seção [Campos](#). Crie o consumidor para acomodar essa ordem fixa.

Por exemplo, considere uma configuração de log em tempo real que inclua apenas estes três campos: `time-to-first-byte`, `sc-status` e `c-country`. Nesse cenário, o último campo, `c-country`, é sempre o campo número 3 em cada registro de log. No entanto, se posteriormente você adicionar campos à configuração de log em tempo real, o posicionamento de cada campo em um registro pode ser alterado.

Por exemplo, se você adicionar os campos `sc-bytes` e `time-taken` à configuração de log em tempo real, esses campos serão inseridos em cada registro de log de acordo com a ordem mostrada na seção [Campos](#). A ordem resultante de todos os cinco campos é `time-to-first-byte`, `sc-status`, `sc-bytes`, `time-taken` e `c-country`. Originalmente, o campo `c-country` era o campo número 3, mas agora é o campo número 5. Verifique se o consumidor da aplicação pode manusear campos que mudam de posição em um registro de log, caso você adicione campos à configuração de log em tempo real.

Solução de problemas de logs em tempo real

Depois de criar uma configuração de log em tempo real, é possível descobrir que nenhum registro (ou nem todos os registros) são entregues ao Kinesis Data Streams. Nesse caso, primeiro verifique se a distribuição do CloudFront está recebendo solicitações do visualizador. Se estiver, você poderá verificar a seguinte configuração para continuar a solução de problemas.

Permissões de função do IAM

Para entregar registros de log em tempo real ao stream de dados do Kinesis, o CloudFront usa a função do IAM na configuração de log em tempo real. Verifique se a política de confiança da função e a política de permissões da função correspondem às políticas mostradas em [IAM role \(Função do IAM\)](#).

Limitação do Kinesis Data Streams

Se o CloudFront gravar registros de log em tempo real no stream de dados do Kinesis de forma mais rápida do que o stream pode processar, o Kinesis Data Streams poderá limitar as solicitações do CloudFront. Nesse caso, é possível aumentar o número de fragmentos no stream de dados do Kinesis. Cada fragmento pode oferecer suporte a gravações de até 1.000 registros por segundo, até um número máximo de gravações de 1 MB de dados por segundo.

Logs de funções de borda

É possível usar o Amazon CloudWatch Logs para obter logs para as funções de borda, tanto do [Lambda@Edge](#) quanto do [CloudFront Functions](#). É possível acessar os logs usando o console do CloudWatch ou a API do CloudWatch Logs.

Important

Recomendamos que você use os logs para compreender a natureza das solicitações do seu conteúdo, não como uma contabilidade completa de todas as solicitações. O CloudFront entrega logs de função de borda com base no melhor esforço. A entrada do log de uma solicitação específica pode ser entregue muito depois do processamento da solicitação e, raramente, nunca ser entregue. Quando uma entrada de log for omitida dos logs de função de borda, o número de entradas nos logs não corresponderá ao uso exibido nos relatórios de uso e faturamento da AWS.

Tópicos

- [Logs do Lambda@Edge](#)
- [Logs do CloudFront Functions](#)

Logs do Lambda@Edge

O Lambda@Edge envia logs de função para o CloudWatch Logs automaticamente, criando fluxos de logs nas Regiões da AWS onde as funções são executadas. O nome do grupo de logs é formatado como `/aws/lambda/us-east-1.function-name`, em que *function-name* é o nome que você deu à função ao criá-la e `us-east-1` é o código da Região da AWS na qual a função foi criada. O nome do grupo de logs sempre contém `us-east-1`, mesmo para grupos de outras regiões nas quais a função é executada.

Note

O Lambda@Edge controla os logs com base no volume da solicitação e no tamanho dos logs.

Você deve analisar os arquivos de log do CloudWatch na Região da AWS correta para ver os arquivos de log da função do Lambda@Edge. Para ver as regiões onde a função do Lambda@Edge está em execução, visualize os grafos das métricas da função no console do CloudFront. As métricas são exibidas para cada Região da AWS. Na mesma página, é possível selecionar uma região e, depois, visualizar os arquivos de log referentes a ela, para investigar problemas.

Para saber mais sobre como usar o CloudWatch Logs com funções do Lambda@Edge, consulte o seguinte:

- Para obter mais informações sobre como visualizar gráficos na seção Monitoring (Monitoramento) no console do CloudFront, consulte [the section called “Monitorar métricas do CloudFront com o Amazon CloudWatch”](#).
- Para informações sobre as permissões necessárias para enviar dados para o CloudWatch Logs, consulte [the section called “Configurar permissões e perfis do IAM”](#).
- Para obter informações sobre como adicionar registro em log a uma função do Lambda@Edge, consulte [Registro em log da função do AWS Lambda em Node.js](#) ou [Registro em log da função do AWS Lambda em Python](#) no Guia do desenvolvedor do AWS Lambda.
- Para obter informações sobre as cotas do CloudWatch Logs (anteriormente conhecidas como limites), consulte [Cotas do CloudWatch Logs](#) no Guia do usuário do Amazon CloudWatch Logs.

Logs do CloudFront Functions

Se o código de uma função do CloudFront contiver instruções `console.log()`, o CloudFront Functions enviará automaticamente essas linhas de log para o CloudWatch Logs. Se não houver instruções `console.log()`, nada será enviado para o CloudWatch Logs.

O CloudFront Functions sempre cria streams de log na região Leste dos EUA (Norte da Virgínia) (`us-east-1`), independentemente de qual local da borda executou a função. O nome do grupo de logs está no formato `/aws/cloudfront/function/FunctionName`, em que *FunctionName* é o nome que você deu à função quando a criou. O nome do stream de log está no formato `YYYY/M/D/UUID`.

Abaixo, é possível visualizar uma mensagem de log de exemplo enviada ao CloudWatch Logs. Cada linha começa com um ID que identifica exclusivamente uma solicitação do CloudFront. A mensagem começa com uma linha `START` que inclui o ID de distribuição do CloudFront e termina com uma linha `END`. Entre as linhas `START` e `END` estão as linhas de log geradas pelas instruções `console.log()` na função.

```
U7b4hR_RaxMADupvKAvr8_m9gsGXvioUggLV50yq-vmAtH8HADpjhw== START DistributionID:
E3E5D42GADAXZZ
U7b4hR_RaxMADupvKAvr8_m9gsGXvioUggLV50yq-vmAtH8HADpjhw== Example function log output
U7b4hR_RaxMADupvKAvr8_m9gsGXvioUggLV50yq-vmAtH8HADpjhw== END
```

Note

O CloudFront Functions envia logs ao CloudWatch apenas para funções na fase `LIVE` que são executadas em resposta a solicitações e respostas de produção. Quando você [testa uma função](#), o CloudFront não envia métricas nem logs para o CloudWatch. A saída de teste contém informações sobre erros, utilização de computação e logs de funções (instruções `console.log()`), mas essas informações não são enviadas para o CloudWatch.

O CloudFront Functions usa uma [função vinculada ao serviço](#) do AWS Identity and Access Management (IAM) para enviar logs para o CloudWatch Logs em sua conta. Uma função vinculada a serviço é uma função do IAM que está vinculada diretamente a um produto da AWS. As funções vinculadas a serviços são predefinidas pelo serviço e incluem todas as permissões de que ele precisa para chamar outros serviços da AWS em seu nome. O CloudFront Functions usa uma função vinculada ao serviço chamada `AWSServiceRoleForCloudFrontLogger`. Para obter mais

informações sobre essa função, consulte [the section called “Funções vinculadas ao serviço para o Lambda@Edge”](#) (O Lambda@Edge usa a mesma função vinculada ao serviço).

Quando uma função falha com um erro de validação ou um erro de execução, as informações são registradas nos [logs padrão](#) e nos [logs em tempo real](#) do CloudFront. As informações sobre o erro são registradas nos campos `x-edge-result-type`, `x-edge-response-result-type` e `x-edge-detailed-result-type`.

Registrar em log chamadas de API do Amazon CloudFront usando o AWS CloudTrail

O CloudFront é integrado ao [AWS CloudTrail](#), um serviço que fornece um registro das ações realizadas por um usuário, um perfil ou um AWS service (Serviço da AWS). O CloudTrail captura todas as chamadas de API para o CloudFront como eventos. As chamadas capturadas incluem as chamadas do console do CloudFront e as chamadas de código para as operações de API do CloudFront. Ao fazer uso das informações coletadas pelo CloudTrail, é possível determinar a solicitação feita ao CloudFront, o endereço IP no qual a solicitação foi feita, quando a solicitação foi feita e detalhes adicionais.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar:

- Se a solicitação foi feita com credenciais de usuário raiz ou credenciais de usuário.
- Se a solicitação foi feita em nome de um usuário do Centro de Identidade do IAM.
- Se a solicitação foi feita com credenciais de segurança temporárias de uma função ou de um usuário federado.
- Se a solicitação foi feita por outro AWS service (Serviço da AWS).

O CloudTrail está ativo em sua Conta da AWS e você tem acesso automático ao Histórico de eventos do CloudTrail. O Histórico de eventos do CloudTrail fornece um registro visualizável, pesquisável, baixável e imutável dos últimos 90 dias de eventos de gerenciamento gravados em uma Região da AWS. Para obter mais informações, consulte [Trabalhar com histórico de eventos do CloudTrail](#) no Guia do usuário do AWS CloudTrail. Não há cobranças do CloudTrail pela visualização do Histórico de eventos.

Para obter um registro contínuo de eventos em sua Conta da AWS nos últimos 90 dias, crie uma trilha ou um armazenamento de dados de eventos do [CloudTrail Lake](#).

Trilhas do CloudTrail

Uma trilha permite que o CloudTrail entregue arquivos de log a um bucket do Amazon S3. As trilhas criadas usando o AWS Management Console são de várias regiões. Só é possível criar uma trilha de região única ou de várias regiões usando a AWS CLI. Criar uma trilha de várias regiões é uma prática recomendada, pois você captura atividades em todas as Regiões da AWS da conta. Se você criar uma trilha de região única, poderá visualizar somente os eventos registrados na Região da AWS da trilha. Para obter mais informações sobre trilhas, consulte [Criar uma trilha para a Conta da AWS](#) e [Criar uma trilha para uma organização](#) no Guia do usuário do AWS CloudTrail.

Uma cópia dos seus eventos de gerenciamento em andamento pode ser entregue no bucket do Amazon S3 sem nenhum custo via CloudTrail com a criação de uma trilha. No entanto, há cobranças de armazenamento do Amazon S3. Para obter mais informações sobre os preços do CloudTrail, consulte [Preços do AWS CloudTrail](#). Para receber informações sobre a definição de preço do Amazon S3, consulte [Definição de preço do Amazon S3](#).

Armazenamentos de dados de eventos do CloudTrail Lake

O CloudTrail Lake permite executar consultas baseadas em SQL em seus eventos. O CloudTrail Lake converte eventos existentes em formato JSON baseado em linhas para o formato [Apache ORC](#). O ORC é um formato colunar de armazenamento otimizado para recuperação rápida de dados. Os eventos são agregados em armazenamentos de dados de eventos, que são coleções imutáveis de eventos baseados nos critérios selecionados com a aplicação de [seletores de eventos avançados](#). Os seletores que você aplica a um armazenamento de dados de eventos controlam quais eventos persistem e estão disponíveis para você consultar. Para obter mais informações sobre o CloudTrail Lake, consulte [Trabalhar com o AWS CloudTrail Lake](#), no Guia do usuário do AWS CloudTrail.

Os armazenamentos de dados de eventos e consultas do CloudTrail Lake incorrem em custos. Ao criar um armazenamento de dados de eventos, você escolhe a [opção de preço](#) que deseja usar para ele. A opção de preço determina o custo para a ingestão e para o armazenamento de eventos, e o período de retenção padrão e máximo para o armazenamento de dados de eventos. Para obter mais informações sobre os preços do CloudTrail, consulte [Preços do AWS CloudTrail](#).

Note

O CloudFront é um serviço global. O CloudTrail registra eventos do CloudFront na região Leste dos EUA (Norte da Virgínia). Para ter mais informações, consulte [Eventos de serviços globais](#) no Guia do usuário do AWS CloudTrail.

Se você usar credenciais de segurança temporárias usando o AWS Security Token Service, as chamadas para endpoints regionais, como `us-west-2`, serão registradas no CloudTrail em sua região apropriada.

Para ter mais informações sobre endpoints do Amazon CloudFront, consulte [Endpoints e cotas do CloudFront](#) na Referência geral da AWS.

Eventos de dados do CloudFront no CloudTrail

Os [eventos de dados](#) fornecem informações sobre as operações de recursos realizadas em um recurso (por exemplo, leitura ou gravação em uma distribuição do CloudFront). Elas também são conhecidas como operações de plano de dados. Eventos de dados geralmente são atividades de alto volume. Por padrão, o CloudTrail não registra eventos de dados em log. O Histórico de eventos do CloudTrail não registra eventos de dados.

Há cobranças adicionais para eventos de dados. Para obter mais informações sobre os preços do CloudTrail, consulte [Preços do AWS CloudTrail](#).

É possível registrar logs de eventos de dados para os tipos de recursos do CloudFront usando o console do CloudTrail, a AWS CLI ou operações da API do CloudTrail. Para obter mais informações sobre como registrar eventos de dados em log, consulte [Registrar eventos de dados com o AWS Management Console](#) e [Registrar eventos de dados com a AWS Command Line Interface](#) no Guia do usuário do AWS CloudTrail.

A tabela a seguir lista o tipo de recurso do CloudFront para o qual você pode registrar eventos de dados em log. A coluna Tipo de evento de dados (console) mostra o valor a ser escolhido na lista Tipo de evento de dados no console do CloudTrail. A coluna do valor `resources.type` mostra o valor de `resources.type` que você especificaria ao configurar seletores de eventos avançados usando a AWS CLI ou as APIs do CloudTrail. A coluna APIs de dados registradas no CloudTrail mostra as chamadas de API registradas no CloudTrail para o tipo de recurso.

Tipo de evento de dados (console)	valor resources.type	APIs de dados registradas no CloudTrail
CloudFront KeyValueStore	AWS::CloudFront::KeyValueStore	<ul style="list-style-type: none"> • DeleteKeys • DescribeKeyValueStore • GetKey • ListKeys • PutKeys • UpdateKeys

É possível configurar seletores de eventos avançados para filtrar os campos `eventName`, `readOnly` e `resources.ARN` para registrar em log somente os eventos que são importantes para você. Para obter mais informações sobre esses campos, consulte [AdvancedFieldSelector](#) na Referência de API do AWS CloudTrail.

Eventos de gerenciamento do CloudFront no CloudTrail

Os [Eventos de gerenciamento](#) fornecem informações sobre operações de gerenciamento executadas em recursos na sua Conta da AWS. Elas também são conhecidas como operações de plano de controle. Por padrão, o CloudTrail registra eventos de gerenciamento em logs.

O Amazon CloudFront registra em log todas as operações do ambiente de gerenciamento do CloudFront como eventos de gerenciamento. Para ter uma lista das operações do ambiente de gerenciamento do Amazon CloudFront que o CloudFront registra em log no CloudTrail, consulte a [Referência da API do Amazon CloudFront](#).

Exemplos de eventos do CloudFront

Um evento representa uma única solicitação de qualquer origem e inclui informações sobre a operação solicitada, a data e a hora da operação da API, os parâmetros de solicitação etc. Os arquivos de log do CloudTrail não são um rastreamento de pilha ordenada de chamadas de API pública, portanto não são exibidos em uma ordem específica.

Sumário

- [Exemplo: UpdateDistribution](#)
- [Exemplo: UpdateKeys](#)

Exemplo: UpdateDistribution

O exemplo a seguir mostra um evento do CloudTrail que demonstra a operação [UpdateDistribution](#).

Para chamadas para a API do CloudFront, eventSource é `cloudfront.amazonaws.com`.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE:role-session-name",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/role-session-name",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Admin",
        "accountId": "111122223333",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2024-02-02T19:23:50Z",
        "mfaAuthenticated": "false"
      }
    }
  },
  "eventTime": "2024-02-02T19:26:01Z",
  "eventSource": "cloudfront.amazonaws.com",
  "eventName": "UpdateDistribution",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "52.94.133.137",
  "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
  like Gecko) Chrome/121.0.0.0 Safari/537.36",
  "requestParameters": {
    "distributionConfig": {
      "defaultRootObject": "",
      "aliases": {
        "quantity": 3,
        "items": [
          "alejandro_rosalez.awsps.myinstance.com",
          "cross-testing.alejandro_rosalez.awsps.myinstance.com",
```

```
        "*.alejandro_rosalez.awsps.myinstance.com"
    ]
},
"cacheBehaviors": {
    "quantity": 0,
    "items": []
},
"httpVersion": "http2and3",
"originGroups": {
    "quantity": 0,
    "items": []
},
"viewerCertificate": {
    "minimumProtocolVersion": "TLSv1.2_2021",
    "cloudFrontDefaultCertificate": false,
    "aCMCertificateArn": "arn:aws:acm:us-east-1:111122223333:certificate/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "sLSupportMethod": "sni-only"
},
"webACLId": "arn:aws:wafv2:us-east-1:111122223333:global/webacl/testing-
acl/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
"customErrorResponses": {
    "quantity": 0,
    "items": []
},
"logging": {
    "includeCookies": false,
    "prefix": "",
    "enabled": false,
    "bucket": ""
},
"priceClass": "PriceClass_All",
"restrictions": {
    "geoRestriction": {
        "restrictionType": "none",
        "quantity": 0,
        "items": []
    }
},
"isIPV6Enabled": true,
"callerReference": "1578329170895",
"continuousDeploymentPolicyId": "",
"enabled": true,
"defaultCacheBehavior": {
```

```
"targetOriginId": "d111111abcdef8",
"minTTL": 0,
"compress": false,
"maxTTL": 31536000,
"functionAssociations": {
  "quantity": 0,
  "items": []
},
"trustedKeyGroups": {
  "quantity": 0,
  "items": [],
  "enabled": false
},
"smoothStreaming": false,
"fieldLevelEncryptionId": "",
"defaultTTL": 86400,
"lambdaFunctionAssociations": {
  "quantity": 0,
  "items": []
},
"viewerProtocolPolicy": "redirect-to-https",
"forwardedValues": {
  "cookies": {"forward": "none"},
  "queryStringCacheKeys": {
    "quantity": 0,
    "items": []
  },
  "queryString": false,
  "headers": {
    "quantity": 1,
    "items": ["*"]
  }
},
"trustedSigners": {
  "items": [],
  "enabled": false,
  "quantity": 0
},
"allowedMethods": {
  "quantity": 2,
  "items": [
    "HEAD",
    "GET"
  ],
}
```

```
        "cachedMethods": {
            "quantity": 2,
            "items": [
                "HEAD",
                "GET"
            ]
        }
    },
    "staging": false,
    "origins": {
        "quantity": 1,
        "items": [
            {
                "originPath": "",
                "connectionTimeout": 10,
                "customOriginConfig": {
                    "originReadTimeout": 30,
                    "httpSPort": 443,
                    "originProtocolPolicy": "https-only",
                    "originKeepaliveTimeout": 5,
                    "httpPort": 80,
                    "originSslProtocols": {
                        "quantity": 3,
                        "items": [
                            "TLSv1",
                            "TLSv1.1",
                            "TLSv1.2"
                        ]
                    }
                },
                "id": "d111111abcdef8",
                "domainName": "d111111abcdef8.cloudfront.net",
                "connectionAttempts": 3,
                "customHeaders": {
                    "quantity": 0,
                    "items": []
                },
                "originShield": {"enabled": false},
                "originAccessControlId": ""
            }
        ]
    },
    "comment": "HIDDEN_DUE_TO_SECURITY_REASONS"
```



```
    },
    "id": "EDFDVBD6EXAMPLE",
    "ifMatch": "E1RTLUR9YES760"
  },
  "responseElements": {
    "distribution": {
      "activeTrustedSigners": {
        "quantity": 0,
        "enabled": false
      },
      "id": "EDFDVBD6EXAMPLE",
      "domainName": "d111111abcdef8.cloudfront.net",
      "distributionConfig": {
        "defaultRootObject": "",
        "aliases": {
          "quantity": 3,
          "items": [
            "alejandro_rosalez.awsps.myinstance.com",
            "cross-testing.alejandro_rosalez.awsps.myinstance.com",
            "*.alejandro_rosalez.awsps.myinstance.com"
          ]
        },
        "cacheBehaviors": {"quantity": 0},
        "httpVersion": "http2and3",
        "originGroups": {"quantity": 0},
        "viewerCertificate": {
          "minimumProtocolVersion": "TLSv1.2_2021",
          "cloudFrontDefaultCertificate": false,
          "aCMCertificateArn": "arn:aws:acm:us-east-1:111122223333:certificate/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
          "sSLSupportMethod": "sni-only",
          "certificateSource": "acm",
          "certificate": "arn:aws:acm:us-east-1:111122223333:certificate/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
        },
        "webACLId": "arn:aws:wafv2:us-east-1:111122223333:global/webacl/testing-acl/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "customErrorResponses": {"quantity": 0},
        "logging": {
          "includeCookies": false,
          "prefix": "",
          "enabled": false,
          "bucket": ""
        }
      }
    },
  },
}
```

```
"priceClass": "PriceClass_All",
"restrictions": {
  "geoRestriction": {
    "restrictionType": "none",
    "quantity": 0
  }
},
"isIPV6Enabled": true,
"callerReference": "1578329170895",
"continuousDeploymentPolicyId": "",
"enabled": true,
"defaultCacheBehavior": {
  "targetOriginId": "d1111111abcdef8",
  "minTTL": 0,
  "compress": false,
  "maxTTL": 31536000,
  "functionAssociations": {"quantity": 0},
  "trustedKeyGroups": {
    "quantity": 0,
    "enabled": false
  },
  "smoothStreaming": false,
  "fieldLevelEncryptionId": "",
  "defaultTTL": 86400,
  "lambdaFunctionAssociations": {"quantity": 0},
  "viewerProtocolPolicy": "redirect-to-https",
  "forwardedValues": {
    "cookies": {"forward": "none"},
    "queryStringCacheKeys": {"quantity": 0},
    "queryString": false,
    "headers": {
      "quantity": 1,
      "items": ["*"]
    }
  },
  "trustedSigners": {
    "enabled": false,
    "quantity": 0
  },
  "allowedMethods": {
    "quantity": 2,
    "items": [
      "HEAD",
      "GET"
    ]
  }
}
```

```
    ],
    "cachedMethods": {
      "quantity": 2,
      "items": [
        "HEAD",
        "GET"
      ]
    }
  },
  "staging": false,
  "origins": {
    "quantity": 1,
    "items": [
      {
        "originPath": "",
        "connectionTimeout": 10,
        "customOriginConfig": {
          "originReadTimeout": 30,
          "hTTPSPort": 443,
          "originProtocolPolicy": "https-only",
          "originKeepaliveTimeout": 5,
          "hTTPPort": 80,
          "originSslProtocols": {
            "quantity": 3,
            "items": [
              "TLSv1",
              "TLSv1.1",
              "TLSv1.2"
            ]
          }
        }
      },
      {
        "id": "d1111111abcdef8",
        "domainName": "d1111111abcdef8.cloudfront.net",
        "connectionAttempts": 3,
        "customHeaders": {"quantity": 0},
        "originShield": {"enabled": false},
        "originAccessControlId": ""
      }
    ]
  },
  "comment": "HIDDEN_DUE_TO_SECURITY_REASONS"
},
"aliasICPRecordals": [
```

```

        {
            "cNAME": "alejandro_rosalez.awsps.myinstance.com",
            "iCPRecordalStatus": "APPROVED"
        },
        {
            "cNAME": "cross-testing.alejandro_rosalez.awsps.myinstance.com",
            "iCPRecordalStatus": "APPROVED"
        },
        {
            "cNAME": "*.alejandro_rosalez.awsps.myinstance.com",
            "iCPRecordalStatus": "APPROVED"
        }
    ],
    "aRN": "arn:aws:cloudfront::111122223333:distribution/EDFDVBD6EXAMPLE",
    "status": "InProgress",
    "lastModifiedTime": "Feb 2, 2024 7:26:01 PM",
    "activeTrustedKeyGroups": {
        "enabled": false,
        "quantity": 0
    },
    "inProgressInvalidationBatches": 0
},
    "eTag": "E1YHBLAB2BJY1G"
},
    "requestID": "4e6b66f9-d548-11e3-a8a9-73e33example",
    "eventID": "5ab02562-0fc5-43d0-b7b6-90293example",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "apiVersion": "2020_05_31",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_128_GCM_SHA256",
        "clientProvidedHostHeader": "cloudfront.amazonaws.com"
    },
    "sessionCredentialFromConsole": "true"
}

```

Exemplo: UpdateKeys

O exemplo a seguir mostra um evento do CloudTrail que demonstra a operação [UpdateKeys](#).

Para chamadas para a API do KeyValueCollection do CloudFront, eventSource é edgekeyvaluestore.amazonaws.com em vez de cloudfront.amazonaws.com.

```
{
  "eventVersion": "1.09",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE:role-session-name",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/role-session-name",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Admin",
        "accountId": "111122223333",
        "userName": "Admin"
      },
      "attributes": {
        "creationDate": "2023-11-01T23:41:14Z",
        "mfaAuthenticated": "false"
      }
    }
  },
  "eventTime": "2023-11-01T23:41:28Z",
  "eventSource": "edgekeyvaluestore.amazonaws.com",
  "eventName": "UpdateKeys",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "3.235.183.252",
  "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
  like Gecko) Chrome/121.0.0.0 Safari/537.36",
  "requestParameters": {
    "kvsARN": "arn:aws:cloudfront::111122223333:key-value-store/a1b2c3d4-5678-90ab-
    cdef-EXAMPLE11111",
    "ifMatch": "KV306B1CX531EBP",
    "deletes": [
      {"key": "key1"}
    ]
  },
  "responseElements": {
    "itemCount": 0,
    "totalSizeInBytes": 0,
  }
}
```

```
    "eTag": "KVDC9VEVZ71ZG0"
  },
  "requestID": "5ccf104c-acce-4ea1-b7fc-73e33example",
  "eventID": "a0b1b5c7-906c-439d-9925-90293example",
  "readOnly": false,
  "resources": [
    {
      "accountId": "111122223333",
      "type": "AWS::CloudFront::KeyValueStore",
      "ARN": "arn:aws:cloudfront::111122223333:key-value-store/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": false,
  "recipientAccountId": "111122223333",
  "eventCategory": "Data",
  "tlsDetails": {
    "tlsVersion": "TLSv1.3",
    "cipherSuite": "TLS_AES_128_GCM_SHA256",
    "clientProvidedHostHeader": "111122223333.cloudfront-kvs.global.api.aws"
  }
}
```

Para obter informações sobre o conteúdo dos registros do CloudTrail, consulte [Conteúdo dos registros do CloudTrail](#) no Guia do usuário do AWS CloudTrail.

Monitorar as alterações de configuração com o AWS Config

Para registrar e avaliar as configurações dos recursos da AWS, é possível usar o AWS Config, que oferece uma visão detalhada da configuração das distribuições. Isso inclui como os recursos estão relacionados entre si e como eles foram configurados no passado, para que você possa analisar as alterações no decorrer do tempo.

Também é possível usar o AWS Config para registrar alterações nas configurações de distribuição do CloudFront. É possível capturar alterações em estados de distribuição, classes de preço, origens, configurações de restrição geográfica e configurações do Lambda@Edge.

Note

O AWS Config não registra tags de chave-valor para distribuições de streaming do CloudFront.

Sumário

- [Configurar o AWS Config com o CloudFront](#)
- [Visualizar o histórico de configuração do CloudFront](#)
- [Avaliar as configurações do CloudFront com o AWS Config Rules](#)

Configurar o AWS Config com o CloudFront

Ao configurar o AWS Config, é possível optar por gravar todos os recursos da AWS compatíveis ou somente os recursos especificados, como gravar apenas as alterações do CloudFront. Consulte uma lista de recursos compatíveis do CloudFront na seção [Amazon CloudFront](#) do tópico “Supported Resource Types” no Guia do desenvolvedor do AWS Config.

Observações

- Para monitorar as alterações de configuração na distribuição do CloudFront, faça login no console do CloudFront na Região da AWS Leste dos EUA (Norte da Virgínia).
- Pode haver um atraso na gravação de recursos com o AWS Config. O AWS Config registra recursos somente depois que descobre os recursos.

Console

Como configurar o AWS Config com o CloudFront

1. Faça login no AWS Management Console e abra o console do AWS Config em <https://console.aws.amazon.com/config/>.
2. Escolha Get Started Now.
3. Na página Settings, para Resource types to record, especifique os tipos de recursos da AWS que você deseja que o AWS Config registre. Para registrar somente mudanças

do CloudFront, escolha *Specific types* (Tipos específicos) e, em CloudFront, escolha a distribuição ou a distribuição em streaming da qual você quer acompanhar as alterações.

Para adicionar ou alterar as distribuições a serem acompanhadas, escolha *Settings* à esquerda, depois de concluir sua configuração inicial.

4. Especifique as opções adicionais necessárias para o AWS Config. Configure uma notificação, especifique um local para as informações de configuração e adicione regras para avaliar os tipos de recursos.

Para obter mais informações, consulte [Configuração do AWS Config com o console](#) no Guia do desenvolvedor do AWS Config.

AWS CLI

Para configurar o AWS Config com o CloudFront usando a AWS CLI, consulte [Setting up AWS Config with the AWS CLI](#) no Guia do desenvolvedor do AWS Config.

AWS Config API

Para configurar o AWS Config com o CloudFront usando a API do AWS Config consulte a operação de API [StartConfigurationRecorder](#) na Referência de API do AWS Config.

Visualizar o histórico de configuração do CloudFront

Depois que o AWS Config começa a gravar as alterações de configuração nas suas distribuições, é possível exibir o histórico de configuração de qualquer distribuição que você tenha configurado para o CloudFront.

É possível visualizar os históricos de configuração das maneiras indicadas a seguir.

Console

Para cada recurso gravado, é possível visualizar uma página de linha do tempo que fornece o histórico com detalhes da configuração. Para visualizar essa página, escolha o ícone cinza na coluna *Config Timeline* (Configurar linha de tempo) da página *Hosts* dedicados.

Para obter mais informações, consulte [Visualização de detalhes de configuração do console do AWS Config](#) no Guia do desenvolvedor do AWS Config.

AWS CLI

Consulte uma lista de todas as distribuições executando o comando [list-discovered-resources](#), conforme mostrado no exemplo a seguir.

```
aws configservice list-discovered-resources --resource-type
AWS::CloudFront::Distribution
```

Para ver os detalhes da configuração da distribuição de um intervalo de tempo específico, execute o comando [get-resource-config-history](#).

Para obter mais informações, consulte [Visualização de detalhes de configuração usando a CLI](#) no Guia do desenvolvedor do AWS Config.

AWS Config API

Para consultar uma lista de todas as suas distribuições, use a operação de API [ListDiscoveredResources](#).

Para consultar os detalhes de configuração de uma distribuição para um intervalo de tempo específico, use a operação de API [GetResourceConfigHistory](#). Para obter mais informações, consulte a [AWS ConfigReferência da API do](#) .

Avaliar as configurações do CloudFront com o AWS Config Rules

É possível avaliar as configurações em relação às configurações desejadas com o AWS Config Rules. Por exemplo, o AWS Config Rules ajuda você a avaliar se os recursos do CloudFront estão em conformidade com as práticas recomendadas comuns de segurança. É possível escolher regras gerenciadas, como política do visualizador HTTPS, SNI habilitada, OAC habilitado, failover de origem habilitado, ACL da web do AWS WAF ou políticas de recursos do AWS Shield Advanced a serem acionadas quando a configuração é alterada.

As regras gerenciadas podem realizar avaliações periodicamente, na frequência escolhida. O AWS Firewall Manager depende do AWS Config para alertas e correções automáticas. Para ter mais informações, consulte [Evaluating Resources with AWS Config Rules](#) e [List of AWS Config Managed Rules](#) no Guia do desenvolvedor do AWS Config.

Segurança no Amazon CloudFront

A segurança para com a nuvem na AWS é a nossa maior prioridade. Como cliente da AWS, você contará com um data center e uma arquitetura de rede criados para atender aos requisitos das organizações com as maiores exigências de segurança.

A segurança é uma responsabilidade compartilhada entre a AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isto como segurança da nuvem e segurança na nuvem:

- Segurança da nuvem: a AWS é responsável pela proteção da infraestrutura que executa produtos da AWS na Nuvem AWS. A AWS também fornece serviços que podem ser usados com segurança. Auditores de terceiros testam e verificam regularmente a eficácia da nossa segurança como parte dos [programas de conformidade da AWS](#). Para saber mais sobre os programas de conformidade que se aplicam ao Amazon CloudFront, consulte [Escopo dos produtos da AWS por programa de conformidade](#).
- Segurança da nuvem: sua responsabilidade é determinada pelo serviço da AWS que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da sua organização e as leis e regulamentos aplicáveis.

Esta documentação ajuda a entender como aplicar o modelo de responsabilidade compartilhada ao usar o CloudFront. Os tópicos a seguir mostram como configurar o CloudFront para atender aos seus objetivos de segurança e de conformidade. Você também saberá mais sobre como usar outros produtos da AWS que ajudam a monitorar e proteger os recursos do CloudFront.

Tópicos

- [Proteção de dados no Amazon CloudFront](#)
- [Identity and Access Management para Amazon CloudFront](#)
- [Registro em log e monitoramento no Amazon CloudFront](#)
- [Validação de conformidade com o Amazon CloudFront](#)
- [Resiliência no Amazon CloudFront](#)
- [Segurança da infraestrutura no Amazon CloudFront](#)

Proteção de dados no Amazon CloudFront

O modelo de [responsabilidade compartilhada](#) da AWS se aplica à proteção de dados no Amazon CloudFront. Conforme descrito nesse modelo, AWS é responsável por proteger a infraestrutura global que executa todas as Nuvem AWS. Você é responsável por manter o controle sobre seu conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que usa. Para obter mais informações sobre a privacidade de dados, consulte as [Perguntas Frequentes sobre Privacidade de Dados](#). Para obter mais informações sobre a proteção de dados na Europa, consulte a postagem do blog [AWS Shared Responsibility Model and GDPR](#) no Blog de segurança da AWS.

Para fins de proteção de dados, recomendamos que você proteja as credenciais da Conta da AWS e configure as contas de usuário individuais com AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com os recursos da AWS. Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure a API e atividade do usuário logando com AWS CloudTrail. Para obter mais informações sobre como usar as trilhas do CloudTrail para capturar atividades da AWS, consulte [Working with CloudTrail trails](#) no Guia do Usuário do AWS CloudTrail.
- Use as soluções de criptografia AWS, juntamente com todos os controles de segurança padrão em Serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sigilosos armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-3 ao acessar a AWS por meio de uma interface de linha de comando ou uma API, use um endpoint do FIPS. Para obter mais informações sobre os endpoints FIPS disponíveis, consulte [Federal Information Processing Standard \(FIPS\) 140-3](#).

É altamente recomendável que nunca sejam colocadas informações de identificação confidenciais, como endereços de e-mail dos seus clientes, em marcações ou campos de formato livre, como um campo Nome. Isso inclui o trabalho com o CloudFront ou outros Serviços da AWS que usem o console, a API, a AWS CLI ou os AWS SDKs. Quaisquer dados inseridos em tags ou campos

de texto de formato livre usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, recomendamos fortemente que não sejam incluídas informações de credenciais no URL para validar a solicitação a esse servidor.

O Amazon CloudFront oferece várias opções que você pode usar para ajudar a proteger o conteúdo que fornece:

- Configure conexões HTTPS.
- Configure a criptografia em nível de campo para fornecer segurança adicional para dados específicos durante o trânsito.
- Restrinja o acesso ao conteúdo para que apenas determinadas pessoas, ou aquelas de uma área específica, possam visualizá-lo.

Os tópicos a seguir detalham mais as opções.

Tópicos

- [Criptografia em trânsito](#)
- [Criptografia em repouso](#)
- [Restringir o acesso ao conteúdo](#)

Criptografia em trânsito

Para criptografar os dados durante o trânsito, configure o Amazon CloudFront para exigir que os visualizadores usem HTTPS para solicitar seus arquivos, a fim de que as conexões sejam criptografadas quando o CloudFront se comunicar com os visualizadores. Também é possível configurar o CloudFront para usar HTTPS e obter arquivos da origem, a fim de que as conexões sejam criptografadas quando o CloudFront se comunicar com os usuários.

Para obter mais informações, consulte [Usar HTTPS com o CloudFront](#).

A criptografia no nível de campo acrescenta uma camada adicional de segurança juntamente com o HTTPS, o que permite a você proteger dados específicos em todo o processamento do sistema, de modo que apenas alguns aplicativos possam vê-los. Ao configurar a criptografia em nível de campo no CloudFront, você pode fazer upload com segurança das informações confidenciais enviadas pelo usuário aos seus servidores Web. As informações confidenciais fornecidas pelos seus clientes são criptografadas no ponto mais próximo ao usuário. Elas permanecem criptografadas em toda a

pilha de aplicações, garantindo que apenas as aplicações que precisam dos dados e possuem as credenciais para descriptografá-los, poderão fazê-lo.

Para obter mais informações, consulte [Use criptografia de nível de campo para ajudar a proteger dados confidenciais](#).

Os endpoints da API do CloudFront `cloudfront.amazonaws.com` e `cloudfront-fips.amazonaws.com` só aceitam tráfego HTTPS. Isso significa que, quando você envia e recebe informações usando a API do CloudFront, seus dados, incluindo configurações de distribuição, políticas de cache e políticas de solicitação de origem, grupos de chaves e chaves públicas e código de função no CloudFront Functions, são sempre criptografados em trânsito. Além disso, todas as solicitações enviadas para os endpoints da API do CloudFront são assinadas com credenciais da AWS e conectadas no AWS CloudTrail.

O código de função e a configuração no CloudFront Functions são sempre criptografados em trânsito quando copiados para os pontos de presença (POPs) do local da borda e entre outros locais de armazenamento usados pelo CloudFront.

Criptografia em repouso

O código de função e a configuração no CloudFront Functions sempre são armazenados em um formato criptografado nos PoPs do local da borda e em outros locais de armazenamento usados pelo CloudFront.

Restringir o acesso ao conteúdo

Várias empresas que distribuem conteúdo pela Internet querem restringir o acesso a documentos, dados de negócio, streams de mídia ou conteúdo destinado a um subgrupo de usuários. Para fornecer esse conteúdo com segurança usando o Amazon CloudFront, você pode:

Usar cookies ou URLs assinados

Restringir o acesso ao conteúdo que é destinado a usuários selecionados, por exemplo, aqueles que pagaram uma taxa, fornecendo esse conteúdo privado por meio do CloudFront usando URL ou cookies assinados. Para obter mais informações, consulte [Veicular conteúdo privado com URLs e cookies assinados](#).

Restringir o acesso ao conteúdo em buckets do Amazon S3

Se você restringir o acesso ao conteúdo usando, por exemplo, URLs assinados pelo CloudFront ou cookies assinados, também certamente não vai querer que as pessoas visualizem os arquivos

usando o URL direto do arquivo. Em vez disso, é melhor que elas acessem os arquivos usando o URL do CloudFront, para que suas proteções funcionem.

Se você usar um bucket do Amazon S3 como a origem de uma distribuição do CloudFront, poderá configurar um controle de acesso à origem (OAC) que possibilita restringir o acesso ao bucket do S3. Para ter mais informações, consulte [the section called “Restringir o acesso a uma origem do Amazon Simple Storage Service”](#).

Restringir o acesso ao conteúdo fornecido por um Application Load Balancer

Ao usar o CloudFront com um Application Load Balancer no Elastic Load Balancing como origem, é possível configurar o CloudFront para evitar que os usuários acessem diretamente o Application Load Balancer. Assim os usuários podem acessar o Application Load Balancer somente por meio do CloudFront, assegurando que você obtenha os benefícios de usá-lo. Para obter mais informações, consulte [Restringir o acesso aos Application Load Balancers](#).

Usar Web ACLs do AWS WAF

É possível usar o AWS WAF, um serviço de firewall de aplicativo web, para criar uma lista de controle de acesso à web (web ACL) e restringir o acesso ao seu conteúdo. Com base nas condições especificadas por você, como o endereço IP de origem da solicitação ou os valores das strings de consulta, o CloudFront responde às solicitações com o conteúdo solicitado ou com um código de status HTTP 403 (Proibido). Para obter mais informações, consulte [Usar proteções do AWS WAF](#).

Usar a restrição geográfica

É possível usar a restrição geográfica, também conhecida como geobloqueio, para impedir que os usuários de algumas localizações específicas acessem o conteúdo que você fornece por meio da distribuição do CloudFront. Há várias opções para escolher na configuração de restrições geográficas. Para obter mais informações, consulte [Restringir a distribuição geográfica do conteúdo](#).

Identity and Access Management para Amazon CloudFront

O AWS Identity and Access Management (IAM) é um serviço da AWS service (Serviço da AWS) que ajuda a controlar o acesso aos recursos da AWS de forma segura. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (ter permissões) para usar os

recursos do CloudFront. O IAM é um AWS service (Serviço da AWS) que pode ser usado sem custo adicional.

Tópicos

- [Público](#)
- [Autenticando com identidades](#)
- [Gerenciando acesso usando políticas](#)
- [Como o Amazon CloudFront funciona com o IAM](#)
- [Exemplos de políticas baseadas em identidade do Amazon CloudFront](#)
- [Políticas gerenciadas pela AWS para Amazon CloudFront](#)
- [Solução de problemas de identidade e acesso do Amazon CloudFront](#)

Público

O uso do AWS Identity and Access Management (IAM) varia dependendo do trabalho que é realizado no CloudFront.

Usuário do serviço: se você usar o serviço CloudFront para fazer o trabalho, o administrador fornecerá as credenciais e as permissões necessárias. À medida que usar mais recursos do CloudFront para fazer seu trabalho, você poderá precisar de permissões adicionais. Entender como o acesso é gerenciado pode ajudar você a solicitar as permissões corretas ao seu administrador. Se não for possível acessar um recurso no CloudFront, consulte [Solução de problemas de identidade e acesso do Amazon CloudFront](#).

Service administrator (Administrador do serviço): se você for o responsável pelos recursos do CloudFront na empresa, provavelmente terá acesso total ao CloudFront. É seu trabalho determinar quais funcionalidades e recursos do CloudFront seus usuários de serviço devem acessar. Assim, você deve enviar solicitações ao administrador do IAM para alterar as permissões dos usuários de seu serviço. Revise as informações nesta página para entender os Introdução ao IAM. Para saber mais sobre como a empresa pode usar o IAM com o CloudFront, consulte [Como o Amazon CloudFront funciona com o IAM](#).

Administrador do IAM: se você for um administrador do IAM, talvez queira saber detalhes sobre a criação de políticas para gerenciar o acesso ao CloudFront. Para visualizar exemplos de políticas baseadas em identidade do CloudFront que podem ser usadas no IAM, consulte [Exemplos de políticas baseadas em identidade do Amazon CloudFront](#).

Autenticando com identidades

A autenticação é a forma como você faz login na AWS usando suas credenciais de identidade. É necessário ser autenticado (fazer login na AWS) como Usuário raiz da conta da AWS, como usuário do IAM, ou assumindo um perfil do IAM.

Você pode fazer login na AWS como uma identidade federada usando credenciais fornecidas por uma fonte de identidades. AWS IAM Identity Center Os usuários (IAM Identity Center), a autenticação única da empresa e as suas credenciais do Google ou do Facebook são exemplos de identidades federadas. Quando você faz login como identidade federada, o administrador já configurou anteriormente a federação de identidades usando perfis do IAM. Quando você acessa a AWS usando a federação, está indiretamente assumindo um perfil.

A depender do tipo de usuário, você pode fazer login no AWS Management Console ou no portal de acesso AWS. Para obter mais informações sobre como fazer login na AWS, consulte [Como fazer login na conta](#) no Início de Sessão da AWS Guia do usuário .

Se você acessar a AWS programaticamente, a AWS fornecerá um kit de desenvolvimento de software (SDK) e uma interface de linha de comando (CLI) para você assinar criptograficamente as solicitações usando as suas credenciais. Se você não utilizar as ferramentas AWS, deverá designar as solicitações por conta própria. Para ter mais informações sobre como usar o método recomendado para assinar solicitações por conta própria, consulte [AWS Signature Version 4 para solicitações de API](#) no Guia do usuário do IAM.

Independente do método de autenticação usado, também pode ser exigido que você forneça informações adicionais de segurança. Por exemplo, a AWS recomenda o uso da autenticação multifator (MFA) para aumentar a segurança de sua conta. Para saber mais, consulte [Multi-factor authentication](#) no Guia do usuário do AWS IAM Identity Center e [Código da autenticação multifator no IAM da AWS](#) no Guia do usuário do IAM.

Usuário raiz Conta da AWS

Ao criar uma Conta da AWS, você começa com uma identidade de login com acesso completo a todos os Serviços da AWS e recursos na conta. Essa identidade, chamada usuário raiz da Conta da AWS, é acessada por login com o endereço de e-mail e a senha usada para criar a conta. É altamente recomendável não usar o usuário raiz para tarefas diárias. Proteja as credenciais do usuário raiz e use-as para executar as tarefas que somente ele puder executar. Para obter a lista completa das tarefas que exigem login como usuário raiz, consulte [Tarefas que exigem credenciais de usuário raiz](#) no Guia do Usuário do IAM.

Identidade federada

Como prática recomendada, exija que os usuários, inclusive os que precisam de acesso de administrador, usem a federação com um provedor de identidades para acessar os Serviços da AWS usando credenciais temporárias.

Identidade federada é um usuário de seu diretório de usuários corporativos, um provedor de identidades da web, o AWS Directory Service, o diretório do Identity Center, ou qualquer usuário que acesse os Serviços da AWS usando credenciais fornecidas por meio de uma fonte de identidade. Quando as identidades federadas acessam Contas da AWS, elas assumem perfis que fornecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, recomendamos usar o AWS IAM Identity Center. Você pode criar usuários e grupos no IAM Identity Center ou conectar-se e sincronizar com um conjunto de usuários e grupos em sua própria fonte de identidade para uso em todas as suas Contas da AWS e aplicações. Para obter mais informações sobre o Centro de Identidade do IAM, consulte [O que é o Centro de Identidade do IAM?](#) no Manual do Usuário do AWS IAM Identity Center.

Usuários e grupos do IAM

Um [usuário do IAM](#) é uma identidade dentro da Conta da AWS que tem permissões específicas para uma única pessoa ou aplicativo. Sempre que possível, recomendamos contar com credenciais temporárias em vez de criar usuários do IAM com credenciais de longo prazo, como senhas e chaves de acesso. No entanto, se você tiver casos de uso específicos que exijam credenciais de longo prazo com usuários do IAM, recomendamos alternar as chaves de acesso. Para obter mais informações, consulte [Altere as chaves de acesso regularmente para casos de uso que exijam credenciais de longo prazo](#) no Guia do Usuário do IAM.

Um [grupo do IAM](#) é uma identidade que especifica uma coleção de usuários do IAM. Não é possível fazer login como um grupo. É possível usar grupos para especificar permissões para vários usuários de uma vez. Os grupos facilitam o gerenciamento de permissões para grandes conjuntos de usuários. Por exemplo, você pode ter um grupo chamado IAMAdmins e conceder a esse grupo permissões para administrar recursos do IAM.

Usuários são diferentes de perfis. Um usuário é exclusivamente associado a uma pessoa ou a uma aplicação, mas um perfil pode ser assumido por qualquer pessoa que precisar dele. Os usuários têm credenciais permanentes de longo prazo, mas os perfis fornecem credenciais temporárias. Para saber mais, consulte [Casos de uso de usuários do IAM](#) no Guia do usuário do IAM.

Perfis do IAM

Um [perfil do IAM](#) é uma identidade dentro da Conta da AWS que tem permissões específicas. Ele é semelhante a um usuário do IAM, mas não está associado a uma pessoa específica. Para assumir temporariamente um perfil do IAM no AWS Management Console, você pode [alternar de um usuário para um perfil do IAM \(console\)](#). É possível presumir um perfil chamando uma operação de API da AWS CLI ou da AWS, ou usando um URL personalizado. Para obter mais informações sobre métodos para usar perfis, consulte [Métodos para assumir um perfil](#) no Guia do usuário do IAM.

Funções do IAM com credenciais temporárias são úteis nas seguintes situações:

- **Acesso de usuário federado:** para atribuir permissões a identidades federadas, você pode criar um perfil e definir permissões para ele. Quando uma identidade federada é autenticada, essa identidade é associada ao perfil e recebe as permissões definidas pelo mesmo. Para obter mais informações sobre perfis para federação, consulte [Criar um perfil para um provedor de identidades de terceiros](#) no Guia do Usuário do IAM. Se você usar o Centro de identidade do IAM, configure um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o Centro de identidade do IAM correlaciona o conjunto de permissões a um perfil no IAM. Para obter informações sobre conjuntos de permissões, consulte [Conjuntos de Permissões](#) no Manual do Usuário do AWS IAM Identity Center.
- **Permissões temporárias para usuários do IAM** — um usuário ou um perfil do IAM pode presumir um perfil do IAM para obter temporariamente permissões diferentes para uma tarefa específica.
- **Acesso entre contas** — é possível usar um perfil do IAM para permitir que alguém (uma entidade principal confiável) em outra conta acesse recursos em sua conta. Os perfis são a principal forma de conceder acesso entre contas. No entanto, alguns Serviços da AWS permitem que você anexe uma política diretamente a um recurso (em vez de usar um perfil como proxy). Para saber a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.
- **Acesso entre serviços:** alguns Serviços da AWS usam atributos em outros Serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicativos no Amazon EC2 ou armazene objetos no Amazon S3. Um serviço pode fazer isso usando as permissões do principal de chamada, usando um perfil de serviço ou um perfil vinculado a serviço.
 - **Encaminhamento de sessões de acesso (FAS):** qualquer pessoa que utilizar uma função ou usuário do IAM para realizar ações na AWS é considerada uma entidade principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O recurso FAS utiliza as permissões da entidade principal que chama um AWS service (Serviço da

AWS), combinadas às permissões do AWS service (Serviço da AWS) solicitante, para realizar solicitações para serviços downstream. As solicitações de FAS só são feitas quando um serviço recebe uma solicitação que exige interações com outros Serviços da AWS ou com recursos para serem concluídas. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Encaminhar sessões de acesso](#).

- Função de serviço: um perfil de serviço é um [perfil do IAM](#) que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte [Criar um perfil para delegar permissões a um AWS service \(Serviço da AWS\)](#) no Guia do Usuário do IAM.
- Perfil vinculado a serviço: um perfil vinculado a serviço é um tipo de perfil de serviço vinculado a um AWS service (Serviço da AWS). O serviço pode presumir a função de executar uma ação em seu nome. Funções vinculadas ao serviço aparecem em sua Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para funções vinculadas ao serviço.
- Aplicações em execução no Amazon EC2: é possível usar um perfil do IAM para gerenciar credenciais temporárias para aplicações em execução em uma instância do EC2 e fazer solicitações da AWS CLI ou da AWS API. É preferível fazer isso a armazenar chaves de acesso na instância do EC2. Para atribuir um perfil da AWS a uma instância do EC2 e disponibilizá-la para todas as suas aplicações, crie um perfil de instância que esteja anexado a ela. Um perfil de instância contém o perfil e permite que os programas em execução na instância do EC2 obtenham credenciais temporárias. Para mais informações, consulte [Utilizar um perfil do IAM para conceder permissões a aplicações em execução nas instâncias do Amazon EC2](#) no Guia do usuário do IAM.

Gerenciando acesso usando políticas

Você controla o acesso na AWS criando políticas e anexando-as a identidades ou atributos da AWS. Uma política é um objeto na AWS que, quando associado a uma identidade ou recurso, define suas permissões. A AWS avalia essas políticas quando uma entidade principal (usuário, usuário raiz ou sessão de perfil) faz uma solicitação. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas é armazenada na AWS como documentos JSON. Para obter mais informações sobre a estrutura e o conteúdo de documentos de políticas JSON, consulte [Visão geral das políticas JSON](#) no Guia do Usuário do IAM.

Os administradores podem usar as políticas JSON da AWS para especificar quem tem acesso a o quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

Por padrão, usuários e funções não têm permissões. Para conceder aos usuários permissões para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis e os usuários podem presumir os perfis.

As políticas do IAM definem permissões para uma ação independente do método usado para executar a operação. Por exemplo, suponha que você tenha uma política que permite a ação `iam:GetRole`. Um usuário com essa política pode obter informações de perfis do AWS Management Console, da AWS CLI ou da API da AWS.

Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário do IAM, grupo de usuários ou perfil. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Defina permissões personalizadas do IAM com políticas gerenciadas pelo cliente](#) no Guia do usuário do IAM.

As políticas baseadas em identidade podem ser categorizadas ainda adicionalmente como políticas em linha ou políticas gerenciadas. As políticas em linha são anexadas diretamente a um único usuário, grupo ou perfil. As políticas gerenciadas são políticas independentes que podem ser anexadas a vários usuários, grupos e perfis na Conta da AWS. As políticas gerenciadas incluem políticas gerenciadas pela AWS e políticas gerenciadas pelo cliente. Para saber como escolher entre uma política gerenciada ou uma política em linha, consulte [Escolher entre políticas gerenciadas e políticas em linha](#) no Guia do usuário do IAM.

Políticas baseadas no recurso

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações um principal especificado pode executar nesse recurso e em que condições. Você deve [especificar uma entidade principal](#) em uma política baseada em recursos. As entidades principais podem incluir contas, usuários, perfis, usuários federados ou Serviços da AWS.

Políticas baseadas em recursos são políticas em linha localizadas nesse serviço. Não é possível usar as políticas gerenciadas da AWS do IAM em uma política baseada em atributos.

Listas de controle de acesso (ACLs)

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou perfis da conta) têm permissões para acessar um recurso. As ACLs são semelhantes as políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

Amazon S3, AWS WAF e Amazon VPC são exemplos de serviços que oferecem compatibilidade com ACLs. Para saber mais sobre ACLs, consulte [Visão geral da lista de controle de acesso \(ACL\)](#) no Guia do Desenvolvedor do Amazon Simple Storage Service.

Outros tipos de política

A AWS oferece compatibilidade com tipos de política menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- **Limites de permissões:** um limite de permissões é um atributo avançado no qual você define o máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM (usuário ou perfil do IAM). É possível definir um limite de permissões para uma entidade. As permissões resultantes são a interseção das políticas baseadas em identidade de uma entidade com seus limites de permissões. As políticas baseadas em recurso que especificam o usuário ou o perfil no campo `Principal` não são limitadas pelo limite de permissões. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações sobre limites de permissões, consulte [Limites de permissões para identidades do IAM](#) no Guia do Usuário do IAM.
- **Políticas de controle de serviço (SCPs)** — SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (UO) no AWS Organizations. O AWS Organizations é um serviço que agrupa e gerencia centralmente várias Contas da AWS pertencentes a sua empresa. Se você habilitar todos os atributos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as contas. O SCP limita as permissões para entidades em contas membro, o que inclui cada Usuário raiz da conta da AWS. Para obter mais informações sobre o Organizations e SCPs, consulte [Service control policies](#) no Guia do usuário do AWS Organizations.
- **Políticas de sessão:** são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para um perfil ou um usuário federado. As permissões da sessão resultante são a interseção das políticas baseadas em identidade do usuário ou do perfil e das políticas de sessão. As permissões também podem ser provenientes de uma política baseada em atributo. Uma negação explícita em qualquer uma dessas políticas

substitui a permissão. Para obter mais informações, consulte [Políticas de sessão](#) no Guia do Usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como a AWS determina permitir ou não uma solicitação quando há vários tipos de política envolvidos, consulte [Lógica da avaliação de políticas](#) no Guia do Usuário do IAM.

Como o Amazon CloudFront funciona com o IAM

Antes de usar o IAM para gerenciar o acesso ao CloudFront, saiba quais recursos do IAM estão disponíveis para uso com o CloudFront.

Recursos do IAM que você pode usar com o Amazon CloudFront

Atributo do IAM	Suporte do CloudFront
Políticas baseadas em identidade	Sim
Políticas baseadas em recursos	Não
Ações das políticas	Sim
Atributos de políticas	Sim
Chaves de condição de política (específicas do serviço)	Sim
ACLs	Não
ABAC (tags em políticas)	Parcial
Credenciais temporárias	Sim
Sessões de acesso direto (FAS)	Não
Perfis de serviço	Não

Atributo do IAM	Suporte do CloudFront
Funções vinculadas ao serviço	Sim

Para obter uma visão geral de como o CloudFront e outros serviços da AWS funcionam com a maioria dos recursos do IAM, consulte [Serviços da AWS que funcionam com o IAM](#) no Guia do usuário do IAM.

Políticas baseadas em identidade para o CloudFront

Compatível com políticas baseadas em identidade: Sim

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário, grupo de usuários ou perfil do IAM. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Defina permissões personalizadas do IAM com políticas gerenciadas pelo cliente](#) no Guia do usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações ou recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. Você não pode especificar a entidade principal em uma política baseada em identidade porque ela se aplica ao usuário ou perfil ao qual ela está anexada. Para saber mais sobre todos os elementos que podem ser usados em uma política JSON, consulte [Referência de elementos da política JSON do IAM](#) no Guia do Usuário do IAM.

Exemplos de políticas baseadas em identidade para o CloudFront

Para visualizar exemplos de políticas baseadas em identidade do CloudFront, consulte [Exemplos de políticas baseadas em identidade do Amazon CloudFront](#).

Políticas baseadas em recursos no CloudFront

Suporte a políticas baseadas em recursos: não

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações um principal especificado

pode executar nesse recurso e em que condições. Você deve [especificar uma entidade principal](#) em uma política baseada em recursos. As entidades principais podem incluir contas, usuários, funções, usuários federados ou Serviços da AWS.

Para permitir o acesso entre contas, você pode especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em atributo. Adicionar uma entidade principal entre contas à política baseada em recurso é apenas metade da tarefa de estabelecimento da relação de confiança. Quando a entidade principal e o recurso estão em diferentes Contas da AWS, um administrador do IAM da conta confiável também deve conceder à entidade principal (usuário ou perfil) permissão para acessar o recurso. Eles concedem permissão ao anexar uma política baseada em identidade para a entidade. No entanto, se uma política baseada em recurso conceder acesso a uma entidade principal na mesma conta, nenhuma política baseada em identidade adicional será necessária. Consulte mais informações em [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.

Ações de políticas do CloudFront

Compatível com ações de políticas: Sim

Os administradores podem usar AWS as políticas JSON para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento `Action` de uma política JSON descreve as ações que você pode usar para permitir ou negar acesso em uma política. As ações de políticas geralmente têm o mesmo nome que a operação de API da AWS associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Algumas operações também exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Incluem ações em uma política para conceder permissões para executar a operação associada.

Para ver uma lista das ações do CloudFront, consulte [Ações definidas pelo Amazon CloudFront](#) na Referência de autorização do serviço.

As ações de políticas no CloudFront usam o seguinte prefixo antes da ação:

```
cloudfront
```

Para especificar várias ações em uma única instrução, separe-as com vírgulas.

```
"Action": [
```



```
"cloudfront:action1",  
"cloudfront:action2"  
]
```

Para visualizar exemplos de políticas baseadas em identidade do CloudFront, consulte [Exemplos de políticas baseadas em identidade do Amazon CloudFront](#).

Recursos de políticas do CloudFront

Compatível com recursos de políticas: Sim

Os administradores podem usar AWS as políticas JSON para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento de política JSON `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou `NotResource`. Como prática recomendada, especifique um recurso usando seu [nome do recurso da Amazon \(ARN\)](#). Isso pode ser feito para ações que oferecem compatibilidade com um tipo de recurso específico, conhecido como permissões em nível de recurso.

Para ações que não oferecem compatibilidade com permissões em nível de recurso, como operações de listagem, use um curinga (*) para indicar que a instrução se aplica a todos os recursos.

```
"Resource": "*"
```

Para ver uma lista dos tipos de recursos do CloudFront e seus ARNs, consulte [Tipos de recursos definidos pelo Amazon CloudFront](#) na Referência de autorização do serviço. Para saber com quais ações você pode especificar o ARN de cada recurso, consulte [Ações definidas pelo Amazon CloudFront](#).

Para visualizar exemplos de políticas baseadas em identidade do CloudFront, consulte [Exemplos de políticas baseadas em identidade do Amazon CloudFront](#).

Chaves de condição de políticas para o CloudFront

Compatível com chaves de condição de política específicas de serviço: Sim

Os administradores podem usar AWS as políticas JSON para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento `Condition` (ou bloco `Condition`) permite que você especifique condições nas quais uma instrução estiver em vigor. O elemento `Condition` é opcional. É possível criar expressões condicionais que usem [agentes de condição](#), como “igual a” ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação.

Se você especificar vários elementos `Condition` em uma instrução ou várias chaves em um único `Condition` elemento, a AWS os avaliará usando uma operação lógica AND. Se você especificar vários valores para uma única chave de condição, a AWS avaliará a condição usando uma operação lógica OR. Todas as condições devem ser atendidas antes que as permissões da instrução sejam concedidas.

Você também pode usar variáveis de espaço reservado ao especificar condições. Por exemplo, é possível conceder a um usuário do IAM permissão para acessar um atributo somente se ele estiver marcado com seu nome de usuário do IAM. Para obter mais informações, consulte [Elementos da política do IAM: variáveis e tags](#) no Guia do usuário do IAM.

A AWS oferece compatibilidade com chaves de condição globais e chaves de condição específicas do serviço. Para ver todas as chaves de condição globais da AWS, consulte [Chaves de Contexto de Condição Globais da AWS](#) no Guia do Usuário do IAM.

Para ver uma lista de chaves de condição do CloudFront, consulte [Chaves de condição do Amazon CloudFront](#) na Referência de autorização do serviço. Para saber com quais ações e recursos você pode usar a chave de condição, consulte [Ações definidas pelo Amazon CloudFront](#).

Para visualizar exemplos de políticas baseadas em identidade do CloudFront, consulte [Exemplos de políticas baseadas em identidade do Amazon CloudFront](#).

ACLs no CloudFront

Compatível com ACLs: Não

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou perfis da conta) têm permissões para acessar um recurso. As ACLs são semelhantes as políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

ABAC com o CloudFront

Compatível com ABAC (tags em políticas): Parcial

O controle de acesso por atributo (ABAC) é uma estratégia de autorização que define permissões com base em atributos. Na AWS, esses atributos são chamados de tags. É possível anexar tags

a entidades do IAM (usuários ou perfis) e a muitos recursos da AWS. A marcação de entidades e atributos é a primeira etapa do ABAC. Em seguida, você cria políticas de ABAC para permitir operações quando a tag da entidade principal corresponder à tag do recurso que ela estiver tentando acessar.

O ABAC é útil em ambientes que estão crescendo rapidamente e ajuda em situações onde o gerenciamento de políticas se torna um problema.

Para controlar o acesso baseado em tags, forneça informações sobre as tags no [elemento de condição](#) de uma política usando as `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou chaves de condição `aws:TagKeys`.

Se um serviço oferecer suporte às três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço oferecer suporte às três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para obter mais informações sobre o ABAC, consulte [Definir permissões com autorização ABAC](#) no Guia do usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte [Utilizar controle de acesso baseado em atributos \(ABAC\)](#) no Guia do usuário do IAM.

O CloudFront é compatível com ABAC somente para distribuições.

Usar credenciais temporárias com o CloudFront

Compatível com credenciais temporárias: Sim

Alguns Serviços da AWS não funcionam quando você faz login usando credenciais temporárias. Para obter informações adicionais, como quais Serviços da AWS funcionam com credenciais temporárias, consulte [Serviços da AWS que funcionem com o IAM](#) no Guia do Usuário do IAM.

Você está usando credenciais temporárias se fizer login no AWS Management Console por qualquer método, exceto nome de usuário e uma senha. Por exemplo, quando você acessa a AWS usando o link de autenticação única (SSO) da sua empresa, esse processo cria credenciais temporárias automaticamente. Você também cria automaticamente credenciais temporárias quando faz login no console como usuário e, em seguida, alterna perfis. Para ter mais informações sobre como alternar perfis, consulte [Mudar de um usuário para um perfil do IAM \(console\)](#) no Guia do usuário do IAM.

Você pode criar credenciais temporárias manualmente usando a API AWS CLI ou AWS. Em seguida, você pode usar essas credenciais temporárias para acessar a AWS. A AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para mais informações, consulte [Credenciais de segurança temporárias no IAM](#).

Sessões de acesso direto do CloudFront

Compatível com sessões de acesso direto (FAS): não

Quando você usa um usuário ou um perfil do IAM para executar ações na AWS, você é considerado uma entidade principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O recurso FAS utiliza as permissões da entidade principal que chama um AWS service (Serviço da AWS), combinadas às permissões do AWS service (Serviço da AWS) solicitante, para realizar solicitações para serviços downstream. As solicitações de FAS só são feitas quando um serviço recebe uma solicitação que exige interações com outros Serviços da AWS ou com recursos para serem concluídas. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Encaminhar sessões de acesso](#).

Perfis de serviço do CloudFront

Compatível com perfis de serviço: não

O perfil de serviço é um [perfil do IAM](#) que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte [Criar um perfil para delegar permissões a um AWS service \(Serviço da AWS\)](#) no Guia do Usuário do IAM.

Warning

Alterar as permissões de um perfil de serviço pode interromper a funcionalidade do CloudFront. Edite perfis de serviço somente quando o CloudFront fornecer orientação para isso.

Funções vinculadas ao serviço do CloudFront

Suporte a perfis vinculados a serviços: sim

Um perfil vinculado ao serviço é um tipo de perfil de serviço vinculado a um AWS service (Serviço da AWS). O serviço pode presumir a função de executar uma ação em seu nome. Funções vinculadas ao serviço aparecem em sua Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para funções vinculadas ao serviço.

O Lambda@Edge usa perfis vinculados ao serviço para realizar ações para você. Consulte mais informações sobre como criar ou gerenciar perfis vinculados ao serviço do CloudFront em [Funções vinculadas ao serviço para o Lambda@Edge](#).

Para obter detalhes sobre como criar ou gerenciar perfis vinculados a serviços, consulte [Serviços da AWS que funcionam com o IAM](#). Encontre um serviço na tabela que inclua um Yes na coluna Função vinculada ao serviço. Escolha o link Sim para visualizar a documentação do perfil vinculado a serviço desse serviço.

Exemplos de políticas baseadas em identidade do Amazon CloudFront

Por padrão, usuários e funções não têm permissão para criar nem modificar recursos do CloudFront. Eles também não podem executar tarefas usando o AWS Management Console, a AWS Command Line Interface (AWS CLI) ou a API AWS. Para conceder aos usuários permissão para executar ações nos recursos de que precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis e os usuários podem presumir os perfis.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documento de política JSON, consulte [Criar políticas do IAM \(console\)](#) no Guia do usuário do IAM.

Para obter detalhes sobre ações e tipos de recurso definidos pelo CloudFront, por exemplo, o formato dos ARNs para cada um dos tipos de recurso, consulte [Ações, recursos e chaves de condição do Amazon CloudFront](#) na Referência de autorização do serviço.

Tópicos

- [Melhores práticas de política](#)
- [Permitir que usuários visualizem suas próprias permissões](#)
- [Permissões para acessar o CloudFront programaticamente](#)
- [Permissões necessárias para usar o console do CloudFront](#)
- [Exemplos de política gerenciada pelo cliente](#)

Melhores práticas de política

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos do CloudFront em sua conta. Essas ações podem incorrer em custos para seus Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas gerenciadas pela AWS e avance para as permissões de privilégio mínimo — para começar a conceder permissões a seus usuários e workloads, use as políticas gerenciadas pela AWS, que concedem permissões para muitos casos de uso comuns. Elas estão disponíveis em sua Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo cliente AWS específicas para seus casos de uso. Para obter mais informações, consulte [Políticas gerenciadas pela AWS](#) ou [Políticas gerenciadas pela AWS para funções de trabalho](#) no Guia do Usuário do IAM.
- Aplique permissões de privilégio mínimo — ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em atributos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para obter mais informações sobre como usar o IAM para aplicar permissões, consulte [Políticas e permissões no IAM](#) no Guia do Usuário do IAM.
- Use condições nas políticas do IAM para restringir ainda mais o acesso — você pode adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, você pode gravar uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso a ações de serviço, se elas forem usadas por meio de um AWS service (Serviço da AWS) específico, como o AWS CloudFormation. Para obter mais informações, consulte [Elementos da política JSON do IAM: Condição](#) no Guia do usuário do IAM.
- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais — o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de 100 verificações de política e recomendações acionáveis para ajudá-lo a criar políticas seguras e funcionais. Para ter mais informações, consulte [Validar políticas com o IAM Access Analyzer](#) no Guia do usuário do IAM.
- Exigir autenticação multifator (MFA) — se houver um cenário que exija usuários do IAM ou um usuário raiz em sua Conta da AWS, ative a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para ter mais informações, consulte [Acesso seguro à API com a MFA](#) no Guia do usuário do IAM.

Para obter mais informações sobre as práticas recomendadas do IAM, consulte [Práticas Recomendadas de Segurança no IAM](#) no Guia do Usuário do IAM.

Permitir que usuários visualizem suas próprias permissões

Este exemplo mostra como criar uma política que permita que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou de forma programática usando a AWS CLI ou a AWS API.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

Permissões para acessar o CloudFront programaticamente

Veja a seguir uma política de permissões. O Sid, ou o ID de instrução, é opcional.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAllCloudFrontPermissions",
      "Effect": "Allow",
      "Action": ["cloudfront:*"],
      "Resource": "*"
    }
  ]
}
```

A política concede permissões para executar todas as operações do CloudFront, o que é suficiente para acessar o CloudFront de forma programática. Se estiver usando o console para acessar o CloudFront, consulte [Permissões necessárias para usar o console do CloudFront](#).

Para visualizar uma lista de ações e o ARN que podem ser especificadas para conceder ou negar permissão para usar cada ação, consulte [Ações, recursos e chaves de condição do Amazon CloudFront](#) na Referência de autorização do serviço.

Permissões necessárias para usar o console do CloudFront

Para permitir acesso total ao console do CloudFront, conceda as permissões na seguinte política de permissões:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "acm:ListCertificates",
        "cloudfront:*",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch:GetMetricStatistics",
        "elasticloadbalancing:DescribeLoadBalancers",
        "iam:ListServerCertificates",

```



```

        "sns:ListSubscriptionsByTopic",
        "sns:ListTopics",
        "waf:GetWebACL",
        "waf:ListWebACLs"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "s3:ListAllMyBuckets",
        "s3:PutBucketPolicy"
    ],
    "Resource": "arn:aws:s3:::*"
}
]
}

```

Veja por que as permissões são necessárias:

acm:ListCertificates

Permite que você visualize uma lista de certificados do ACM quando estiver criando e atualizando distribuições por meio do console do CloudFront e quiser configurar o CloudFront para exigir HTTPS entre o visualizador e o CloudFront ou entre o CloudFront e a origem.

Essa permissão não é necessária se você não estiver usando o console do CloudFront.

cloudfront:*

Permite executar todas as ações do CloudFront.

cloudwatch:DescribeAlarms e **cloudwatch:PutMetricAlarm**

Permite criar e exibir alarmes do CloudWatch no console do CloudFront. Consulte também `sns:ListSubscriptionsByTopic` e `sns:ListTopics`.

Essas permissões não serão necessárias se você não estiver usando o console do CloudFront.

cloudwatch:GetMetricStatistics

Permite que o CloudFront renderize as métricas do CloudWatch no console do CloudFront.

Essa permissão não é necessária se você não estiver usando o console do CloudFront.

elasticloadbalancing:DescribeLoadBalancers

Quando estiver criando e atualizando distribuições, permite que você visualize uma lista de balanceadores de carga do Elastic Load Balancing na lista de origens disponíveis.

Essa permissão não é necessária se você não estiver usando o console do CloudFront.

iam:ListServerCertificates

Permite que você visualize uma lista de certificados no repositório de certificados do IAM quando estiver criando e atualizando distribuições por meio do console do CloudFront e quiser configurar o CloudFront para exigir HTTPS entre o visualizador e o CloudFront ou entre o CloudFront e a origem.

Essa permissão não é necessária se você não estiver usando o console do CloudFront.

s3:ListAllMyBuckets

Quando estiver criando e atualizando distribuições, permite que você execute as seguintes operações:

- Visualizar uma lista de buckets do S3 na lista de origens disponíveis.
- Visualizar uma lista de buckets do S3 nos quais você pode salvar logs de acesso..

Essa permissão não é necessária se você não estiver usando o console do CloudFront.

S3:PutBucketPolicy

Ao criar ou atualizar distribuições que restringem o acesso a buckets do S3, permite que um usuário atualize a política do bucket para conceder acesso à identidade de acesso de origem do CloudFront. Para ter mais informações, consulte [the section called “Usar uma identidade de acesso à origem \(herdada, não recomendado\)”](#).

Essa permissão não é necessária se você não estiver usando o console do CloudFront.

sns:ListSubscriptionsByTopic e sns:ListTopics

Ao criar alarmes do CloudWatch no console do CloudFront, permite escolher um tópico do SNS para notificações.

Essas permissões não serão necessárias se você não estiver usando o console do CloudFront.

waf:GetWebACL e waf:ListWebACLs

Permite exibir uma lista de ACLs da Web do AWS WAF no console do CloudFront.

Essas permissões não serão necessárias se você não estiver usando o console do CloudFront.

Exemplos de política gerenciada pelo cliente

Você pode criar suas próprias políticas personalizadas do IAM para conceder permissões a ações da API do CloudFront. Você pode anexar essas políticas personalizadas a usuários ou grupos do IAM que exijam as permissões especificadas. Essas políticas funcionam quando você está usando a API do CloudFront, os AWS SDKs ou a AWS CLI. Os exemplos a seguir mostram permissões para alguns casos de uso comuns. Para a política que concede acesso total a um usuário ao CloudFront, consulte [Permissões necessárias para usar o console do CloudFront](#).

Exemplos

- [Exemplo 1: permitir acesso de leitura a todas as distribuições](#)
- [Exemplo 2: permitir criar, atualizar e excluir distribuições](#)
- [Exemplo 3: permitir a criação e a listagem de invalidações](#)
- [Exemplo 4: permitir criar uma distribuição](#)

Exemplo 1: permitir acesso de leitura a todas as distribuições

A política a seguir concede ao usuário permissões para visualizar todas as distribuições no console do CloudFront:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "acm:ListCertificates",
        "cloudfront:GetDistribution",
        "cloudfront:GetDistributionConfig",
        "cloudfront:ListDistributions",
        "cloudfront:ListCloudFrontOriginAccessIdentities",
        "elasticloadbalancing:DescribeLoadBalancers",
        "iam:ListServerCertificates",
        "sns:ListSubscriptionsByTopic",
        "sns:ListTopics",
        "waf:GetWebACL",
        "waf:ListWebACLs"
      ]
    }
  ]
}
```

```

    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "s3:ListAllMyBuckets"
    ],
    "Resource": "arn:aws:s3:::*"
  }
]
}

```

Exemplo 2: permitir criar, atualizar e excluir distribuições

A política a seguir concede aos usuários permissão para criar, atualizar e excluir distribuições usando o console do CloudFront:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "acm:ListCertificates",
        "cloudfront:CreateDistribution",
        "cloudfront>DeleteDistribution",
        "cloudfront:GetDistribution",
        "cloudfront:GetDistributionConfig",
        "cloudfront:ListDistributions",
        "cloudfront:UpdateDistribution",
        "cloudfront:ListCloudFrontOriginAccessIdentities",
        "elasticloadbalancing:DescribeLoadBalancers",
        "iam:ListServerCertificates",
        "sns:ListSubscriptionsByTopic",
        "sns:ListTopics",
        "waf:GetWebACL",
        "waf:ListWebACLs"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",

```

```
    "Action":[
      "s3:ListAllMyBuckets",
      "s3:PutBucketPolicy"
    ],
    "Resource":"arn:aws:s3:::*"
  }
]
```

Com a permissão `cloudfront:ListCloudFrontOriginAccessIdentities`, os usuários podem conceder automaticamente a uma identidade de acesso de origem a permissão para acessar objetos em um bucket do Amazon S3. Se você quiser que os usuários também possam criar identidades de acesso de origem, é necessário conceder a permissão `cloudfront:CreateCloudFrontOriginAccessIdentity`.

Exemplo 3: permitir a criação e a listagem de invalidações

A política de a seguir permite que os usuários criem e indiquem invalidações. Ela inclui acesso de leitura a distribuições do CloudFront, pois você cria e visualiza invalidações exibindo as configurações de uma distribuição:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "acm:ListCertificates",
        "cloudfront:GetDistribution",
        "cloudfront:GetStreamingDistribution",
        "cloudfront:GetDistributionConfig",
        "cloudfront:ListDistributions",
        "cloudfront:ListCloudFrontOriginAccessIdentities",
        "cloudfront:CreateInvalidation",
        "cloudfront:GetInvalidation",
        "cloudfront:ListInvalidations",
        "elasticloadbalancing:DescribeLoadBalancers",
        "iam:ListServerCertificates",
        "sns:ListSubscriptionsByTopic",
        "sns:ListTopics",
        "waf:GetWebACL",
        "waf:ListWebACLs"
      ]
    }
  ]
}
```

```
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "s3:ListAllMyBuckets"
    ],
    "Resource": "arn:aws:s3:::*"
  }
]
```

Exemplo 4: permitir criar uma distribuição

A política a seguir concede ao usuário permissão para criar e listar as distribuições no console do CloudFront. Para a ação `CreateDistribution`, especifique o caractere curinga (*) para o Resource em vez de um curinga para o ARN (`arn:aws:cloudfront::123456789012:distribution/*`) de distribuição. Para ter mais informações sobre o elemento Resource, consulte [Elementos de política JSON do IAM: recurso](#) no Guia do usuário do IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "cloudfront:CreateDistribution",
      "Resource": "*"
    },
    {
      "Sid": "VisualEditor1",
      "Effect": "Allow",
      "Action": "cloudfront:ListDistributions",
      "Resource": "*"
    }
  ]
}
```

Políticas gerenciadas pela AWS para Amazon CloudFront

Para adicionar permissões a usuários, grupos e funções, é mais fácil usar políticas gerenciadas pela AWS do que gravar políticas por conta própria. É necessário tempo e experiência para [criar políticas gerenciadas pelo cliente do IAM](#) que fornecem aos seus usuários apenas as permissões de que eles precisam. Para começar rapidamente, você pode usar nossas políticas gerenciadas pela AWS. Essas políticas abrangem casos de uso comuns e estão disponíveis na sua Conta da AWS. Para obter mais informações sobre as políticas gerenciadas da AWS, consulte [Políticas gerenciadas da AWS](#) no IAM User Guide.

Os serviços da AWS mantêm e atualizam políticas gerenciadas pela AWS. Não é possível alterar as permissões em políticas gerenciadas pela AWS. Os serviços ocasionalmente acrescentam permissões adicionais a uma política gerenciada pela AWS para oferecer suporte a novos recursos. Esse tipo de atualização afeta todas as identidades (usuários, grupos e funções) em que a política está anexada. É mais provável que os serviços atualizem uma política gerenciada pela AWS quando um novo recurso for iniciado ou novas permissões estiverem disponíveis. Os serviços não removem permissões de uma política gerenciada pela AWS, portanto, as atualizações de políticas não suspenderão suas permissões existentes.

Além disso, a AWS oferece suporte a políticas gerenciadas para funções de trabalho que abrangem vários serviços. Por exemplo, a política gerenciada pela denominada `ReadOnlyAccess` AWS fornece acesso somente leitura a todos os serviços e recursos da AWS. Quando um serviço executa um novo atributo, a AWS adiciona permissões somente leitura para novas operações e recursos. Para obter uma lista e descrições das políticas de funções de trabalho, consulte [Políticas gerenciadas pela AWS para funções de trabalho](#) no Guia do usuário do IAM.

Tópicos

- [Política gerenciada pela AWS: `CloudFrontReadOnlyAccess`](#)
- [Política gerenciada pela AWS: `CloudFrontFullAccess`](#)
- [Política gerenciada pela AWS: `AWSCloudFrontLogger`](#)
- [Política gerenciada pela AWS: `AWSLambdaReplicator`](#)
- [Atualizações do CloudFront em políticas gerenciadas pela AWS](#)

Política gerenciada pela AWS: CloudFrontReadOnlyAccess

Você pode anexar a política CloudFrontReadOnlyAccess a suas identidades do IAM. Essa política concede permissões somente leitura aos recursos do CloudFront. Ela também concede permissões somente leitura a outros recursos de serviços da AWS que estão relacionados ao CloudFront e são visíveis no console do CloudFront.

Detalhes das permissões

Esta política inclui as seguintes permissões:

- `cloudfront:Describe*`: permite que as entidades principais obtenham informações sobre metadados de recursos do CloudFront.
- `cloudfront:Get*`: permite que os principais obtenham informações detalhadas e configurações para recursos do CloudFront.
- `cloudfront:List*`: permite que os principais obtenham listas de recursos do CloudFront.
- `cloudfront-keyvaluestore:Describe*`: permite que as entidades principais obtenham informações sobre armazenamento de chave-valor.
- `cloudfront-keyvaluestore:Get*`: permite que as entidades principais obtenham informações detalhadas e configurações para o armazenamento de chave-valor.
- `cloudfront-keyvaluestore:List*`: permite que as entidades principais obtenham listas de armazenamentos de chave-valor.
- `acm:ListCertificates`: permite que os principais obtenham uma lista de certificados do ACM.
- `iam:ListServerCertificates`: permite que os principais obtenham uma lista de certificados de servidor armazenados no IAM.
- `route53:List*`: permite que os principais obtenham listas de recursos do Route 53.
- `waf:ListWebACLs`: permite que os principais obtenham uma lista de ACLs da Web no AWS WAF.
- `waf:GetWebACL`: permite que os principais obtenham informações detalhadas sobre ACLs da Web no AWS WAF.
- `wafv2:ListWebACLs`: permite que os principais obtenham uma lista de ACLs da Web no AWS WAF.
- `wafv2:GetWebACL`: permite que os principais obtenham informações detalhadas sobre ACLs da Web no AWS WAF.


```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "cfReadOnly",
      "Effect": "Allow",
      "Action": [
        "acm:ListCertificates",
        "cloudfront:Describe*",
        "cloudfront:Get*",
        "cloudfront:List*",
        "cloudfront-keyvaluestore:Describe*",
        "cloudfront-keyvaluestore:Get*",
        "cloudfront-keyvaluestore:List*",
        "iam:ListServerCertificates",
        "route53:List*",
        "waf:ListWebACLs",
        "waf:GetWebACL",
        "wafv2:ListWebACLs",
        "wafv2:GetWebACL"
      ],
      "Resource": "*"
    }
  ]
}
```

Política gerenciada pela AWS: CloudFrontFullAccess

Você pode anexar a política CloudFrontFullAccess a suas identidades do IAM. Essa política concede permissões administrativas aos recursos do CloudFront. Ela também concede permissões somente leitura a outros recursos de serviços da AWS que estão relacionados ao CloudFront e são visíveis no console do CloudFront.

Detalhes das permissões

Esta política inclui as seguintes permissões:

- `s3:ListAllMyBuckets`: permite que os principais obtenham uma lista de todos os buckets do Amazon S3.
- `acm:ListCertificates`: permite que os principais obtenham uma lista de certificados do ACM.

- `cloudfront:*`: permite que os principais realizem todas as ações em todos os recursos do CloudFront.
- `cloudfront-keyvaluestore:*`: permite que as entidades principais realizem todas as ações no armazenamento de chave-valor.
- `iam:ListServerCertificates`: permite que os principais obtenham uma lista de certificados de servidor armazenados no IAM.
- `waf:ListWebACLs`: permite que os principais obtenham uma lista de ACLs da Web no AWS WAF.
- `waf:GetWebACL`: permite que os principais obtenham informações detalhadas sobre ACLs da Web no AWS WAF.
- `wafv2:ListWebACLs`: permite que os principais obtenham uma lista de ACLs da Web no AWS WAF.
- `wafv2:GetWebACL`: permite que os principais obtenham informações detalhadas sobre ACLs da Web no AWS WAF.
- `kinesis:ListStreams`: permite que os principais obtenham uma lista de fluxos do Amazon Kinesis.
- `kinesis:DescribeStream`: permite que os principais obtenham informações detalhadas sobre um fluxo do Kinesis.
- `iam:ListRoles`: permite que os principais obtenham uma lista de funções no IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "cfflistbuckets",
      "Action": [
        "s3:ListAllMyBuckets"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::*"
    },
    {
      "Sid": "cfffullaccess",
      "Action": [
        "acm:ListCertificates",
        "cloudfront:*",
        "cloudfront-keyvaluestore:*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

```

    "iam:ListServerCertificates",
    "waf:ListWebACLs",
    "waf:GetWebACL",
    "wafv2:ListWebACLs",
    "wafv2:GetWebACL",
    "kinesis:ListStreams"
  ],
  "Effect": "Allow",
  "Resource": "*"
},
{
  "Sid": "cffdescribestream",
  "Action": [
    "kinesis:DescribeStream"
  ],
  "Effect": "Allow",
  "Resource": "arn:aws:kinesis:*:*:*"
},
{
  "Sid": "cfflistroles",
  "Action": [
    "iam:ListRoles"
  ],
  "Effect": "Allow",
  "Resource": "arn:aws:iam:*:*:*"
}
]
}

```

Important

Para que o CloudFront crie e salve logs de acesso, você precisa conceder outras permissões. Para ter mais informações, consulte [Permissões necessárias para configurar o registro em log padrão e acessar os arquivos de log](#).

Política gerenciada pela AWS: AWSCloudFrontLogger

Não é possível anexar a política AWSCloudFrontLogger a suas identidades do IAM. Essa política é anexada a uma função vinculada ao serviço que permite que o CloudFront realize ações em seu nome. Para ter mais informações, consulte [the section called “Funções vinculadas ao serviço para o Lambda@Edge”](#).

Essa política permite que o CloudFront envie arquivos de log para o Amazon CloudWatch. Para obter detalhes sobre as permissões incluídas nesta política, consulte [the section called “Permissões de função vinculada ao serviço para o CloudFront Logger”](#).

Política gerenciada pela AWS: AWSLambdaReplicator

Não é possível anexar a política AWSLambdaReplicator a suas identidades do IAM. Essa política é anexada a uma função vinculada ao serviço que permite que o CloudFront realize ações em seu nome. Para ter mais informações, consulte [the section called “Funções vinculadas ao serviço para o Lambda@Edge”](#).

Esta política permite que o CloudFront crie, exclua e desabilite funções no AWS Lambda para replicar funções do Lambda@Edge para Regiões da AWS. Para obter detalhes sobre as permissões incluídas nesta política, consulte [the section called “Permissões de função vinculada ao serviço para o replicador do Lambda”](#).

Atualizações do CloudFront em políticas gerenciadas pela AWS

Veja detalhes sobre atualizações em políticas gerenciadas pela AWS para o CloudFront desde que esse serviço começou a monitorar essas alterações. Para obter alertas automáticos sobre alterações feitas nesta página, inscreva-se no feed RSS na página [Document history](#) (Histórico de documentos) do CloudFront.

Alteração	Descrição	Data
CloudFrontReadOnlyAccess e CloudFrontFullAccess : atualização em duas políticas existentes.	O CloudFront adicionou novas permissões para armazenamentos de chave-valor. As novas permissões permitem que os usuários obtenham informações e ajam em armazenamentos de chave-valor.	19 de dezembro de 2023
CloudFrontReadOnlyAccess : atualização em uma política existente	O CloudFront adicionou uma nova permissão para descrever as CloudFront Functions.	8 de setembro de 2021

Alteração	Descrição	Data
	Ela permite que o usuário, grupo ou perfil leia informações e metadados sobre uma função, mas não o código da função.	
CloudFront começa a monitorar alterações	CloudFront começa a monitorar alterações para suas políticas gerenciadas pela AWS.	8 de setembro de 2021

Solução de problemas de identidade e acesso do Amazon CloudFront

Use as informações a seguir para ajudar a diagnosticar e corrigir problemas comuns que você possa encontrar ao trabalhar com o CloudFront e o IAM.

Tópicos

- [Não tenho autorização para executar uma ação no CloudFront](#)
- [Não estou autorizado a executar iam:PassRole](#)
- [Quero permitir que as pessoas fora da minha Conta da AWS acessem meus recursos do CloudFront](#)

Não tenho autorização para executar uma ação no CloudFront

Se você receber uma mensagem de erro informando que não tem autorização para executar uma ação, suas políticas deverão ser atualizadas para permitir que você realize a ação.

O erro do exemplo a seguir ocorre quando o usuário do IAM mateojackson tenta usar o console para visualizar detalhes sobre um atributo *my-example-widget* fictício, mas não tem as permissões `cloudfront:GetWidget` fictícias.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
cloudfront:GetWidget on resource: my-example-widget
```

Nesse caso, a política do usuário `mateojackson` deve ser atualizada para permitir o acesso ao recurso `my-example-widget` usando a ação `cloudfront:GetWidget`.

Se você precisar de ajuda, entre em contato com seu administrador AWS. Seu administrador é a pessoa que forneceu suas credenciais de login.

Não estou autorizado a executar `iam:PassRole`

Se você receber uma mensagem de erro informando que não tem autorização para realizar a ação `iam:PassRole`, as suas políticas deverão ser atualizadas para permitir que você passe um perfil para o CloudFront.

Alguns Serviços da AWS permitem que você passe um perfil existente para o serviço, em vez de criar um novo perfil de serviço ou perfil vinculado ao serviço. Para fazer isso, é preciso ter permissões para passar o perfil para o serviço.

O erro de exemplo a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta usar o console para realizar uma ação no CloudFront. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se você precisar de ajuda, entre em contato com seu administrador AWS. Seu administrador é a pessoa que forneceu suas credenciais de login.

Quero permitir que as pessoas fora da minha Conta da AWS acessem meus recursos do CloudFront

Você pode criar um perfil que os usuários de outras contas ou pessoas fora da sua organização podem usar para acessar seus recursos. Você pode especificar quem é confiável para assumir o perfil. Para serviços que oferecem compatibilidade com políticas baseadas em recursos ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se o CloudFront é compatível com esses recursos, consulte [Como o Amazon CloudFront funciona com o IAM](#).
- Para saber como conceder acesso a seus recursos em todas as Contas da AWS pertencentes a você, consulte [Fornecimento de acesso a um usuário do IAM em outra Conta da AWS pertencente a você](#) no Guia de usuário do IAM.
- Para saber como conceder acesso a seus recursos para terceiros Contas da AWS, consulte [Fornecimento de acesso a Contas da AWS pertencentes a terceiros](#) no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte [Conceder acesso a usuários autenticados externamente \(federação de identidades\)](#) no Guia do usuário do IAM.
- Para saber a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.

Registro em log e monitoramento no Amazon CloudFront

O monitoramento é uma parte importante para manter a disponibilidade e a performance do CloudFront e das suas soluções da AWS. Você deve coletar dados de monitoramento de todas as partes de sua solução da AWS para depurar uma falha de vários pontos com facilidade, caso ocorra. A AWS fornece várias ferramentas para monitorar seus recursos e suas atividades no CloudFront e responder a incidentes em potencial:

Alarmes do Amazon CloudWatch

Ao usar alarmes do CloudWatch, você observa uma única métrica durante um período especificado. Se a métrica exceder determinado limite, uma notificação será enviada para um tópico do Amazon SNS ou para uma política do AWS Auto Scaling. Os alarmes do CloudWatch não invocam ações quando uma métrica está em um estado específico. O estado deve ter sido alterado e mantido por uma quantidade especificada de períodos.

Para obter mais informações, consulte [Monitorar métricas do CloudFront com o Amazon CloudWatch](#).

Logs do AWS CloudTrail

O CloudTrail fornece um registro de ações de API realizadas por um usuário, uma função ou um serviço da AWS no CloudFront. Usando as informações coletadas pelo CloudTrail, é possível determinar a solicitação de API realizada para o CloudFront, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita, além de detalhes adicionais.

Para ter mais informações, consulte [Registrar em log chamadas de API do Amazon CloudFront usando o AWS CloudTrail](#).

Logs padrão do CloudFront e logs em tempo real

Os logs do CloudFront fornecem registros detalhados das solicitações feitas para uma distribuição. Esses logs são úteis para muitas aplicações. Por exemplo, as informações do log podem ser úteis em auditorias de segurança e acesso.

Para ter mais informações, consulte [Configurar e usar logs padrão \(logs de acesso\)](#) e [Criar e usar configurações de log em tempo real](#).

Logs de funções de borda

Os logs gerados pelas funções de borda, tanto do CloudFront Functions quanto do Lambda@Edge, são enviados diretamente para o Amazon CloudWatch Logs e não são armazenados em nenhum lugar pelo CloudFront. O CloudFront Functions usa uma [função vinculada ao serviço](#) do AWS Identity and Access Management (IAM) para enviar logs gerados pelo cliente diretamente para o CloudWatch Logs em sua conta.

Para ter mais informações, consulte [Logs de funções de borda](#).

Relatórios do CloudFront no console

O console do CloudFront inclui uma variedade de relatórios, incluindo o relatório de estatística de cache, o relatório de objetos populares, e o relatório de indicadores principais. A maioria dos relatórios do console do CloudFront são baseados nos dados nos logs de acesso do CloudFront, que contêm informações detalhadas sobre cada solicitação do usuário recebida pelo CloudFront. No entanto, não é necessário permitir que os logs de acesso visualizem os relatórios.

Para obter mais informações, consulte [Visualizar relatórios do CloudFront no console](#).

Validação de conformidade com o Amazon CloudFront

Audidores externos avaliam a segurança e a conformidade do Amazon CloudFront como parte de vários programas de conformidade da AWS. Isso inclui SOC, PCI, HIPAA e outros.

Para obter uma lista dos produtos da AWS no escopo de programas de conformidade específicos, consulte [Produtos da AWS no escopo por programa de conformidade](#). Para obter informações gerais, consulte [Programas de conformidade da AWS](#).

É possível baixar relatórios de auditoria de terceiros usando o AWS Artifact. Para obter mais informações, consulte [Fazer download dos relatórios no AWS Artifact](#).

Sua responsabilidade com relação à conformidade ao usar o CloudFront é determinada pela confidencialidade dos dados, pelos objetivos de conformidade da empresa e pelos regulamentos e leis aplicáveis. A AWS fornece os seguintes recursos para ajudar com a conformidade:

- [Guias de início rápido de segurança e conformidade](#): esses guias de implantação abordam as considerações de arquitetura e fornecem etapas para a implantação de ambientes de linha de base concentrados em conformidade e segurança na AWS.
- [Whitepaper Arquitetura para segurança e conformidade com HIPAA na AWS](#): este whitepaper descreve como as empresas podem usar a AWS para criar aplicativos que estejam em conformidade com a HIPAA.

O programa de conformidade com a HIPAA da AWS inclui o CloudFront (exceto a entrega de conteúdo por meio de POPs incorporados do CloudFront) como um serviço qualificado para a HIPAA. Se você tiver um Adendo de associado comercial (BAA) assinado com a AWS, poderá usar o CloudFront (exceto a entrega de conteúdo por meio de POPs incorporados do CloudFront) para entregar conteúdo com informações de saúde protegidas (PHI). Para obter mais informações, consulte [Conformidade com a HIPAA](#).

- [Recursos de compatibilidade da AWS](#) – Esta coleção de guias e pastas de trabalho pode ser aplicada ao seu setor e local.
- [AWS Config](#): esse serviço da AWS avalia até que ponto suas configurações de recursos atendem adequadamente às práticas internas e às diretrizes e regulamentações do setor.
- [AWS Security Hub](#): esse serviço da AWS usa controles de segurança para avaliar configurações de recursos e padrões de segurança que ajudam você a cumprir vários frameworks de conformidade. Para obter mais informações sobre como usar o Security Hub para avaliar os recursos do CloudFront, consulte [Controles do Amazon CloudFront](#) no Guia do usuário do AWS Security Hub.

Melhores práticas de conformidade do CloudFront

Esta seção fornece as melhores práticas e recomendações sobre conformidade ao usar o Amazon CloudFront para fornecer conteúdo.

Se você tiver workloads em conformidade com o PCI ou HIPAA, com base no [Modelo de responsabilidade compartilhada da AWS](#), recomendamos registrar em log os dados de uso do CloudFront dos últimos 365 dias para fins de auditoria futura. Para registrar dados de uso:

- Habilitar logs de acesso do CloudFront Para ter mais informações, consulte [Configurar e usar logs padrão \(logs de acesso\)](#).
- Solicitações de captura que são enviadas à API do CloudFront. Para ter mais informações, consulte [Registrar em log chamadas de API do Amazon CloudFront usando o AWS CloudTrail](#).

Além disso, consulte as seções a seguir para obter detalhes sobre como o CloudFront está em conformidade com os padrões PCI DSS e SOC.

Padrão de segurança de dados do setor de cartão de pagamento (PCI DSS – Payment Card Industry Data Security Standard)

O CloudFront (exceto a entrega de conteúdo por meio de POPs incorporados do CloudFront) é compatível com o processamento, o armazenamento e transmissão de dados de cartão de crédito por um comerciante ou provedor de serviços, e foi validado como em conformidade com o Data Security Standard (DSS, Padrão de segurança de dados) da Payment Card Industry (PCI, Padrão de cartão de crédito). Para obter mais informações sobre o PCI DSS, incluindo como solicitar uma cópia do pacote de conformidade com o PCI da AWS, consulte [Nível 1 do PCI DSS](#).

Como uma prática recomendada de segurança, recomendamos não armazenar informações de cartão de crédito em pontos de presença de caches de borda do CloudFront. Por exemplo, você pode configurar a origem para incluir um cabeçalho `Cache-Control: no-cache="field-name"` em respostas que contenham informações de cartão de crédito, como os últimos quatro dígitos do número do cartão de crédito e as informações de contato do proprietário do cartão.

Controles do Sistema e da Organização (CSO)

O CloudFront (exceto a entrega de conteúdo por meio de POPs incorporados do CloudFront) é compatível com medidas de controles de organizações e sistemas (SOC), incluindo SOC 1, SOC 2 e SOC 3. Os relatórios SOC são relatórios de exame terceiros e independentes que demonstram como a AWS satisfaz os principais controles e objetivos de conformidade. Essas auditorias garantem que os procedimentos e as proteções devidos sejam estabelecidos para minimizar os riscos que podem afetar a segurança, a confidencialidade, e a disponibilidade dos dados dos clientes e da empresa. Os resultados dessas auditorias de terceiros são disponibilizados no [Site de conformidade com o](#)

[SOC da AWS](#), onde é possível ver os relatórios publicados para obter mais informações sobre os controles que oferecem suporte às operações e conformidade da AWS.

Resiliência no Amazon CloudFront

A infraestrutura global da AWS é criada com base em regiões da AWS e zonas de disponibilidade. As regiões da AWS fornecem várias zonas de disponibilidade separadas e isoladas fisicamente, as quais são conectadas com baixa latência, alto throughput e redes altamente redundantes. Com as zonas de disponibilidade, você pode projetar e operar aplicações e bancos de dados que executam o failover automaticamente entre as zonas de disponibilidade sem interrupção. As zonas de disponibilidade são mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para mais informações sobre regiões e zonas de disponibilidade da AWS, consulte [Infraestrutura global da AWS](#).

Failover da origem do CloudFront

Além do suporte da infraestrutura global da AWS, o Amazon CloudFront oferece um recurso de failover de origem para ajudar a dar suporte às suas necessidades de resiliência de dados. O CloudFront é um serviço global que fornece seu conteúdo por meio de uma rede mundial de datacenters chamada pontos de presença ou locais de borda (POPs). Se o conteúdo ainda não estiver armazenado em cache em um ponto de presença, o CloudFront o recupera de uma origem que você identificou como a fonte da versão definitiva do conteúdo.

É possível melhorar a resiliência e aumentar a disponibilidade para cenários específicos configurando o CloudFront com failover de origem. Para começar, crie um grupo de origem no qual designar uma origem primária para o CloudFront mais uma segunda origem. O CloudFront alterna automaticamente para a segunda origem quando a origem primária retornar respostas de falha de código de status HTTP específicas. Para ter mais informações, consulte [Otimizar a alta disponibilidade com o failover de origem do CloudFront](#).

Segurança da infraestrutura no Amazon CloudFront

Como um serviço gerenciado, o Amazon CloudFront é protegido pela segurança da rede global da AWS. Para obter informações sobre serviços de segurança da AWS e como a AWS protege a infraestrutura, consulte [Segurança na Nuvem AWS](#). Para projetar seu ambiente da AWS usando as

práticas recomendadas de segurança da infraestrutura, consulte [Proteção de Infraestrutura](#) em Pilar de Segurança: AWS Well-Architected Framework.

Você usa as chamadas de API da AWS para acessar o CloudFront por meio da rede. Os clientes devem oferecer suporte para:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com perfect forward secrecy (PFS) como DHE (Ephemeral Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Além disso, as solicitações devem ser assinadas usando um ID da chave de acesso e uma chave de acesso secreta associada a uma entidade principal do IAM. Ou você pode usar o [AWS Security Token Service](#) (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

O CloudFront Functions usa uma barreira de isolamento altamente segura entre as contas da AWS, garantindo que os ambientes do cliente estejam seguros contra ataques de canal lateral, como Spectre e Meltdown. O Functions não consegue acessar ou modificar dados que pertençam a outros clientes. O Functions é executado em um processo de encadeamento único dedicado em uma CPU dedicada sem hyperthreading. Em qualquer ponto de presença (POP) do local da borda do CloudFront, o CloudFront Functions atende apenas a um cliente por vez e todos os dados específicos do cliente são limpos entre execuções de funções.

Solução de problemas

Use esta seção para solucionar problemas comuns que você pode encontrar ao configurar o Amazon CloudFront para distribuir conteúdo.

Cada tópico fornece orientação detalhada sobre como identificar a causa-raiz de problemas comuns e instruções passo a passo para resolvê-los.

Tópicos

- [Como solucionar problemas de distribuição](#)
- [Solucionar problemas de códigos de status de resposta de erros no CloudFront](#)
- [Testes de carga do CloudFront](#)

Como solucionar problemas de distribuição

Use as informações fornecidas aqui para ajudar a diagnosticar e corrigir erros de certificado, problemas de acesso negado ou outros problemas comuns que você possa encontrar ao configurar seu site ou aplicação com distribuições do Amazon CloudFront.

Tópicos

- [O CloudFront retorna um erro Access Denied](#)
- [O CloudFront retorna um erro InvalidViewerCertificate quando tento adicionar um nome de domínio alternativo](#)
- [O CloudFront retorna um erro de registro DNS configurado incorretamente quando tento adicionar um novo CNAME](#)
- [Não consigo visualizar os arquivos na minha distribuição](#)
- [Mensagem de erro: Certificate: <certificate-id> Is Being Used by CloudFront \(O certificado: <certificate-id> está sendo usado pelo CloudFront\)](#)

O CloudFront retorna um erro Access Denied

Se estiver usando um bucket do Amazon S3 como a origem de sua distribuição do CloudFront, você poderá ver uma mensagem de erro de acesso negado (403) nos exemplos a seguir.

Sumário

- [Você especificou um objeto ausente da origem do Amazon S3](#)
- [A origem do Amazon S3 não tem permissões do IAM](#)
- [Você está usando credenciais inválidas ou não tem permissões suficientes](#)

Você especificou um objeto ausente da origem do Amazon S3

Verifique se o objeto solicitado no bucket existe. Os nomes de objeto diferenciam letras maiúsculas de minúsculas. A inserção de um nome de objeto inválido pode retornar um código de erro de acesso negado.

Por exemplo, se você seguir o [tutorial do CloudFront](#) para criar uma distribuição básica, deverá criar um bucket do Amazon S3 como origem e carregar um arquivo `index.html` de exemplo.

No navegador da web, se inserir `https://d111111abcdef8.cloudfront.net/INDEX.HTML` em vez de `https://d111111abcdef8.cloudfront.net/index.html`, você poderá ver uma mensagem semelhante porque o arquivo `index.html` no caminho do URL diferencia letras maiúsculas de minúsculas.

```
<Error>
<Code>AccessDenied</Code>
<Message>Access Denied</Message>
<RequestId>22Q367AHT7Y1ABCD</RequestId>
<HostId>
ABCDE/Vg+7PSNa/d/IffQ8Fb92TGQ0KH0ZwG5iEKbc6+e06DdMS1ZW+ryB9GFRIVtS66rSSy6So=
</HostId>
</Error>
```

A origem do Amazon S3 não tem permissões do IAM

Verifique se você selecionou o bucket do Amazon S3 como o nome e o domínio de origem. A origem (Amazon S3) deve ter as permissões corretas.

Se você não especificar as permissões corretas, a mensagem de acesso negado a seguir poderá aparecer para seus visualizadores.

```
<Code>AccessDenied</Code>
<Message>User: arn:aws:sts::856369053181:assumed-role/OriginAccessControlRole/
EdgeCredentialsProxy+EdgeHostAuthenticationClient is not authorized to perform:
kms:Decrypt on the resource associated with this ciphertext because the resource does
```

```
not exist in this Region, no resource-based policies allow access, or a resource-based
policy explicitly denies access</Message>
<RequestId>22Q367AHT7Y1ABCD/RequestId>
<HostId>
ABCDE/Vg+7PSNa/d/IffQ8Fb92TGQ0KH0ZwG5iEKbc6+e06DdMS1ZW+ryB9GFRIVtS66rSSy6So=
</HostId>
</Error>
```

Note

Nessa mensagem de erro, o ID da conta 856369053181 é uma conta gerenciada pela AWS.

Ao distribuir conteúdo do Amazon S3 e também ao usar a criptografia do lado do serviço do AWS Key Management Service (AWS KMS) (SSE-KMS), você precisa especificar permissões adicionais do IAM para a chave do KMS e o bucket do Amazon S3. A distribuição do CloudFront precisa dessas permissões para usar a chave do KMS, que é usada para criptografia do bucket do Amazon S3 de origem.

As configurações da política de bucket do Amazon S3 permitem que a distribuição do CloudFront recupere os objetos criptografados para entrega de conteúdo.

Como verificar as permissões de chave do KMS e bucket do Amazon S3

1. Verifique se a chave do KMS que você está usando é a mesma que o bucket do Amazon S3 usa para criptografia padrão. Consulte mais informações em [Especificando criptografia no lado do servidor com o AWS KMS \(SSE-KMS\)](#) no Guia do usuário do Amazon Simple Storage Service.
2. Verifique se os objetos no bucket estão criptografados com a mesma chave do KMS. É possível selecionar qualquer objeto do bucket do Amazon S3 e verificar as configurações de criptografia do lado do servidor para verificar o ARN da chave do KMS.
3. Edite a política de bucket do Amazon S3 a fim de conceder ao CloudFront permissão para chamar a operação de API `GetObject` do bucket do Amazon S3. Consulte um exemplo de política de bucket do Amazon S3 que usa controle de acesso à origem em [Conceder permissão ao controle de acesso à origem para acessar o bucket do S3](#).
4. Edite a política de chave do KMS a fim de conceder permissão ao CloudFront para realizar as ações referentes a `Encrypt`, `Decrypt` e `GenerateDataKey*`. Para se alinhar com a permissão de privilégio mínimo, especifique um elemento `Condition`. Dessa forma, somente a distribuição especificada do CloudFront poderá realizar as ações listadas. É possível

personalizar a política para a política do AWS KMS existente. Consulte um exemplo da política de chave do KMS em [SSE-KMS](#).

Se estiver usando identidade do acesso de origem (OAI) em vez de OAC, as permissões para o bucket do Amazon S3 serão um pouco diferentes, já que você concede permissão a uma identidade, e não a um AWS service (Serviço da AWS). Para ter mais informações, consulte [Conceder permissão a uma identidade de acesso à origem para ler arquivos no bucket do Amazon S3](#).

Se você ainda não conseguir visualizar os arquivos na distribuição, consulte [Não consigo visualizar os arquivos na minha distribuição](#).

Você está usando credenciais inválidas ou não tem permissões suficientes

Uma mensagem de erro de acesso negado poderá aparecer se você estiver usando credenciais do AWS SCT incorretas ou expiradas (chave de acesso e chave secreta) ou se o perfil ou usuário do IAM não tiver a permissão necessária para realizar uma ação em um recurso do CloudFront. Consulte mais informações sobre mensagens de erro de acesso negado em [Solução de problemas de mensagens de erro de acesso negado](#) no Guia do usuário do IAM.

Consulte mais informações sobre como o IAM funciona com o CloudFront em [Identity and Access Management para Amazon CloudFront](#).

O CloudFront retorna um erro InvalidViewerCertificate quando tento adicionar um nome de domínio alternativo

Se o CloudFront retornar um erro `InvalidViewerCertificate` quando você tentar adicionar um nome de domínio alternativo (CNAME) à distribuição, revise as informações a seguir para ajudar a solucionar o problema. Esse erro pode indicar que um dos seguintes problemas devem ser resolvidos para que você possa adicionar o nome de domínio alternativo com êxito.

Os erros a seguir são listados na ordem em que o CloudFront verifica a autorização para adicionar um nome de domínio alternativo. Isso pode ajudar a solucionar problemas, pois, de acordo com o erro que o CloudFront retornar, você poderá ver quais verificações foram concluídas com êxito.

Não há nenhum certificado anexado à distribuição.

Para adicionar um nome de domínio alternativo (CNAME), você deve anexar um certificado válido e confiável à distribuição. Revise os requisitos, obtenha um certificado válido que os atenda,

anexe-o à distribuição e tente novamente. Para obter mais informações, consulte [Requisitos para o uso de nomes de domínio alternativos](#).

Há muitos certificados na cadeia de certificados para o certificado que foi anexado.

Você só pode ter até cinco certificados em uma cadeia de certificados. Reduza o número de certificados na cadeia e tente novamente.

A cadeia de certificados inclui um ou mais certificados que não são válidos para a data atual.

A cadeia de certificados para um certificado que foi adicionado tem um ou mais certificados que não são válidos, ou porque ainda não é um certificado válido ou porque um certificado expirou. Verifique os campos Not Valid Before (Não válido antes de) e Not Valid After (Não válido depois de) nos certificados da cadeia de certificados para garantir que todos eles sejam válidos de acordo com as datas listadas.

O certificado anexado não está assinado por uma autoridade de certificação (CA) confiável.

O certificado anexado ao CloudFront para verificar um nome de domínio alternativo não pode ser um certificado autoassinado. Ele deve ser assinado por uma CA confiável. Para obter mais informações, consulte [Requisitos para o uso de nomes de domínio alternativos](#).

O certificado anexado não está formatado corretamente

O nome de domínio e o formato do endereço IP que estão incluídos no certificado e o formato do próprio certificado devem seguir o padrão de certificados.

Ocorreu um erro interno do CloudFront.

O CloudFront foi bloqueado por um problema interno e não pôde validar os certificados. Nesse caso, o CloudFront retorna um código de status HTTP 500 e indica que há um problema interno do CloudFront com a anexação do certificado. Aguarde alguns minutos e tente novamente para adicionar o nome de domínio alternativo com o certificado.

O certificado anexado não abrange o nome de domínio alternativo que você está tentando adicionar.

Para cada nome de domínio alternativo que você adiciona, o CloudFront requer a anexação de um certificado SSL/TLS de uma autoridade de certificação (CA) confiável que abranja o nome de domínio, para validar sua autorização para usá-lo. Atualize o certificado para incluir um nome de domínio que abranja o CNAME que você está tentando adicionar. Para obter mais informações e exemplos de como usar nomes de domínio com caracteres curingas, consulte [Requisitos para o uso de nomes de domínio alternativos](#).

O CloudFront retorna um erro de registro DNS configurado incorretamente quando tento adicionar um novo CNAME

Quando há uma entrada de DNS curinga existente apontando para uma distribuição do CloudFront, se tentar adicionar um novo CNAME com um nome mais específico, você poderá encontrar o seguinte erro:

```
One or more aliases specified for the distribution includes an incorrectly configured DNS record that points to another CloudFront distribution. You must update the DNS record to correct the problem.
```

Esse erro ocorre porque o CloudFront consulta o DNS em relação ao CNAME e a entrada de DNS curinga é resolvida para outra distribuição.

Para resolver isso, primeiro crie outra distribuição e, depois, crie uma entrada de DNS apontando para a nova distribuição. Por fim, adicione o CNAME mais específico. Consulte mais informações sobre como adicionar CNAMEs em [Adicionar um nome de domínio alternativo](#).

Não consigo visualizar os arquivos na minha distribuição

Se você não puder visualizar os arquivos na distribuição do CloudFront, consulte os tópicos a seguir para obter algumas soluções comuns.

Você se cadastrou no CloudFront e no Amazon S3?

Para usar o Amazon CloudFront com uma origem do Amazon S3, cadastre-se no CloudFront e no Amazon S3 separadamente. Para obter mais informações sobre como se cadastrar no CloudFront e no Amazon S3, consulte [Configurar a Conta da AWS](#).

Suas permissões do bucket e do objeto do Amazon S3 estão definidas corretamente?

Se você estiver usando o CloudFront com uma origem do Amazon S3, as versões originais do seu conteúdo serão armazenadas em um bucket do S3. A maneira mais fácil de usar o CloudFront com o Amazon S3 é tornar todos os seus objetos legíveis publicamente no Amazon S3. Para fazer isso, você deve habilitar de forma explícita os privilégios de leitura pública em cada objeto carregado no Amazon S3.

Caso seu conteúdo não possa ser lido publicamente, você deverá criar um controle de acesso à origem (OAC) do CloudFront para que o CloudFront possa acessar o conteúdo. Para obter

mais informações sobre o controle de acesso à origem do CloudFront, consulte [the section called “Restringir o acesso a uma origem do Amazon Simple Storage Service”](#).

As propriedades do bucket e do objeto são independentes. Você deve conceder privilégios explicitamente a cada objeto no Amazon S3. Os objetos não herdam as propriedades dos buckets, e as propriedades dos objetos devem ser definidas de forma independente do bucket.

Seu nome de domínio alternativo (CNAME) está configurado corretamente?

Se você já tem um registro CNAME para seu nome de domínio, atualize-o ou substitua-o por um novo que aponte para o nome do domínio da sua distribuição.

Além disso, verifique se o CNAME aponta para o nome do domínio da distribuição, não para o bucket do Amazon S3. Você pode confirmar se o registro CNAME do seu sistema DNS aponta para o nome do domínio da sua distribuição. Para fazer isso, use uma ferramenta de DNS, como o dig.

O exemplo a seguir mostra uma solicitação do DIG de um nome de domínio `images.example.com` e a parte relevante do resultado. Em `ANSWER SECTION`, consulte a linha que contém CNAME. O registro CNAME do nome de domínio estará configurado corretamente se o valor à direita do CNAME for o nome de domínio da distribuição do CloudFront. Se ele for o bucket do servidor de origem do Amazon S3 ou outro nome de domínio, o registro CNAME estará configurado incorretamente.

```
[prompt]> dig images.example.com

; <<> DiG 9.3.3rc2 <<> images.example.com
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15917
;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 2, ADDITIONAL: 0
;; QUESTION SECTION:
;images.example.com.    IN  A
;; ANSWER SECTION:
images.example.com. 10800 IN CNAME d111111abcdef8.cloudfront.net.
...
...
```

Para obter mais informações sobre os CNAMEs, consulte [Usar URLs personalizados adicionando nomes de domínio alternativos \(CNAMEs\)](#).

Você está usando o URL correto na distribuição do CloudFront?

Verifique se o URL mencionado usa o nome de domínio (ou CNAME) da sua distribuição do CloudFront, não do bucket do Amazon S3 ou da origem personalizada.

Você precisa de ajuda para solucionar um problema com a origem personalizada?

Se você precisar que a AWS ajude a solucionar um problema com uma origem personalizada, é possível que precisemos inspecionar as entradas do cabeçalho X-Amz-Cf-Id das suas solicitações. Se você ainda não estiver registrando essas entradas, considere fazer isso no futuro. Para ter mais informações, consulte [the section called “Usar o Amazon EC2 \(ou outra origem personalizada\)”](#). Para obter ajuda adicional, consulte a [Central de Suporte da AWS](#).

Mensagem de erro: Certificate: <certificate-id> Is Being Used by CloudFront (O certificado: <certificate-id> está sendo usado pelo CloudFront)

Problema: você está tentando excluir um certificado SSL/TLS do armazenamento de certificados do IAM e está recebendo a mensagem "Certificate: <certificate-id> is being used by CloudFront (O certificado: certificate-id) está sendo usado pelo CloudFront)".

Solução: cada distribuição do CloudFront deve ser associada ao certificado padrão do CloudFront ou a um certificado SSL/TLS personalizado. Antes de excluir um certificado SSL/TLS, você deve alternar o certificado (substituir o certificado SSL/TLS personalizado atual por outro) ou reverter do uso de um certificado SSL/TLS personalizado para o uso do certificado padrão do CloudFront. Para corrigir isso, conclua as etapas de um dos seguintes procedimentos:

- [Trocar certificados SSL/TLS](#)
- [Reverter um certificado SSL/TLS personalizado para o certificado padrão do CloudFront](#)

Solucionar problemas de códigos de status de resposta de erros no CloudFront

Se o CloudFront solicitar um objeto de sua origem, e a origem retornar um código de status HTTP 4xx ou 5xx, haverá um problema com a comunicação entre o CloudFront e a origem.

Este tópico também inclui etapas de solução de problemas para esses códigos de status ao usar o Lambda@Edge ou o CloudFront Functions.

Os tópicos a seguir fornecem explicações detalhadas das possíveis causas dessas respostas de erro e oferecem orientação passo a passo sobre como diagnosticar e resolver os problemas subjacentes.

Tópicos

- [Código de status HTTP 400 \(solicitação inválida\)](#)
- [Código de status HTTP 401 \(Não autorizado\)](#)
- [Código de status HTTP 403 \(Permissão negada\)](#)
- [Código de status HTTP 404 \(Não encontrado\)](#)
- [Código de status HTTP 405 \(Método não permitido\)](#)
- [Código de status HTTP 412 \(Falha na pré-condição\)](#)
- [Código de status HTTP 500 \(Erro interno do servidor\)](#)
- [Código de status HTTP 502 \(Gateway inválido\)](#)
- [Código de status HTTP 503 \(Serviço indisponível\)](#)
- [Código de status HTTP 504 \(tempo limite do gateway\)](#)

Código de status HTTP 400 (solicitação inválida)

O CloudFront exibe o erro “400 Solicitação inválida” quando o cliente envia alguns dados inválidos na solicitação, como conteúdo ausente ou incorreto na carga útil ou nos parâmetros. Isso também pode representar um erro genérico do cliente.

A origem do Amazon S3 exibe um erro 400

Se você estiver usando uma origem do Amazon S3 com a distribuição do CloudFront, sua distribuição poderá enviar respostas com o código de status HTTP “400 Solicitação inválida” e uma mensagem semelhante à seguinte:

```
The authorization header is malformed; the region '<AWS Region>' is wrong; expecting '<AWS Region>'
```

Por exemplo:

```
The authorization header is malformed; the region 'us-east-1' is wrong; expecting 'us-west-2'
```

Esse problema pode ocorrer no seguinte cenário:

1. A origem da distribuição do CloudFront é um bucket do Amazon S3.

2. Você moveu o bucket do S3 de uma região da AWS para outra. Ou seja, você excluiu o bucket do S3 e, posteriormente, criou um novo bucket com o mesmo nome de bucket, mas em uma região da AWS diferente de onde o bucket original do S3 estava localizado.

Para corrigir esse erro, atualize a distribuição do CloudFront para que ele localize o bucket do S3 na região da AWS atual do bucket.

Como atualizar a distribuição do CloudFront

1. Faça login no AWS Management Console e abra o console do CloudFront em <https://console.aws.amazon.com/cloudfront/v4/home>.
2. Escolha a distribuição que produz esse erro.
3. Escolha Origins and Origin Groups (Origens e Grupos de Origem).
4. Localize a origem do bucket do S3 que você moveu. Marque a caixa de seleção ao lado dessa origem e escolha Edit (Editar).
5. Escolha Yes, Edit. Você não precisa alterar nenhuma configuração antes de escolher Yes, Edit (Sim, editar).

Ao concluir essas etapas, o CloudFront reimplantar a sua distribuição. Enquanto a distribuição estiver sendo implantada, você verá o status Implantando na coluna Última modificação. Algum tempo depois que a implantação estiver concluída, você deverá parar de receber as respostas de erro `AuthorizationHeaderMalformed`.

A origem do Application Load Balancer exibe um erro 400

Se estiver usando uma origem do Application Load Balancer com a distribuição do CloudFront, as possíveis causas do erro 400 são as seguintes:

- O cliente enviou uma solicitação malformada que não atende às especificações de HTTP.
- O cabeçalho de solicitação excedeu 16 K por linha de solicitação, 16 K por cabeçalho único ou 64 K para o cabeçalho da solicitação inteira.
- O cliente fechou a conexão antes de enviar o corpo completo da solicitação.

Código de status HTTP 401 (Não autorizado)

Um código de status de resposta “401 Não autorizado” indica que a solicitação do cliente não foi concluída porque não tem credenciais de autenticação válidas para o recurso solicitado. Esse código de status é enviado com um cabeçalho de resposta WWW-Authenticate HTTP que contém informações sobre como o cliente pode solicitar o recurso novamente depois de solicitar as credenciais de autenticação do usuário. Para ter mais informações, consulte [401 Não autorizado](#).

No CloudFront, quando sua origem espera que um cabeçalho `Authorization` autentique as solicitações, o CloudFront precisa encaminhar o cabeçalho `Authorization` à origem para evitar um erro “401 Não autorizado”. Quando o CloudFront encaminha uma solicitação de visualizador para sua origem, o CloudFront remove alguns cabeçalhos de visualizador por padrão, incluindo o `Authorization` cabeçalho. Para garantir que sua origem sempre receba o `Authorization` cabeçalho em solicitações de origem, você tem as seguintes opções:

- Adicione o cabeçalho `Authorization` à chave de cache usando uma política de cache. Todos os cabeçalhos na chave de cache são automaticamente incluídos nas solicitações de origem. Para ter mais informações, consulte [Controlar a chave de cache com uma política](#).
- Use uma política de solicitação de origem que encaminha todos os cabeçalhos do visualizador para a origem. Não é possível encaminhar o cabeçalho `Authorization` individualmente em uma política de solicitação de origem, mas ao encaminhar todos os cabeçalhos do visualizador, o CloudFront inclui o cabeçalho `Authorization` nas solicitações do visualizador. O CloudFront fornece a política gerenciada de solicitação de origem `AllViewer` para esse caso de uso. Para ter mais informações, consulte [Usar políticas de solicitação de origem gerenciadas](#).

Para ter mais informações, consulte [Como configuro o CloudFront de forma que o cabeçalho `Authorization` seja encaminhado para a origem?](#).

Código de status HTTP 403 (Permissão negada)

Um erro HTTP 403 significa que o cliente não está autorizado a acessar o recurso solicitado. O cliente entende a solicitação, mas não pode autorizar o acesso do visualizador. As causas comuns para o CloudFront exibir esse código de status são as seguintes:

Tópicos

- [O CNAME alternativo está configurado incorretamente.](#)
- [O AWS WAF está configurado na distribuição do CloudFront ou na origem.](#)

- [A origem personalizada exibe um erro 403](#)
- [A origem do Amazon S3 exibe um erro 403](#)
- [Restrições geográficas exibem um erro 403](#)
- [URL assinado ou configuração de cookie assinado exibe um erro 403](#)
- [Distribuições empilhadas causam um erro 403](#)

O CNAME alternativo está configurado incorretamente.

Verifique se você especificou o CNAME correto para nossa distribuição. Para usar um CNAME alternativo em vez do URL padrão do CloudFront:

1. Crie um registro CNAME no DNS para apontar o CNAME para o URL de distribuição do CloudFront.
2. Adicione o CNAME à configuração de distribuição do CloudFront.

Se você criar o registro DNS e não adicionar o CNAME à configuração de distribuição do CloudFront, a solicitação exibirá um erro 403. Para ter mais informações sobre como configurar um CNAME personalizado, consulte [Usar URLs personalizados adicionando nomes de domínio alternativos \(CNAMEs\)](#).

O AWS WAF está configurado na distribuição do CloudFront ou na origem.

Quando o AWS WAF está entre o cliente e o CloudFront, o CloudFront não consegue diferenciar entre um código de erro 403 exibido por sua origem e um exibido pelo AWS WAF quando uma solicitação é bloqueada.

Para encontrar a origem do código de status 403, confira a regra de lista de controle de acesso (ACL) da web do AWS WAF para ver se há uma solicitação bloqueada. Para obter mais informações, consulte os tópicos a seguir.

- [Listas de controle de acesso da web \(ACLs da web\) do AWS WAF](#)
- [Testar e ajustar as proteções do AWS WAF](#)

A origem personalizada exibe um erro 403

Se você estiver usando uma origem personalizada, poderá ver um erro 403 se tiver uma configuração de firewall personalizada na origem. Para solucionar o problema, faça a solicitação

diretamente na origem. Se você conseguir replicar o erro sem o CloudFront, isso significa que a origem está causando o erro 403.

Se a origem personalizada estiver causando o erro, confira os logs de origem para identificar o que pode estar causando o erro. Para ter mais informações, consulte os seguintes tópicos de solução de problemas:

- [Como soluciono erros HTTP 403 do API Gateway?](#)
- [Como soluciono erros proibidos de HTTP 403 do Application Load Balancer?](#)

A origem do Amazon S3 exibe um erro 403

Você pode ver um erro 403 pelos seguintes motivos:

- O CloudFront não tem acesso ao bucket do Amazon S3. Isso poderá acontecer se a identidade do acesso de origem (OAI) ou o controle do acesso de origem (OAC) não estiverem habilitados para sua distribuição e o bucket for privado.
- O caminho especificado no URL solicitado não está correto.
- O objeto solicitado não existe.
- O cabeçalho do host foi encaminhado com o endpoint da API REST. Para ter mais informações, consulte [Especificação de bucket do cabeçalho Host HTTP](#), no Guia do usuário do Amazon Simple Storage Service.
- Você configurou páginas de erro personalizadas. Para ter mais informações, consulte [Como o CloudFront processará erros quando páginas de erro personalizadas estiverem configuradas](#).

Restrições geográficas exibem um erro 403

Se você habilitou restrições geográficas (também conhecidas como bloqueios geográficos), para impedir que usuários em localizações geográficas específicas acessem o conteúdo que você está distribuindo por meio de uma distribuição do CloudFront, os usuários bloqueados receberão um erro 403.

Para ter mais informações, consulte [Restringir a distribuição geográfica do conteúdo](#).

URL assinado ou configuração de cookie assinado exibe um erro 403

Se você habilitou Restringir acesso do visualizador para a configuração de comportamento da distribuição, as solicitações que não usam cookies assinados ou URLs assinados vão gerar um erro 403. Para obter mais informações, consulte os tópicos a seguir.

- [Veicular conteúdo privado com URLs e cookies assinados](#)
- [Como soluciono problemas relacionados a um URL assinado ou cookies assinados no CloudFront?](#)

Distribuições empilhadas causam um erro 403

Se você tiver duas ou mais distribuições em uma cadeia de solicitações para o endpoint de origem, o CloudFront exibirá um erro 403. Não recomendamos colocar uma distribuição na frente da outra.

Código de status HTTP 404 (Não encontrado)

O CloudFront exibe o erro “404 (Não encontrado)” quando o cliente tenta acessar um recurso que não existe. Se você receber esse erro com a distribuição do CloudFront, as causas comuns são as seguintes:

- O recurso não existe.
- O URL está incorreto.
- A origem personalizada está exibindo um erro 404.
- Páginas de erro personalizadas estão exibindo um erro 404. (Qualquer código de erro pode ser convertido em 404.) Para ter mais informações, consulte [Como o CloudFront processará erros quando páginas de erro personalizadas estiverem configuradas](#).
- A página de erro personalizada foi excluída acidentalmente, gerando um erro 404 porque a solicitação procura a página de erro personalizada excluída. Para ter mais informações, consulte [Como o CloudFront processará erros quando páginas de erro personalizadas não estiverem configuradas](#).
- Caminho de origem incorreto. Se o caminho de origem for preenchido, o valor será anexado ao caminho de cada solicitação do navegador antes que a solicitação seja encaminhada à origem. Para ter mais informações, consulte [Caminho de origem](#).

Código de status HTTP 405 (Método não permitido)

O CloudFront exibirá o erro “405 (Método não permitido)” se você estiver tentando usar um método HTTP que não tenha especificado na distribuição do CloudFront. É possível especificar uma das seguintes opções para sua distribuição:

- O CloudFront encaminha somente as solicitações GET e HEAD.
- O CloudFront encaminha somente as solicitações GET, HEAD e OPTIONS.
- O CloudFront encaminha as solicitações GET, HEAD, OPTIONS, PUT, PATCH, POST e DELETE. (Se você selecionar essa opção, talvez seja necessário restringir o acesso ao bucket do Amazon S3 ou à origem personalizada para impedir que os usuários executem operações indesejadas.) Por exemplo, talvez não seja necessário que os usuários tenham permissões para excluir objetos de sua origem.

Código de status HTTP 412 (Falha na pré-condição)

O CloudFront exibe um código de erro “412 (Falha na pré-condição)” quando o acesso ao recurso de destino é negado. Em alguns casos, um servidor é configurado para aceitar solicitações somente depois de determinadas condições serem atendidas. Se alguma das condições especificadas não for atendida, o servidor não permitirá que o cliente acesse o recurso fornecido. Em vez disso, o servidor responderá com um código de erro 412.

As causas comuns de um erro 412 no CloudFront são:

- Solicitações condicionais em métodos diferentes de GET ou HEAD quando a condição definida pelos cabeçalhos `If-Unmodified-Since` ou `If-None-Match` não é atendida. Nesse caso, a solicitação, geralmente um upload ou a modificação de um recurso, não pode ser feita.
- Uma condição em um ou mais campos da solicitação na operação de API [UpdateDistribution](#) do CloudFront é avaliada como falsa.

Código de status HTTP 500 (Erro interno do servidor)

Um código de status HTTP “500 (Erro interno do servidor)” indica que o servidor encontrou uma condição inesperada que o impediu de atender à solicitação. Veja a seguir algumas causas comuns de erros 500 no Amazon CloudFront.

Tópicos

- [O servidor de origem exibe o erro 500 para o CloudFront](#)

O servidor de origem exibe o erro 500 para o CloudFront

O servidor de origem pode estar exibindo um erro 500 para o CloudFront. Consulte os seguintes tópicos de solução de problemas para ter mais informações:

- Se o Amazon S3 exibir um erro 500, consulte [Como solucionar um erro HTTP 500 ou 503 do Amazon S3?](#).
- Se o API Gateway exibir um erro 500, consulte [Como posso solucionar os erros 5xx da API REST do API Gateway?](#).
- Se o Elastic Load Balancing exibir um erro 500, consulte [HTTP 500: Internal server error](#) no Guia do usuário de Application Load Balancers.

Se a lista anterior não resolver o erro 500, é provável que o problema seja que um ponto de presença do CloudFront Point está exibindo um erro interno do servidor. É possível entrar em contato com o [AWS Support](#) para receber assistência.

Código de status HTTP 502 (Gateway inválido)

O CloudFront exibe um código de status HTTP “502 (Gateway inválido)” quando ele não consegue fornecer o objeto solicitado devido à impossibilidade de se conectar ao servidor de origem.

Se você estiver usando o Lambda@Edge, o problema pode ser um erro de validação do Lambda. Se você receber um erro HTTP 502 com o código de erro `NonS3OriginDnsError`, provavelmente há um problema de configuração de DNS que impede o CloudFront de se conectar à origem.

Tópicos

- [Falha de negociação SSL/TLS entre o CloudFront e um servidor de origem personalizado](#)
- [A origem não está respondendo com criptografias/protocolos compatíveis](#)
- [O certificado SSL/TLS da origem expirou, é inválido ou autoassinado, ou a cadeia de certificados está no pedido errado](#)
- [A origem não está respondendo nas portas especificadas nas configurações](#)
- [Erro de validação do Lambda](#)
- [Erro de validação da função do CloudFront](#)
- [Erro de DNS \(NonS3OriginDnsError\)](#)

- [Erro 502 de origem do Application Load Balancer](#)
- [Erro 502 de origem do API Gateway](#)

Falha de negociação SSL/TLS entre o CloudFront e um servidor de origem personalizado

Se você usar uma origem personalizada que exija HTTPS entre o CloudFront e a origem, nomes de domínio não correspondentes poderão causar erros. O certificado SSL/TLS em sua origem deve incluir um nome de domínio que corresponda ao domínio de origem que você especificou para a distribuição do CloudFront ou ao cabeçalho Host da solicitação de origem.

Se os nomes de domínio não coincidirem, o handshake SSL/TLS falhará e o CloudFront retornará um código de status HTTP 502 (gateway inválido) e definirá o cabeçalho X-Cache como `Error from cloudfront`.

Para determinar se os nomes de domínio no certificado correspondem ao Domínio de origem da distribuição ou ao cabeçalho Host, você pode usar um verificador SSL on-line ou o OpenSSL. Se os nomes de domínio não forem correspondentes, você tem duas opções:

- Obtenha um novo certificado SSL/TLS que inclua os nomes de domínio aplicáveis.

Se você usar o AWS Certificate Manager (ACM), consulte [Solicitar um certificado](#) no Guia do usuário do AWS Certificate Manager para solicitar um novo certificado.

- Altere a configuração da distribuição para que o CloudFront não tente mais usar SSL para conectar-se à origem.

Verificador SSL online

Para encontrar uma ferramenta de teste de SSL, pesquise na Internet por "verificador ssl online". Normalmente, você especifica o nome do seu domínio, e a ferramenta retorna uma variedade de informações sobre seu certificado SSL/TLS. Confirme se o certificado contém seu nome de domínio nos campos Nome comum ou Nomes alternativos da entidade.

OpenSSL

Para ajudar a solucionar problemas de erros HTTP 502 no CloudFront, você pode usar o OpenSSL para tentar fazer uma conexão SSL/TLS com o servidor de origem. Se o OpenSSL não for capaz de fazer uma conexão, isso pode indicar um problema com a configuração de SSL/TLS do servidor de origem. Se o OpenSSL for capaz de fazer uma conexão, ele retornará informações sobre

o certificado do servidor de origem, incluindo o nome comum (campo Subject CN) e o nome alternativo da entidade (campo Subject Alternative Name) do certificado.

Use o seguinte comando OpenSSL para testar a conexão com o servidor de origem (substitua o *domínio de origem* pelo nome de domínio do servidor de origem, como exemplo.com):

```
openssl s_client -connect origin domain name:443
```

Se o seguinte for verdadeiro:

- Seu servidor de origem oferece suporte a vários nomes de domínio com vários certificados SSL/TLS
- Sua distribuição está configurada para encaminhar o cabeçalho Host para a origem

Depois adicione a opção `-servername` ao comando do OpenSSL, como no exemplo a seguir (substitua *CNAME* pelo CNAME configurado em sua distribuição):

```
openssl s_client -connect origin domain name:443 -servername CNAME
```

A origem não está respondendo com criptografias/protocolos compatíveis

O CloudFront se conecta a servidores de origem usando criptografias e protocolos. Para obter uma lista de criptografias e protocolos compatíveis com o CloudFront, consulte [the section called “Protocolos e criptografias compatíveis entre o CloudFront e a origem”](#). Caso a origem não responda com uma dessas criptografias ou protocolos na troca de SSL/TLS, ocorrerá uma falha na conexão do CloudFront. Você pode validar se a sua origem é compatível com as criptografias e os protocolos usando uma ferramenta online, como o [SSL Labs](#). Digite o nome de domínio da sua origem no campo Hostname e, em seguida, escolha Submit. Analise os campos Common names e Alternative names do teste para ver se eles são correspondentes com o nome de domínio da sua origem. Após a conclusão do teste, encontre as seções Protocols e Cipher Suites nos resultados do teste para ver quais criptografias ou protocolos são compatíveis com sua origem. Compare o conteúdo com a lista de [the section called “Protocolos e criptografias compatíveis entre o CloudFront e a origem”](#).

O certificado SSL/TLS da origem expirou, é inválido ou autoassinado, ou a cadeia de certificados está no pedido errado

Se o servidor de origem retornar o seguinte, o CloudFront interromperá a conexão TCP, retornará o código de status HTTP 502 (gateway inválido) e definirá o cabeçalho X-Cache como `Error from cloudfront`:

- Um certificado expirado
- Certificado inválido
- Certificado autoassinado
- Cadeia de certificados na ordem errada

Note

Se toda a cadeia de certificados, inclusive o certificado intermediário, não estiver presente, o CloudFront interromperá a conexão TCP.

Para obter informações sobre como instalar um certificado SSL/TLS no seu servidor de origem personalizado, consulte [the section called “Exigir HTTPS para uma origem personalizada”](#).

A origem não está respondendo nas portas especificadas nas configurações

Ao criar uma origem na distribuição do CloudFront, defina as portas de conexão do CloudFront com a origem para tráfego HTTP e HTTPS. Por padrão, são elas: TCP 80/443. Você tem a opção de modificar essas portas. Caso a origem esteja rejeitando o tráfego nessas portas por algum motivo ou o servidor de backend não esteja respondendo nas portas, ocorrerá uma falha na conexão do CloudFront.

Para solucionar esses problemas, marque todos os firewalls em execução na sua infraestrutura e valide se eles não são bloqueando os intervalos de IP compatíveis. Consulte mais informações em [Intervalos de endereços IP da AWS](#) no Guia do usuário da Amazon VPC. Além disso, verifique se o seu servidor da web está em execução na origem.

Erro de validação do Lambda

Se você estiver usando o Lambda@Edge, um código de status HTTP 502 poderá indicar que a resposta da função do Lambda foi formada incorretamente ou incluía conteúdo inválido. Para obter mais informações sobre a solução de problema de erros do Lambda@Edge, consulte [Testar e depurar as funções do Lambda@Edge](#).

Erro de validação da função do CloudFront

Se você estiver usando funções do CloudFront, um código de status HTTP 502 poderá indicar que a função do CloudFront está tentando adicionar, excluir ou alterar um cabeçalho somente para

leitura. Esse erro não é exibido durante o teste, mas aparecerá depois que você implantar a função e executar a solicitação. Para solucionar esse erro, verifique e atualize a função do CloudFront. Para ter mais informações, consulte [Atualizar funções](#).

Erro de DNS (**NonS3OriginDnsError**)

Um erro HTTP 502 com o código de erro `NonS3OriginDnsError` indica que há um problema de configuração de DNS que impede o CloudFront de se conectar à origem. Se você receber esse erro do CloudFront, verifique se a configuração de DNS da origem está correta e funcionando.

Ao receber uma solicitação de um objeto expirado ou não armazenado no cache, o CloudFront faz uma solicitação para a origem obter o objeto. Para fazer uma solicitação bem-sucedida para a origem, o CloudFront executa uma resolução de DNS no domínio da origem. Se o serviço DNS do domínio estiver com problemas, o CloudFront não poderá resolver o nome de domínio para obter o endereço IP, o que resultará em um erro HTTP 502 (`NonS3OriginDnsError`). Para corrigir esse problema, entre em contato com o seu provedor de DNS ou, se estiver utilizando o Amazon Route 53, consulte [Why can't I access my website that uses Route 53 DNS services?](#) (Por que não consigo acessar meu site que usa os serviços de DNS do Route 53?)

Para resolver ainda mais esse problema, certifique-se de que os [servidores de nome autoritativos](#) do domínio raiz da origem ou do apex de zona (como `example.com`) estejam funcionando corretamente. Você pode usar os seguintes comandos para encontrar os servidores de nome da origem do apex com uma ferramenta como [dig](#) ou [nslookup](#):

```
dig OriginAPEXDomainName NS +short
```

```
nslookup -query=NS OriginAPEXDomainName
```

Quando você tiver os nomes dos seus servidores de nome, use os seguintes comandos para consultar o nome de domínio da sua origem neles e garantir que cada um responda com uma resposta:

```
dig OriginDomainName @NameServer
```

```
nslookup OriginDomainName NameServer
```


Important

Executa essa resolução de problemas de DNS usando um computador conectado à Internet pública. Como o CloudFront resolve o domínio de origem usando DNS público na internet, é importante solucionar problemas em um contexto semelhante.

Se a origem for um subdomínio cuja autoridade DNS está delegada a um servidor de nomes diferente do domínio raiz, verifique se os registros do servidor de nomes (NS) e início de autoridade (SOA) estão configurados corretamente para o subdomínio. É possível verificar esses registros usando comandos semelhantes aos exemplos anteriores.

Para obter mais informações sobre DNS, consulte [Conceitos de Domain Name System \(DNS\)](#) na documentação do Amazon Route 53.

Erro 502 de origem do Application Load Balancer

Se você usa o Application Load Balancer como origem e recebe um erro 502, consulte [Como soluciono os erros HTTP 502 do Application Load Balancer?](#)

Erro 502 de origem do API Gateway

Se você usa o API Gateway e recebe um erro 502, consulte [Como faço para resolver erros HTTP 502 das APIs REST do API Gateway com integração de proxy do Lambda?](#)

Código de status HTTP 503 (Serviço indisponível)

Um código de status HTTP 503 (Serviço indisponível) normalmente indica um problema de performance no servidor de origem. Em casos raros, ele indica que o CloudFront não pode atender a uma solicitação temporariamente devido a uma limitação de recursos em um local da borda.

Se você estiver usando o Lambda@Edge ou o CloudFront Functions, o problema pode ser um erro de execução ou um erro de limite excedido do Lambda@Edge.

Tópicos

- [O servidor de origem não tem capacidade suficiente para oferecer suporte à taxa de solicitação](#)
- [O CloudFront provocou o erro devido a restrições de recursos no local da borda](#)
- [Erro de execução do Lambda@Edge ou do CloudFront Function](#)

- [Limite do Lambda@Edge excedido](#)

O servidor de origem não tem capacidade suficiente para oferecer suporte à taxa de solicitação

Quando um servidor de origem não está disponível ou não consegue atender às solicitações recebidas, ele exibe um código de status HTTP 503 (serviço indisponível). O CloudFront então transmite o erro de volta para o usuário. Para resolver esse problema, tente as seguintes soluções:

- Se você usa o Amazon S3 como servidor de origem:
 - É possível enviar 3.500 solicitações PUT/COPY/POST/DELETE ou 5.500 solicitações GET/HEAD por segundo por prefixo particionado do Amazon S3. Quando o Amazon S3 exibe uma resposta 503 Slow Down, isso normalmente indica uma taxa de solicitação excessiva em relação a um prefixo específico do Amazon S3.

Como as taxas de solicitação se aplicam por prefixo em um bucket do S3, os objetos devem ser distribuídos em vários prefixos. À medida que a taxa de solicitação nos prefixos aumenta de forma gradual, o Amazon S3 aumenta a escala verticalmente para lidar com as solicitações de cada um dos prefixos de maneira separada. Como resultado, a taxa geral de solicitações processada pelo bucket é um múltiplo do número de prefixos.

- Para obter mais detalhes, consulte [Otimizar a performance do Amazon S3](#), no Guia do usuário do Amazon Simple Storage Service.
- Se você usar o Elastic Load Balancing como servidor de origem:
 - Garanta que suas instâncias de back-end possam responder às verificações de integridade.
 - Garanta que o balanceador de carga e as instâncias de back-end possam lidar com a carga.

Para obter mais informações, consulte:

- [Como soluciono erros 503 exibidos ao usar o Classic Load Balancer?](#)
- [Como soluciono erros 503 \(serviço indisponível\) do Application Load Balancer?](#)
- Se você usa uma origem personalizada:
 - Examine os logs da aplicação para garantir que a origem tenha recursos suficientes, como memória, CPU e tamanho do disco.
 - Se você usa o Amazon EC2 como o backend, verifique se o tipo de instância tem os recursos apropriados para atender às solicitações recebidas. Para ter mais informações, consulte [Instance types](#), no Guia do usuário do Amazon EC2.

- Se você usa o API Gateway:
 - Esse erro está relacionado à integração de back-end quando a API do API Gateway não consegue receber uma resposta. O servidor de back-end pode estar:
 - Sobrecarregado além da capacidade e incapaz de processar novas solicitações de clientes.
 - Em manutenção temporária.
 - Para resolver esse erro, consulte os logs da aplicação do API Gateway para determinar se há algum problema com a capacidade de back-end, a integração ou qualquer outra coisa.

O CloudFront provocou o erro devido a restrições de recursos no local da borda

Você receberá esse erro se o CloudFront não puder encaminhar solicitações ao melhor local da borda seguinte disponível e, portanto, não poderá atender a uma solicitação. Esse erro é comum ao executar testes de carga na distribuição do CloudFront. Para evitar isso, siga as diretrizes de [the section called “Testes de carga do CloudFront”](#) para evitar erros 503 (capacidade excedida).

Se isso acontecer no ambiente de produção, entre em contato com o [AWS Support](#).

Erro de execução do Lambda@Edge ou do CloudFront Function

Se você estiver usando o Lambda@Edge ou o CloudFront Functions, um código de status HTTP 503 pode indicar que sua função retornou um erro de execução.

Para ter mais detalhes sobre como identificar e resolver erros do Lambda@Edge, consulte [Testar e depurar as funções do Lambda@Edge](#).

Para ter mais informações sobre como testar o CloudFront Functions, consulte [Testar funções](#).

Limite do Lambda@Edge excedido

Se você estiver usando o Lambda@Edge, um código de status HTTP 503 poderá indicar que o Lambda exibiu um erro. O erro pode ser causado por uma das seguintes situações:

- O número de execuções de função excedeu uma das cotas que o Lambda define para limitar as execuções em uma Região da AWS (execuções simultâneas ou frequência de invocação).
- A função excedeu a cota de tempo limite da função do Lambda.

Para ter mais informações sobre as cotas do Lambda@Edge, consulte [Cotas do Lambda@Edge](#). Para ter mais detalhes sobre como identificar e resolver erros do Lambda@Edge, consulte [the](#)

[section called “Testar e depurar”](#). Você também pode ver as [Cotas do serviço do Lambda](#) no Guia do desenvolvedor do AWS Lambda.

Código de status HTTP 504 (tempo limite do gateway)

Um código de status HTTP 504 (tempo limite do gateway) indica que, quando o CloudFront encaminhou uma solicitação para a origem (porque o objeto solicitado não estava no cache de borda), ocorreu uma das seguintes situações:

- A origem retornou um código de status HTTP 504 para o CloudFront.
- A origem não respondeu antes que a solicitação expirasse.

O CloudFront retornará um código de status HTTP 504 se o tráfego estiver bloqueado para a origem por um firewall ou por um grupo de segurança ou se a origem não estiver acessível na Internet. Verifique esses problemas primeiro. Então, se o acesso não for o problema, explore os atrasos das aplicações e os tempos limite do servidor para ajudar a identificar e corrigir os problemas.

Tópicos

- [Configurar o firewall no servidor de origem para permitir o tráfego do CloudFront](#)
- [Configure os grupos de segurança no servidor de origem para permitir o tráfego do CloudFront](#)
- [Torne seu servidor de origem personalizado acessível na Internet](#)
- [Encontre e corrija respostas atrasadas de aplicações no seu servidor de origem](#)

Configurar o firewall no servidor de origem para permitir o tráfego do CloudFront

Se o firewall no servidor de origem bloquear o tráfego do CloudFront, este retornará um código de status HTTP 504. Portanto, verifique se o problema não é esse antes de examinar outras possibilidades.

O método usado para determinar se isso é um problema com o firewall depende do sistema usado pelo servidor de origem:

- Se você usar um firewall IPTable em um servidor Linux, poderá pesquisar ferramentas e informações para ajudá-lo a trabalhar com IPTables.
- Se você utiliza o Firewall do Windows em um servidor Windows, consulte [Adicionar ou editar regras de firewall](#), na documentação da Microsoft.

Ao avaliar a configuração do firewall no servidor de origem, procure por firewalls ou regras de segurança que bloqueiam o tráfego dos pontos de presença do CloudFront, com base no intervalo de endereços IP publicado. Para ter mais informações, consulte [Localizações e intervalos de endereço IP dos servidores de borda do CloudFront](#).

Se o intervalo de endereços IP do CloudFront tiver permissão para se conectar ao servidor de origem, atualize as regras de segurança do servidor para incorporar as alterações. Você pode assinar um tópico do Amazon SNS e receber notificações quando o arquivo de intervalo de endereços IP for atualizado. Depois de receber a notificação, você poderá usar o código para recuperar o arquivo, analisá-lo e fazer ajustes de acordo com o seu ambiente local. Para obter mais informações, consulte [Assinar alterações de endereços IP público da AWS via Amazon SNS](#), no blog de notícias da AWS.

Configure os grupos de segurança no servidor de origem para permitir o tráfego do CloudFront

Se a origem usar o Elastic Load Balancing, revise os [grupos de segurança do ELB](#) e verifique se os grupos de segurança permitem o tráfego de entrada no CloudFront.

É possível usar o AWS Lambda para atualizar seus grupos de segurança automaticamente e permitir o tráfego de entrada do CloudFront.

Torne seu servidor de origem personalizado acessível na Internet

Se o CloudFront não puder acessar o servidor de origem personalizado porque não está disponível publicamente na Internet, o CloudFront retornará um erro HTTP 504.

Os pontos de presença do CloudFront se conectam aos servidores de origem por meio da Internet. Se a origem personalizada estiver em uma rede privada, o CloudFront não poderá acessá-la. Por causa disso, não é possível usar servidores privados, incluindo [Classic Load Balancers internos](#), como servidores de origem com o CloudFront.

Para conferir se o tráfego da Internet pode se conectar ao servidor de origem, execute os seguintes comandos (em que *OriginDomainName* é o nome de domínio do servidor):

Para tráfego de HTTP:

- `nc -zv OriginDomainName 443`
- `telnet OriginDomainName 443`

Para tráfego HTTP:

- `nc -zv OriginDomainName 80`
- `telnet OriginDomainName 80`

Encontre e corrija respostas atrasadas de aplicações no seu servidor de origem

Os tempos limite do servidor geralmente são resultado de um aplicativo que demora muito tempo para responder ou de um valor de tempo limite definido como muito baixo.

Uma solução rápida para ajudar a evitar erros HTTP 504 é definir um valor maior para o tempo limite do CloudFront para a distribuição. Mas recomendamos que você primeiro verifique se há problemas de performance e latência com o aplicativo e o servidor de origem. Em seguida, você pode definir um valor de tempo limite razoável que ajuda a evitar erros HTTP 504 e fornece boa capacidade de resposta para os usuários.

Veja a seguir uma visão geral das etapas que você pode realizar para encontrar problemas de performance e corrigi-los:

1. Meça a latência típica e de alta carga (capacidade de resposta) do seu aplicativo web.
2. Adicione outros recursos, como CPU ou memória, se necessário. Tome outras medidas para resolver problemas, como ajuste de consultas de banco de dados para acomodar cenários de alta carga.
3. Se necessário, ajuste o valor de tempo limite para sua distribuição do CloudFront.

Veja a seguir os detalhes sobre cada etapa.

Meça a latência típica e de alta carga

Para determinar se um ou mais servidores de aplicativos web de backend estão com alta latência, execute o seguinte comando "curl" do Linux em cada servidor:

```
curl -w "DNS Lookup Time: %{time_namelookup} \nConnect time: %{time_connect} \nTLS Setup: %{time_appconnect} \nRedirect Time: %{time_redirect} \nTime to first byte: %{time_starttransfer} \nTotal time: %{time_total} \n" -o /dev/null https://www.example.com/yourobject
```

Note

Se executar o Windows nos servidores, você poderá procurar e fazer download do "curl" para que o Windows execute um comando semelhante.

Conforme você mede e avalia a latência de um aplicativo executado no seu servidor, lembre-se do seguinte:

- Os valores de latência são relativos a cada aplicativo. No entanto, um tempo até o primeiro byte em milissegundos, em vez de segundos ou mais, é razoável.
- Se você medir a latência da aplicação com carga normal e ela estiver boa, os visualizadores ainda poderão se deparar com tempo limite com alta carga. Quando há alta demanda, os servidores podem ter respostas atrasadas ou nem retornar respostas. Para ajudar a evitar problemas de latência de alta carga, verifique os recursos do servidor, como leituras e gravações de CPU e memória e disco, para garantir que os servidores tenham a capacidade de escalar para alta carga.

Você pode executar o seguinte comando do Linux para verificar a memória usada pelos processos do Apache:

```
watch -n 1 "echo -n 'Apache Processes: ' && ps -C apache2 --no-headers | wc -l && free -m"
```

- A alta utilização da CPU no servidor pode reduzir significativamente a performance de uma aplicação. Se você usar uma instância do Amazon EC2 para o servidor de backend, revise as métricas do CloudWatch para o servidor verificar a utilização da CPU. Para obter mais informações, consulte o [Guia do usuário do Amazon CloudWatch](#). Ou se você estiver usando seu próprio servidor, consulte a respectiva documentação de ajuda para ter instruções sobre como conferir a utilização da CPU.
- Verifique outros problemas possíveis com cargas altas, como consultas de banco de dados que são executadas lentamente quando há grande volume de solicitações.

Adicione recursos e ajuste servidores e bancos de dados

Depois de avaliar a capacidade de resposta dos seus aplicativos e servidores, verifique se há recursos suficientes para tráfego típico e situações de carga alta:

- Se você tiver seu próprio servidor, verifique se ele tem espaço em disco, CPU e memória suficiente para lidar com solicitações do visualizador, com base na sua avaliação.
- Se você usar uma instância do Amazon EC2 como o servidor de backend, verifique se o tipo de instância tem os recursos apropriados para atender às solicitações de entrada. Para ter mais informações, consulte [Instance types](#), no Guia do usuário do Amazon EC2.

Além disso, considere as seguintes etapas para ajudar a evitar tempos limite:

- Se o valor de tempo para o primeiro byte retornado pelo comando "curl" for alto, execute etapas para melhorar a performance do seu aplicativo. Melhorar a capacidade de resposta do aplicativo ajudará a reduzir os erros de tempo limite.
- Ajuste as consultas de banco de dados para garantir que elas manipulem volumes altos de solicitações sem performance lenta.
- Configure as conexões [keep-alive \(persistente\)](#) no seu servidor de backend. Essa opção ajuda a evitar latências que ocorrem quando as conexões precisam ser restabelecidas para solicitações ou usuários subsequentes.
- Se você usar o Elastic Load Balancing como origem, as causas de um erro 504 são as seguintes:
 - O balanceador de carga não conseguiu estabelecer conexão com o destino antes do término do tempo limite de conexão (dez segundos).
 - O balanceador de carga estabelece uma conexão com o destino, mas o destino não responde antes de decorrido o período de tempo limite de inatividade.
 - A lista de controle de acesso (ACL) da rede para a sub-rede não permite tráfego dos destinos para os nós do balanceador de carga nas portas efêmeras (1024-65535).
 - O destino retorna um cabeçalho content-length maior do que o corpo da entidade. O balanceador de carga atinge o tempo limite enquanto aguarda pelos bytes ausentes.
 - O destino é uma função do Lambda e o Lambda não responde antes do término do tempo limite de conexão.

Para ter mais informações sobre como reduzir a latência, consulte [Como soluciono problemas de alta latência em meu Classic Load Balancer do ELB?](#)

- Se você usa o MediaTailor como origem, as possíveis causas de um erro 504 são as seguintes:
 - Se os URLs relativos forem mal processados, o MediaTailor poderá receber URLs malformados dos jogadores.

- Se o MediaPackage for a origem do manifesto do MediaTailor, os erros de manifesto 404 do MediaPackage poderão fazer com que o MediaTailor exiba um erro 504.
- A solicitação para o servidor de origem do MediaTailor leva mais de dois segundos para ser concluída.
- Se você usa o Amazon API Gateway como origem, uma possível causa do erro 504 é a seguinte:
 - Uma solicitação de integração demora mais do que o parâmetro de tempo limite máximo de integração da API REST do API Gateway. Para ter mais informações, consulte [Como posso solucionar erros de tempo limite da API HTTP 504 com o API Gateway?](#).

Se necessário, ajuste o valor de tempo limite do CloudFront

Se tiver avaliado e corrigido a performance lenta da aplicação, a capacidade do servidor de origem e outros problemas, mas os visualizadores ainda estiverem enfrentando erros HTTP 504, considere a possibilidade de alterar o tempo especificado no tempo limite de resposta de origem na sua distribuição. Para ter mais informações, consulte [the section called “Tempo limite de resposta \(somente origens personalizadas\)”](#).

Testes de carga do CloudFront

Os métodos tradicionais de teste de carga não funcionam bem com o CloudFront porque ele usa o DNS para balancear cargas entre pontos de presença geograficamente dispersos e em cada ponto de presença. Ao solicitar conteúdo do CloudFront, o cliente recebe uma resposta do DNS que inclui um conjunto de endereços IP. Se você testar o envio de solicitações para apenas um dos endereços IP retornados pelo DNS, estará testando apenas um pequeno subconjunto dos recursos em um ponto de presença do CloudFront, que não representa os padrões reais de tráfego com precisão. Dependendo do volume de dados solicitado, esse tipo de teste pode sobrecarregar e diminuir a performance desse pequeno subconjunto de servidores do CloudFront.

O CloudFront foi desenvolvido para ser dimensionado para visualizadores com diferentes endereços IP de cliente e resolvedores de DNS entre várias regiões geográficas. Para executar um teste de carga que avalie com precisão a performance do CloudFront, recomendamos a execução de todas estas ações:

- Envie solicitações de cliente de várias regiões geográficas.
- Configure o teste para que cada cliente faça uma solicitação de DNS independente. Cada cliente receberá um conjunto diferente de endereços IP do DNS.

- Para cada cliente que estiver fazendo solicitações, distribua as solicitações do cliente entre o conjunto de endereços IP exibidos pelo DNS. Isso garante que a carga seja distribuída em vários servidores em um local da borda do CloudFront.

Observações

- O teste de carga não é permitido em comportamentos de cache que tenham os [gatilhos de resposta do visualizador ou solicitação do visualizador](#) do Lambda@Edge.
- O teste de carga não é permitido em origens que tenham [Proteção de origem](#) habilitada.

Cotas

Você pode solicitar o aumento da cota do CloudFront usando as seguintes opções:

- É possível usar o console do Service Quotas ou a AWS Command Line Interface. Para obter mais informações, consulte os tópicos a seguir.
 - [Requesting a quota increase](#) no Guia do usuário do Service Quotas
 - [request-service-quota-increase](#) na Referência de comandos da AWS CLI
- Se uma cota do CloudFront ainda não estiver disponível no Service Quotas, use o AWS Support Center Console para criar um [caso de aumento de cota de serviço](#).

O CloudFront está sujeito às cotas a seguir.

Tópicos

- [Cotas gerais](#)
- [Cotas gerais para distribuições](#)
- [Cotas gerais para políticas](#)
- [Cotas no CloudFront Functions](#)
- [Cotas em armazenamentos de chave-valor](#)
- [Cotas do Lambda@Edge](#)
- [Cotas para certificados SSL](#)
- [Cotas para invalidações](#)
- [Cotas em grupos de chave](#)
- [Cotas para conexões do WebSocket](#)
- [Cotas para criptografia no nível de campo](#)
- [Cotas para cookies \(configurações de cache herdadas\)](#)
- [Cotas em cadeias de consulta \(configurações de cache herdadas\)](#)
- [Cotas para cabeçalhos](#)
- [Informações relacionadas](#)

Cotas gerais

Entidade	Cota padrão
Taxa de transferência de dados por distribuição	150 Gbps Solicite uma cota maior.
Solicitações por segundo por distribuição	250.000 Solicite uma cota maior.
Tags que podem ser adicionadas a uma distribuição	50 Solicite uma cota maior.
Arquivos que podem ser fornecidos por distribuição	Sem cota
Tamanho máximo de uma solicitação ou resposta de origem, com cabeçalhos e strings de consulta, mas sem incluir o corpo da solicitação	20.480 bytes
Tamanho máximo de um URL	8.192 bytes

Cotas gerais para distribuições

Entidade	Cota padrão
Alternar nomes de domínio (CNAMEs) por distribuição	100
Para obter mais informações, consulte Usar URLs personalizados adicionando nomes de domínio alternativos (CNAMEs) .	Solicite uma cota maior.
Comportamentos de cache por distribuição	25

Entidade	Cota padrão
	Solicite uma cota maior.
Tentativas de conexão por origem Para obter mais informações, consulte Tentativas de conexão.	1 a 3
Tempo limite de conexão por origem Para obter mais informações, consulte Tempo limite da conexão.	1 a 10 segundos
Distribuições por Conta da AWS Para ter mais informações, consulte Criar uma distribuição.	200 Solicite uma cota maior.
Controle de acesso às distribuições por origem	100 Solicite uma cota maior.
Distribuições dentro da cadeia de solicitações para o endpoint de origem Não recomendamos colocar uma distribuição na frente da outra. Um erro 403 é gerado quando essa cota é excedida.	2
Compactação de arquivo: variedade de tamanhos de arquivos compactados pelo CloudFront Para obter mais informações, consulte Fornecer arquivos compactados.	1.000 a 10.000.000 bytes
Tempo limite da conexão keep-alive por origem Para ter mais informações, consulte Tempo limite keep alive (somente origens personalizadas).	1 a 60 segundos Solicite uma cota maior.

Entidade	Cota padrão
<p>Tamanho máximo de arquivo armazenável em cache por resposta HTTP GET.</p> <p>Somente as respostas para um HTTP GET são armazenadas em cache. As respostas para POST ou PUT não são armazenadas em cache.</p>	50 GB
Controles de acesso de origem por Conta da AWS	100
Identidades de acesso de origem por Conta da AWS	100
	Solicite uma cota maior.
Origens por distribuição	25
	Solicite uma cota maior.
Grupos de origens por distribuição	10
	Solicite uma cota maior.
Tempo limite da resposta por origem	1 a 60 segundos
Para obter mais informações, consulte Tempo limite de resposta (somente origens personalizadas) .	Solicite uma cota maior.
Distribuições de preparação por Conta da AWS	20
Para ter mais informações, consulte the section called “Usar implantação contínua para testar com segurança as alterações” .	Solicite uma cota maior.

Cotas gerais para políticas

Entidade	Cota padrão
Políticas de cache por Conta da AWS	20 Solicite uma cota maior.
Distribuições associadas à mesma política de cache	100
Strings de consulta por política de cache	10 Solicite uma cota maior.
Cabeçalhos por política de cache	10 Solicite uma cota maior.
Cookies por política de cache	10 Solicite uma cota maior.
Comprimento total combinado de todos os nomes de cookie, cabeçalho e string de consulta em uma política de cache	1024
Políticas de solicitação de origem por Conta da AWS	20 Solicite uma cota maior.
Distribuições associadas à mesma política de solicitação de origem	100
Strings de consulta por política de solicitação de origem	10 Solicite uma cota maior.

Entidade	Cota padrão
Cabeçalhos por política de solicitação de origem	10 Solicite uma cota maior.
Cookies por política de solicitação de origem	10 Solicite uma cota maior.
Comprimento total combinado de todos os nomes de cookie, cabeçalho e string de consulta em uma política de solicitação de origem	1024
Políticas de cabeçalhos de resposta por Conta da AWS	20 Solicite uma cota maior.
Distribuições associadas à mesma política de cabeçalhos de resposta	100 Solicite uma cota maior.
Cabeçalhos personalizados por política de cabeçalhos de resposta	10 Solicite uma cota maior.
Políticas de implantação contínua por Conta da AWS	20 Solicite uma cota maior.

Cotas no CloudFront Functions

Entidade	Cota padrão
Funções por Conta da AWS	100
Tamanho máximo da função Essa cota não é ajustável. Para armazenar dados adicionais para o CloudFront Functions, crie um armazenamento de chave-valor e adicione pares de chave-valor. Para ter mais informações, consulte KeyValueStore do Amazon CloudFront .	10 KB
Memória máxima da função	2 MB
Distribuições associadas à mesma função	100

Além dessas cotas, há algumas outras restrições ao usar o CloudFront Functions. Para ter mais informações, consulte [Restrições do CloudFront Functions](#).

Cotas em armazenamentos de chave-valor

Entidade	Cota padrão
O tamanho máximo de uma chave em um par de chave-valor	512 bytes
O tamanho máximo do valor em um par de chave-valor	1 KB
Máximo de pares de chave-valor que você pode atualizar em uma única solicitação de API	Cinquenta chaves ou 3 MB de carga útil, o que for atingido primeiro
Tamanho máximo de um armazenamento de chave-valor	5 MB
Número máximo de funções às quais um único armazenamento de chave-valor pode ser associado	10

Entidade	Cota padrão
Número máximo de armazenamentos de chave-valor por função	1
Número máximo de armazenamentos de chave-valor por conta	50 Solicite uma cota maior.

Cotas do Lambda@Edge

Cotas gerais

Entidade	Cota padrão
Distribuições por Conta da AWS que podem ter funções do Lambda@Edge	500 Solicite uma cota maior.
Funções do Lambda@Edge por distribuição	100 Solicite uma cota maior.
Execuções simultâneas	1.000 (em cada Região da AWS) Solicite uma cota maior.
<div data-bbox="142 1360 264 1398">  Note </div> <div data-bbox="186 1417 1110 1549"> <p>O Lambda gerencia as cotas de simultaneidade do Lambda@Edge. Todas as funções do Lambda na Região da AWS compartilham essa cota.</p> </div>	
<p>Para obter mais informações, consulte Escalabilidade de função no Guia do desenvolvedor do AWS Lambda.</p>	
Distribuições associadas à mesma função	500

Entidade	Cota padrão
Tamanho máximo compactado de uma função do Lambda e de todas as bibliotecas incluídas	50 MB
Solicitações do Lambda@Edge por segundo (cada Região da AWS compatível)	10.000

Cotas que diferem por tipo de evento

Entidade	Eventos de solicitação e resposta do visualizador	Eventos de solicitação e resposta da origem
Tamanho da memória da função	128 MB	O mesmo que Cotas do Lambda
Tempo-limite da função. A função pode fazer chamadas de rede para recursos, como buckets do Amazon S3, tabelas do DynamoDB ou instâncias do Amazon EC2 nas Regiões da AWS.	5 segundos	30 segundos
Tamanho de uma resposta gerada por uma função do Lambda, incluindo cabeçalhos e corpo	40 KB	1 MB

Além dessas cotas, lembre-se de que há algumas outras restrições ao usar as funções do Lambda@Edge. Para obter mais informações, consulte [Restrições ao Lambda@Edge](#).

Cotas para certificados SSL

Entidade	Cota padrão
Certificados SSL por Conta da AWS ao atender a solicitações HTTPS usando endereços IP dedicados (sem cota ao atender a solicitações HTTPS usando SNI)	2

Entidade	Cota padrão
Para ter mais informações, consulte Usar HTTPS com o CloudFront .	Solicite uma cota maior .
Certificados SSL que podem ser associados a uma distribuição do CloudFront	1

Se o certificado SSL for específico para comunicação HTTPS entre visualizadores e o CloudFront, e se você tiver usado o AWS Certificate Manager (ACM) ou o armazenamento de certificados do IAM para provisionar ou importar o certificado, cotas adicionais serão aplicadas. Para ter mais informações, consulte [Cotas no uso de certificados SSL/TLS com o CloudFront \(somente HTTPS entre visualizadores e o CloudFront\)](#).

Também há cotas para o número de certificados SSL que você pode importar para o AWS Certificate Manager (ACM) ou fazer upload para o AWS Identity and Access Management (IAM). Para ter mais informações, consulte [Aumentar as cotas de certificados SSL/TLS](#).

Cotas para invalidações

Entidade	Cota padrão
Invalidação de arquivos: número máximo de arquivos permitidos em solicitações ativas de invalidação, exceto de curingas Para obter mais informações, consulte Invalidar arquivos para remover conteúdo .	3.000
Invalidação de arquivos: número máximo permitido de invalidações ativas de curingas	15
Invalidação de arquivos: número máximo de arquivos que podem ser processados por uma invalidação de curinga	Sem cota

Cotas em grupos de chave

Entidade	Cota padrão
Chaves públicas em um único grupo de chaves	5 Solicite uma cota maior.
Principais grupos de chaves associados a um único comportamento de cache	4 Solicite uma cota maior.
Grupos de chaves por Conta da AWS	10 Solicite uma cota maior.
Distribuições associadas a um único grupo de chaves	100 Solicite uma cota maior.

Cotas para conexões do WebSocket

Entidade	Cota padrão
Limite da resposta de origem (tempo limite de inatividade)	10 minutos Se o CloudFront não tiver detectado nenhum byte enviado da origem para o cliente nos últimos 10 minutos, a conexão será considerada

Entidade	Cota padrão
	como inativa e será fechada.

Cotas para criptografia no nível de campo

Entidade	Cota padrão
O tamanho máximo de um campo a ser criptografado	16 KB
Para obter mais informações, consulte Use criptografia de nível de campo para ajudar a proteger dados confidenciais .	
Número máximo de campos no corpo de uma solicitação quando a criptografia no nível do campo está configurada	10
O tamanho máximo de um corpo da solicitação quando a criptografia em nível de campo estiver configurada	1 MB
Número máximo de configurações de criptografia no nível do campo que pode ser associado a uma Conta da AWS	10
Número máximo de perfis de criptografia no nível do campo que pode ser associado a uma Conta da AWS	10
Número máximo de chaves públicas que pode ser adicionado a uma Conta da AWS	10
Número máximo de campos para criptografar que pode ser especificado em um perfil	10
Número máximo de distribuições do CloudFront que pode ser associado a uma configuração de criptografia em nível de campo	20
O número máximo de mapeamentos de perfil de argumento de consulta que pode ser incluído em uma configuração de criptografia no nível do campo	5

Cotas para cookies (configurações de cache herdadas)

Essas cotas se aplicam às configurações de cache herdadas do CloudFront. Recomendamos usar uma [política de cache](#) ou uma [política de solicitação de origem](#) em vez das configurações herdadas.

Entidade	Cota padrão
Cookies por comportamento de cache	10
Para obter mais informações, consulte Conteúdo em cache com base em cookies .	Solicite uma cota maior .
Número total de bytes dos nomes de cookies (não se aplicará se você configurar o CloudFront para encaminhar todos os cookies para a origem)	512 menos o número de cookies

Cotas em cadeias de consulta (configurações de cache herdadas)

Essas cotas se aplicam às configurações de cache herdadas do CloudFront. Recomendamos usar uma [política de cache](#) ou uma [política de solicitação de origem](#) em vez das configurações herdadas.

Entidade	Cota padrão
Número máximo de caracteres em uma cadeia de consulta	128 caracteres
Número máximo do total de caracteres para todas as cadeias de consulta no mesmo parâmetro	512 caracteres
Cadeias de consulta por comportamento de cache	10
Para obter mais informações, consulte Conteúdo em cache com base em parâmetros de string de consulta .	Solicite uma cota maior .

Cotas para cabeçalhos

Entidade	Cota padrão
<p>Cabeçalhos por comportamento de cache (configurações de cache herdadas)</p> <p>Para ter mais informações, consulte the section called “Conteúdo em cache com base nos cabeçalhos de solicitação”.</p>	<p>10</p> <p>Solicite uma cota maior.</p>
<p>Cabeçalhos personalizados: número máximo de cabeçalhos personalizados que você pode configurar o CloudFront para adicionar às solicitações de origem</p> <p>Para ter mais informações, consulte the section called “Adicionar cabeçalhos personalizados às solicitações de origem”.</p>	<p>10</p> <p>Solicite uma cota maior.</p>
<p>Cabeçalhos personalizados: número máximo de cabeçalhos personalizados que você pode adicionar a uma política de cabeçalhos de resposta</p>	<p>10</p> <p>Solicite uma cota maior.</p>
<p>Cabeçalhos personalizados: tamanho máximo do nome de um cabeçalho</p>	<p>256 caracteres</p>
<p>Cabeçalhos personalizados: tamanho máximo do valor de um cabeçalho</p>	<p>1,783 caracteres</p>
<p>Cabeçalhos personalizados: tamanho máximo de todos os valores e nomes de cabeçalho combinados</p>	<p>10,240 caracteres</p>
<p>Tamanho máximo do valor de cabeçalho de Content-Security-Policy</p>	<p>1,783 caracteres</p> <p>Solicite uma cota maior.</p>
<p>Tamanho máximo de um valor de cabeçalho de CORS (Access-Control-Allow-Origin)</p>	<p>1,783 caracteres</p> <p>Solicite uma cota maior.</p>

Informações relacionadas

Para ter mais informações, consulte [Amazon CloudFront endpoints and quotas](#) na Referência geral da AWS.

Exemplos de código para o CloudFront usando SDKs da AWS

Os exemplos de código a seguir mostram como usar o CloudFront com um kit de desenvolvimento de software (SDK) da AWS.

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar funções de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Os cenários são exemplos de código que mostram como realizar tarefas específicas chamando várias funções dentro de um serviço ou combinadas com outros Serviços da AWS.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Exemplos de código

- [Exemplos básicos para o CloudFront usando os SDKs da AWS](#)
 - [Ações para o CloudFront usando SDKs da AWS](#)
 - [Usar CreateDistribution com o AWS SDK ou a CLI](#)
 - [Usar CreateFunction com um SDK da AWS](#)
 - [Usar CreateInvalidation com uma CLI](#)
 - [Usar CreateKeyGroup com um SDK da AWS](#)
 - [Usar CreatePublicKey com o AWS SDK ou a CLI](#)
 - [Usar DeleteDistribution com o AWS SDK ou a CLI](#)
 - [Usar GetCloudFrontOriginAccessIdentity com uma CLI](#)
 - [Usar GetCloudFrontOriginAccessIdentityConfig com uma CLI](#)
 - [Usar GetDistribution com uma CLI](#)
 - [Usar o GetDistributionConfig com um AWS SDK ou a CLI](#)
 - [Usar ListCloudFrontOriginAccessIdentities com uma CLI](#)
 - [Usar o ListDistributions com um AWS SDK ou a CLI](#)
 - [Usar UpdateDistribution com o AWS SDK ou a CLI](#)
 - [Cenários para o CloudFront usando SDKs da AWS](#)

- [Exclua recursos de assinatura do CloudFront usando o AWS SDK](#)
- [Criar URLs e cookies assinados usando um AWS SDK](#)
- [Exemplos do CloudFront Functions para o CloudFront usando SDKs da AWS](#)
 - [Adicionar cabeçalhos de segurança HTTP a um evento de resposta do visualizador do CloudFront Functions](#)
 - [Adicionar um cabeçalho de CORS a um evento de resposta do visualizador do CloudFront Functions](#)
 - [Adicionar um cabeçalho de controle de cache a um evento de resposta do visualizador do CloudFront Functions](#)
 - [Adicione um cabeçalho IP de cliente verdadeiro a um evento de solicitação do visualizador do CloudFront Functions](#)
 - [Adicionar um cabeçalho de origem a um evento de resposta do visualizador do CloudFront Functions](#)
 - [Adicionar index.html aos URLs de solicitação sem um nome de arquivo em um evento de solicitação do visualizador do CloudFront Functions](#)
 - [Normalizar os parâmetros de string de consulta em uma solicitação do visualizador do CloudFront Functions](#)
 - [Redirecionar para um novo URL em um evento de solicitação do visualizador do CloudFront Functions](#)
 - [Reescrever um URI de solicitação com base na configuração do KeyValueStore para um evento de solicitação do visualizador do CloudFront Functions](#)
 - [Usar pares de chave-valor em uma solicitação do visualizador do CloudFront Functions](#)
 - [Validar um token simples em uma solicitação do visualizador do CloudFront Functions](#)

Exemplos básicos para o CloudFront usando os SDKs da AWS

Os exemplos de código a seguir mostram como usar os conceitos básicos do Amazon CloudFront com SDKs da AWS.

Exemplos

- [Ações para o CloudFront usando SDKs da AWS](#)
 - [Usar CreateDistribution com o AWS SDK ou a CLI](#)
 - [Usar CreateFunction com um SDK da AWS](#)

- [Usar CreateInvalidation com uma CLI](#)
- [Usar CreateKeyGroup com um SDK da AWS](#)
- [Usar CreatePublicKey com o AWS SDK ou a CLI](#)
- [Usar DeleteDistribution com o AWS SDK ou a CLI](#)
- [Usar GetCloudFrontOriginAccessIdentity com uma CLI](#)
- [Usar GetCloudFrontOriginAccessIdentityConfig com uma CLI](#)
- [Usar GetDistribution com uma CLI](#)
- [Usar o GetDistributionConfig com um AWS SDK ou a CLI](#)
- [Usar ListCloudFrontOriginAccessIdentities com uma CLI](#)
- [Usar o ListDistributions com um AWS SDK ou a CLI](#)
- [Usar UpdateDistribution com o AWS SDK ou a CLI](#)

Ações para o CloudFront usando SDKs da AWS

Os exemplos de código a seguir demonstram como realizar ações individuais do CloudFront com SDKs da AWS. Cada exemplo inclui um link para o GitHub, em que é possível encontrar instruções para configurar e executar o código.

Esses trechos chamam a API do CloudFront e são trechos de código de programas maiores que devem ser executados no contexto. Você pode ver as ações em contexto em [Cenários para o CloudFront usando SDKs da AWS](#).

Os exemplos a seguir incluem apenas as ações mais utilizadas. Para obter uma lista completa, consulte a [Referência de APIs do Amazon CloudFront](#).

Exemplos

- [Usar CreateDistribution com o AWS SDK ou a CLI](#)
- [Usar CreateFunction com um SDK da AWS](#)
- [Usar CreateInvalidation com uma CLI](#)
- [Usar CreateKeyGroup com um SDK da AWS](#)
- [Usar CreatePublicKey com o AWS SDK ou a CLI](#)
- [Usar DeleteDistribution com o AWS SDK ou a CLI](#)
- [Usar GetCloudFrontOriginAccessIdentity com uma CLI](#)
- [Usar GetCloudFrontOriginAccessIdentityConfig com uma CLI](#)

- [Usar GetDistribution com uma CLI](#)
- [Usar o GetDistributionConfig com um AWS SDK ou a CLI](#)
- [Usar ListCloudFrontOriginAccessIdentities com uma CLI](#)
- [Usar o ListDistributions com um AWS SDK ou a CLI](#)
- [Usar UpdateDistribution com o AWS SDK ou a CLI](#)

Usar **CreateDistribution** com o AWS SDK ou a CLI

Os exemplos de código a seguir mostram como usar o `CreateDistribution`.

CLI

AWS CLI

Como criar uma distribuição do CloudFront

O exemplo a seguir cria uma distribuição para um bucket do S3 chamado `awsexamplebucket` e também especifica `index.html` como o objeto raiz padrão, usando argumentos de linha de comando:

```
aws cloudfront create-distribution \  
  --origin-domain-name awsexamplebucket.s3.amazonaws.com \  
  --default-root-object index.html
```

Em vez de usar argumentos de linha de comando, forneça a configuração de distribuição em um arquivo JSON, conforme mostrado no seguinte exemplo:

```
aws cloudfront create-distribution \  
  --distribution-config file://dist-config.json
```

O arquivo `dist-config.json` é um documento JSON na pasta atual que contém o seguinte:

```
{  
  "CallerReference": "cli-example",  
  "Aliases": {  
    "Quantity": 0  
  },  
  "DefaultRootObject": "index.html",  
  "Origins": {
```

```
"Quantity": 1,
"Items": [
  {
    "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
    "DomainName": "awsexamplebucket.s3.amazonaws.com",
    "OriginPath": "",
    "CustomHeaders": {
      "Quantity": 0
    },
    "S3OriginConfig": {
      "OriginAccessIdentity": ""
    }
  }
],
"OriginGroups": {
  "Quantity": 0
},
"DefaultCacheBehavior": {
  "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
  "ForwardedValues": {
    "QueryString": false,
    "Cookies": {
      "Forward": "none"
    },
    "Headers": {
      "Quantity": 0
    },
    "QueryStringCacheKeys": {
      "Quantity": 0
    }
  },
  "TrustedSigners": {
    "Enabled": false,
    "Quantity": 0
  },
  "ViewerProtocolPolicy": "allow-all",
  "MinTTL": 0,
  "AllowedMethods": {
    "Quantity": 2,
    "Items": [
      "HEAD",
      "GET"
    ]
  },

```

```
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
},
"Restrictions": {
    "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
    }
},
"WebACLId": "",
```

```
"HttpVersion": "http2",
"IsIPv6Enabled": true
}
```

Se você fornecer as informações de distribuição com um argumento de linha de comando ou um arquivo JSON, a saída será a mesma:

```
{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/
EMLARXS9EXAMPLE",
  "ETag": "E9LHASXEXAMPLE",
  "Distribution": {
    "Id": "EMLARXS9EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EMLARXS9EXAMPLE",
    "Status": "InProgress",
    "LastModifiedTime": "2019-11-22T00:55:15.705Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d111111abcdef8.cloudfront.net",
    "ActiveTrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
    "DistributionConfig": {
      "CallerReference": "cli-example",
      "Aliases": {
        "Quantity": 0
      },
      "DefaultRootObject": "index.html",
      "Origins": {
        "Quantity": 1,
        "Items": [
          {
            "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
            "DomainName": "awsexamplebucket.s3.amazonaws.com",
            "OriginPath": "",
            "CustomHeaders": {
              "Quantity": 0
            },
            "S3OriginConfig": {
              "OriginAccessIdentity": ""
            }
          }
        ]
      }
    }
  }
}
```



```
    },
    "OriginGroups": {
      "Quantity": 0
    },
  },
  "DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-
example",
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      },
    },
    "Headers": {
      "Quantity": 0
    },
    "QueryStringCacheKeys": {
      "Quantity": 0
    }
  },
  "TrustedSigners": {
    "Enabled": false,
    "Quantity": 0
  },
  "ViewerProtocolPolicy": "allow-all",
  "MinTTL": 0,
  "AllowedMethods": {
    "Quantity": 2,
    "Items": [
      "HEAD",
      "GET"
    ],
  },
  "CachedMethods": {
    "Quantity": 2,
    "Items": [
      "HEAD",
      "GET"
    ]
  },
  "SmoothStreaming": false,
  "DefaultTTL": 86400,
  "MaxTTL": 31536000,
  "Compress": false,
  "LambdaFunctionAssociations": {
```

```

        "Quantity": 0
      },
      "FieldLevelEncryptionId": ""
    },
    "CacheBehaviors": {
      "Quantity": 0
    },
    "CustomErrorResponses": {
      "Quantity": 0
    },
    "Comment": "",
    "Logging": {
      "Enabled": false,
      "IncludeCookies": false,
      "Bucket": "",
      "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
      "CloudFrontDefaultCertificate": true,
      "MinimumProtocolVersion": "TLSv1",
      "CertificateSource": "cloudfront"
    },
    "Restrictions": {
      "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
  }
}

```

- Para obter detalhes da API, consulte [CreateDistribution](#) na Referência de comandos da AWS CLI.

Java

SDK para Java 2.x

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

O exemplo a seguir usa um bucket do Amazon Simple Storage Service (Amazon S3) como origem de conteúdo.

Depois de criar a distribuição, o código cria um [CloudFrontWaiter](#) para esperar até que a distribuição seja implantada antes de retornar a distribuição.

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import software.amazon.awssdk.core.internal.waiters.ResponseOrException;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import
    software.amazon.awssdk.services.cloudfront.model.CreateDistributionResponse;
import software.amazon.awssdk.services.cloudfront.model.Distribution;
import software.amazon.awssdk.services.cloudfront.model.GetDistributionResponse;
import software.amazon.awssdk.services.cloudfront.model.ItemSelection;
import software.amazon.awssdk.services.cloudfront.model.Method;
import software.amazon.awssdk.services.cloudfront.model.ViewerProtocolPolicy;
import software.amazon.awssdk.services.cloudfront.waiters.CloudFrontWaiter;
import software.amazon.awssdk.services.s3.S3Client;

import java.time.Instant;

public class CreateDistribution {

    private static final Logger logger =
        LoggerFactory.getLogger(CreateDistribution.class);

    public static Distribution createDistribution(CloudFrontClient
        cloudFrontClient, S3Client s3Client,
        final String bucketName, final String keyGroupId, final
        String originAccessControlId) {
```

```
        final String region = s3Client.headBucket(b ->
b.bucket(bucketName)).sdkHttpResponse().headers()
            .get("x-amz-bucket-region").get(0);
        final String originDomain = bucketName + ".s3." + region +
".amazonaws.com";
        String originId = originDomain; // Use the originDomain value for
the originId.

        // The service API requires some deprecated methods, such as
// DefaultCacheBehavior.Builder#minTTL and #forwardedValue.
        CreateDistributionResponse createDistResponse =
cloudFrontClient.createDistribution(builder -> builder
            .distributionConfig(b1 -> b1
                .origins(b2 -> b2
                    .quantity(1)
                    .items(b3 -> b3

                .domainName(originDomain)

                .id(originId)

                .s3OriginConfig(builder4 -> builder4
                    .originAccessIdentity(
                        ""))

                .originAccessControlId(
                    originAccessControlId)))
            .defaultCacheBehavior(b2 -> b2

                .viewerProtocolPolicy(ViewerProtocolPolicy.ALLOW_ALL)

                .targetOriginId(originId)
                    .minTTL(200L)

                .forwardedValues(b5 -> b5

                .cookies(cp -> cp

                .forward(ItemSelection.NONE))
```

```
.queryString(true))

.trustedKeyGroups(b3 -> b3

.quantity(1)

.items(keyGroupId)

.enabled(true))

.allowedMethods(b4 -> b4

.quantity(2)

.items(Method.HEAD, Method.GET)

.cachedMethods(b5 -> b5

    .quantity(2)

    .items(Method.HEAD,

                Method.GET))))

.cacheBehaviors(b -> b

    .quantity(1)

    .items(b2 -> b2

        .pathPattern("/index.html")

        .viewerProtocolPolicy(

            ViewerProtocolPolicy.ALLOW_ALL)

        .targetOriginId(originId)

        .trustedKeyGroups(b3 -> b3

            .quantity(1)

            .items(keyGroupId)

            .enabled(true))
```

```

.minTTL(200L)

.forwardedValues(b4 -> b4

    .cookies(cp -> cp

        .forward(ItemSelection.NONE))

    .queryString(true))

.allowedMethods(b5 -> b5.quantity(2)

    .items(Method.HEAD,

        Method.GET)

    .cachedMethods(b6 -> b6

        .quantity(2)

        .items(Method.HEAD,

            Method.GET))))))
    .enabled(true)
    .comment("Distribution built with
java")

.callerReference(Instant.now().toString()));

    final Distribution distribution =
createDistResponse.distribution();
    logger.info("Distribution created. DomainName: [{}] Id: [{}]",
distribution.domainName(),
        distribution.id());
    logger.info("Waiting for distribution to be deployed ...");
    try (CloudFrontWaiter cfWaiter =
CloudFrontWaiter.builder().client(cloudFrontClient).build()) {
        ResponseOrException<GetDistributionResponse>
responseOrException = cfWaiter
            .waitUntilDistributionDeployed(builder ->
builder.id(distribution.id()))
            .matched();
        responseOrException.response()

```

```

        .orElseThrow(() -> new
RuntimeException("Distribution not created"));
        logger.info("Distribution deployed. DomainName: [{}] Id:
[{}]", distribution.domainName(),
distribution.id());
    }
    return distribution;
}
}
}

```

- Para obter detalhes da API, consulte [CreateDistribution](#) na Referência da API AWS SDK for Java 2.x.

PowerShell

Ferramentas para PowerShell

Exemplo 1: cria uma distribuição básica do CloudFront, configurada com registro em log e armazenamento em cache.

```

$origin = New-Object Amazon.CloudFront.Model.Origin
$origin.DomainName = "amzn-s3-demo-bucket.s3.amazonaws.com"
$origin.Id = "UniqueOrigin1"
$origin.S3OriginConfig = New-Object Amazon.CloudFront.Model.S3OriginConfig
$origin.S3OriginConfig.OriginAccessIdentity = ""
New-CFDistribution `
    -DistributionConfig_Enabled $true `
    -DistributionConfig_Comment "Test distribution" `
    -Origins_Item $origin `
    -Origins_Quantity 1 `
    -Logging_Enabled $true `
    -Logging_IncludeCookie $true `
    -Logging_Bucket amzn-s3-demo-logging-bucket.s3.amazonaws.com `
    -Logging_Prefix "help/" `
    -DistributionConfig_CallerReference Client1 `
    -DistributionConfig_DefaultRootObject index.html `
    -DefaultCacheBehavior_TargetOriginId $origin.Id `
    -ForwardedValues_QueryString $true `
    -Cookies_Forward all `
    -WhitelistedNames_Quantity 0 `
    -TrustedSigners_Enabled $false `

```

```
-TrustedSigners_Quantity 0 `
-DefaultCacheBehavior_ViewerProtocolPolicy allow-all `
-DefaultCacheBehavior_MinTTL 1000 `
-DistributionConfig_PriceClass "PriceClass_All" `
-CacheBehaviors_Quantity 0 `
-Aliases_Quantity 0
```

- Para detalhes da API, consulte [CreateDistribution](#) na AWS Tools for PowerShell Cmdlet Reference.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar **CreateFunction** com um SDK da AWS

O código de exemplo a seguir mostra como usar `CreateFunction`.

Java

SDK para Java 2.x

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

```
import software.amazon.awssdk.core.SdkBytes;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import software.amazon.awssdk.services.cloudfront.model.CloudFrontException;
import software.amazon.awssdk.services.cloudfront.model.CreateFunctionRequest;
import software.amazon.awssdk.services.cloudfront.model.CreateFunctionResponse;
import software.amazon.awssdk.services.cloudfront.model.FunctionConfig;
import software.amazon.awssdk.services.cloudfront.model.FunctionRuntime;
import java.io.InputStream;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
```



```
*
* For more information, see the following documentation topic:
*
* https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-started.html
*/
public class CreateFunction {

    public static void main(String[] args) {
        final String usage = ""

            Usage:
                <functionName> <filePath>

            Where:
                functionName - The name of the function to create.\s
                filePath - The path to a file that contains the application
            logic for the function.\s
            """;

        if (args.length != 2) {
            System.out.println(usage);
            System.exit(1);
        }

        String functionName = args[0];
        String filePath = args[1];
        CloudFrontClient cloudFrontClient = CloudFrontClient.builder()
            .region(Region.AWS_GLOBAL)
            .build();

        String funArn = createNewFunction(cloudFrontClient, functionName,
filePath);
        System.out.println("The function ARN is " + funArn);
        cloudFrontClient.close();
    }

    public static String createNewFunction(CloudFrontClient cloudFrontClient,
String functionName, String filePath) {
        try {
            InputStream fileIs =
CreateFunction.class.getClassLoader().getResourceAsStream(filePath);
            SdkBytes functionCode = SdkBytes.fromInputStream(fileIs);
```

```
        FunctionConfig config = FunctionConfig.builder()
            .comment("Created by using the CloudFront Java API")
            .runtime(FunctionRuntime.CLOUDFRONT_JS_1_0)
            .build();

        CreateFunctionRequest functionRequest =
        CreateFunctionRequest.builder()
            .name(functionName)
            .functionCode(functionCode)
            .functionConfig(config)
            .build();

        CreateFunctionResponse response =
        cloudFrontClient.createFunction(functionRequest);
        return response.functionSummary().functionMetadata().functionARN();

    } catch (CloudFrontException e) {
        System.err.println(e.getMessage());
        System.exit(1);
    }
    return "";
}
}
```

- Para obter detalhes da API, consulte [CreateFunction](#) na Referência da API AWS SDK for Java 2.x.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar **CreateInvalidation** com uma CLI

Os exemplos de código a seguir mostram como usar o `CreateInvalidation`.

CLI

AWS CLI

Como criar uma invalidação para uma distribuição do CloudFront

O exemplo `create-invalidation` a seguir cria uma invalidação para os arquivos especificados na distribuição especificada do CloudFront:

```
aws cloudfront create-invalidation \  
  --distribution-id EDFDVBD6EXAMPLE \  
  --paths "/example-path/example-file.jpg" "/example-path/example-file2.png"
```

Saída:

```
{  
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/  
EDFDVBD6EXAMPLE/invalidation/I1JLWSDAP8FU89",  
  "Invalidation": {  
    "Id": "I1JLWSDAP8FU89",  
    "Status": "InProgress",  
    "CreateTime": "2019-12-05T18:24:51.407Z",  
    "InvalidationBatch": {  
      "Paths": {  
        "Quantity": 2,  
        "Items": [  
          "/example-path/example-file2.png",  
          "/example-path/example-file.jpg"  
        ]  
      },  
      "CallerReference": "cli-1575570291-670203"  
    }  
  }  
}
```

No exemplo anterior, a AWS CLI gerou automaticamente um `CallerReference` aleatório. Para especificar seu próprio `CallerReference` ou evitar passar os parâmetros de invalidação como argumentos da linha de comando, é possível usar um arquivo JSON. O exemplo a seguir cria uma invalidação para dois arquivos fornecendo os parâmetros de invalidação em um arquivo JSON chamado `inv-batch.json`:

```
aws cloudfront create-invalidation \  
  --distribution-id EDFDVBD6EXAMPLE \  
  --invalidation-batch file://inv-batch.json
```

Conteúdo de `inv-batch.json`:

```
{
  "Paths": {
    "Quantity": 2,
    "Items": [
      "/example-path/example-file.jpg",
      "/example-path/example-file2.png"
    ]
  },
  "CallerReference": "cli-example"
}
```

Saída:

```
{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/EDFDVBD6EXAMPLE/invalidation/I2J0I21PCUY0IK",
  "Invalidation": {
    "Id": "I2J0I21PCUY0IK",
    "Status": "InProgress",
    "CreateTime": "2019-12-05T18:40:49.413Z",
    "InvalidationBatch": {
      "Paths": {
        "Quantity": 2,
        "Items": [
          "/example-path/example-file.jpg",
          "/example-path/example-file2.png"
        ]
      },
      "CallerReference": "cli-example"
    }
  }
}
```

- Para detalhes da API, consulte [CreateInvalidation](#) na Referência de comandos da AWS CLI.

PowerShell

Ferramentas para PowerShell

Exemplo 1: este exemplo cria uma nova invalidação em uma distribuição com um ID de EXAMPLNSTXAXE. O CallerReference é um ID exclusivo escolhido pelo usuário; nesse caso, é usado um carimbo de data/hora representando 15 de maio de 2019 às 9h. A variável

\$Paths armazena três caminhos para arquivos de imagem e mídia que o usuário não deseja como parte do cache distribuído. O valor do parâmetro `-Paths_Quantity` é o número total de caminhos especificados no parâmetro `-Paths_Item`.

```
$Paths = "/images/*.gif", "/images/image1.jpg", "/videos/*.mp4"
New-CFInvalidation -DistributionId "EXAMPLENSTXAXE" -
InvalidationBatch_CallerReference 201905150900000 -Paths_Item $Paths -
Paths_Quantity 3
```

Saída:

```
Invalidation          Location
-----
Amazon.CloudFront.Model.Invalidatio https://cloudfront.amazonaws.com/2018-11-05/
distribution/EXAMPLENSTXAXE/invalidation/EXAMPLE8NOK9H
```

- Para detalhes da API, consulte [CreateInvalidation](#) na AWS Tools for PowerShell Cmdlet Reference.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar `CreateKeyGroup` com um SDK da AWS

O código de exemplo a seguir mostra como usar `CreateKeyGroup`.

Java

SDK para Java 2.x

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

Um grupo de chaves exige pelo menos uma chave pública usada para verificar URLs ou cookies assinados.

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;

import java.util.UUID;

public class CreateKeyGroup {
    private static final Logger logger =
        LoggerFactory.getLogger(CreateKeyGroup.class);

    public static String createKeyGroup(CloudFrontClient cloudFrontClient, String
publicKeyId) {
        String keyGroupId = cloudFrontClient.createKeyGroup(b ->
b.keyGroupConfig(c -> c
            .items(publicKeyId)
            .name("JavaKeyGroup" + UUID.randomUUID()))
            .keyGroup().id());
        logger.info("KeyGroup created with ID: [{}]", keyGroupId);
        return keyGroupId;
    }
}
```

- Para obter detalhes da API, consulte [CreateKeyGroup](#) na Referência da API AWS SDK for Java 2.x.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar **CreatePublicKey** com o AWS SDK ou a CLI

Os exemplos de código a seguir mostram como usar o `CreatePublicKey`.

CLI

AWS CLI

Como criar uma chave pública do CloudFront

O exemplo a seguir cria uma chave pública do CloudFront fornecendo os parâmetros em um arquivo JSON chamado `pub-key-config.json`. Antes de usar esse comando, você deve ter uma chave pública codificada em PEM. Para obter mais informações, consulte [Criar um par de chaves do RSA](#) no Guia do desenvolvedor do Amazon CloudFront.

```
aws cloudfront create-public-key \
  --public-key-config file://pub-key-config.json
```

O arquivo `pub-key-config.json` é um documento JSON na pasta atual que contém o seguinte. Observe que a chave pública é codificada no formato PEM.

```
{
  "CallerReference": "cli-example",
  "Name": "ExampleKey",
  "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPMbCA2Ks01nd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPNhinFJ8Wdy8YBXSPpy7r95y1UQd9LfYTBzVZYG2tSesplc0kjM3\n2Uu
+oMwxQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nnq
+kGZ2NQ0FyIyT2eiLK0X5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nnrwIDAQAB\n-----
END PUBLIC KEY-----\n",
  "Comment": "example public key"
}
```

Saída:

```
{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/public-key/
KDFB19YGCR002",
  "ETag": "E2QWRUHEXAMPLE",
  "PublicKey": {
    "Id": "KDFB19YGCR002",
    "CreatedTime": "2019-12-05T18:51:43.781Z",
    "PublicKeyConfig": {
      "CallerReference": "cli-example",
      "Name": "ExampleKey",
      "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPMbCA2Ks01nd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1U0WcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAz03ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPNhinFJ8Wdy8YBXSPpy7r95y1UQd9LfYTBzVZYG2tSesplc0kjM3\n2Uu
```

```
+oMwxQAw1NINnSLPinMVsutJy6Zq1V3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nq
+kGZ2NQ0FyIyT2eiLK0X5RgB/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nrwIDAQAB\n-----
END PUBLIC KEY-----\n",
    "Comment": "example public key"
  }
}
}
```

- Para obter detalhes da API, consulte [CreatePublicKey](#) na Referência de comandos da AWS CLI.

Java

SDK para Java 2.x

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

O exemplo de código a seguir lê uma chave pública e a carrega no Amazon CloudFront.

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import software.amazon.awssdk.services.cloudfront.model.CreatePublicKeyResponse;
import software.amazon.awssdk.utils.IoUtils;

import java.io.IOException;
import java.io.InputStream;
import java.util.UUID;

public class CreatePublicKey {
    private static final Logger logger =
        LoggerFactory.getLogger(CreatePublicKey.class);

    public static String createPublicKey(CloudFrontClient cloudFrontClient,
        String publicKeyFileName) {
        try (InputStream is =
            CreatePublicKey.class.getClassLoader().getResourceAsStream(publicKeyFileName)) {
            String publicKeyString = IoUtils.toUtf8String(is);
```



```
        CreatePublicKeyResponse createPublicKeyResponse = cloudFrontClient
            .createPublicKey(b -> b.publicKeyConfig(c -> c
                .name("JavaCreatedPublicKey" + UUID.randomUUID())
                .encodedKey(publicKeyString)
                .callerReference(UUID.randomUUID().toString())));
        String createdPublicKeyId = createPublicKeyResponse.publicKey().id();
        logger.info("Public key created with id: [{}]", createdPublicKeyId);
        return createdPublicKeyId;

    } catch (IOException e) {
        throw new RuntimeException(e);
    }
}
```

- Para obter detalhes da API, consulte [CreatePublicKey](#) na Referência da API AWS SDK for Java 2.x.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar **DeleteDistribution** com o AWS SDK ou a CLI

Os exemplos de código a seguir mostram como usar o DeleteDistribution.

CLI

AWS CLI

Como excluir uma distribuição do CloudFront

O exemplo a seguir exclui a distribuição do CloudFront com o ID EDFDVBD6EXAMPLE. Você deve desabilitar uma distribuição antes de excluí-la. Use o comando update-distribution para desabilitar uma distribuição. Para obter mais informações, consulte exemplos do comando update-distribution.

Uma distribuição pode ser excluída depois de ser desabilitada. Para excluir uma distribuição, use a opção `--if-match` para fornecer o ETag da distribuição. Use o comando get-distribution ou get-distribution-config para obter o ETag.

```
aws cloudfront delete-distribution \  
  --id EDFDVBD6EXAMPLE \  
  --if-match E2QWRUHEXAMPLE
```

Quando bem-sucedido, este comando não tem saída.

- Para obter detalhes da API, consulte [DeleteDistribution](#) na Referência de comandos da AWS CLI.

Java

SDK para Java 2.x

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

O exemplo de código a seguir atualiza uma distribuição para desativada, usa um waiter que aguarda a implantação da alteração e, em seguida, exclui a distribuição.

```
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import software.amazon.awssdk.core.internal.waiters.ResponseOrException;  
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;  
import  
  software.amazon.awssdk.services.cloudfront.model.DeleteDistributionResponse;  
import software.amazon.awssdk.services.cloudfront.model.DistributionConfig;  
import software.amazon.awssdk.services.cloudfront.model.GetDistributionResponse;  
import software.amazon.awssdk.services.cloudfront.waiters.CloudFrontWaiter;  
  
public class DeleteDistribution {  
    private static final Logger logger =  
        LoggerFactory.getLogger(DeleteDistribution.class);  
  
    public static void deleteDistribution(final CloudFrontClient  
cloudFrontClient, final String distributionId) {  
        // First, disable the distribution by updating it.  
        GetDistributionResponse response =  
cloudFrontClient.getDistribution(b -> b
```

```

        .id(distributionId));
    String etag = response.eTag();
    DistributionConfig distConfig =
response.distribution().distributionConfig();

    cloudFrontClient.updateDistribution(builder -> builder
        .id(distributionId)
        .distributionConfig(builder1 -> builder1

.cacheBehaviors(distConfig.cacheBehaviors())

.defaultCacheBehavior(distConfig.defaultCacheBehavior())
        .enabled(false)
        .origins(distConfig.origins())
        .comment(distConfig.comment())

.callerReference(distConfig.callerReference())

.defaultCacheBehavior(distConfig.defaultCacheBehavior())

.priceClass(distConfig.priceClass())
        .aliases(distConfig.aliases())
        .logging(distConfig.logging())

.defaultRootObject(distConfig.defaultRootObject())

.customErrorResponses(distConfig.customErrorResponses())

.httpVersion(distConfig.httpVersion())

.isIPV6Enabled(distConfig.isIPV6Enabled())

.restrictions(distConfig.restrictions())

.viewerCertificate(distConfig.viewerCertificate())
        .webACLId(distConfig.webACLId())

.originGroups(distConfig.originGroups()))
        .ifMatch(etag));

    logger.info("Distribution [{}] is DISABLED, waiting for
deployment before deleting ...",
        distributionId);
    GetDistributionResponse distributionResponse;

```

```

        try (CloudFrontWaiter cfWaiter =
CloudFrontWaiter.builder().client(cloudFrontClient).build()) {
            ResponseOrException<GetDistributionResponse>
responseOrException = cfWaiter
                .waitUntilDistributionDeployed(builder ->
builder.id(distributionId)).matched();
            distributionResponse = responseOrException.response()
                .orElseThrow(() -> new
RuntimeException("Could not disable distribution"));
        }

        DeleteDistributionResponse deleteDistributionResponse =
cloudFrontClient
                .deleteDistribution(builder -> builder
                .id(distributionId)

                .ifMatch(distributionResponse.eTag()));
        if (deleteDistributionResponse.sdkHttpResponse().isSuccessful())
{
            logger.info("Distribution [{}] DELETED", distributionId);
        }
    }
}

```

- Para obter detalhes da API, consulte [DeleteDistribution](#) na Referência da API do AWS SDK for Java 2.x.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar `GetCloudFrontOriginAccessIdentity` com uma CLI

Os exemplos de código a seguir mostram como usar o `GetCloudFrontOriginAccessIdentity`.

CLI

AWS CLI

Para obter uma identidade do acesso de origem do CloudFront

O exemplo a seguir obtém a identidade do acesso de origem (OAI) do CloudFront com a ID E74FTE3AEXAMPLE, incluindo sua ETag e a ID canônica associada do S3. O ID da OAI é retornado na saída dos comandos `create-cloud-front-origin-access-identity` e `list-cloud-front-origin-access-identities`.

```
aws cloudfront get-cloud-front-origin-access-identity --id E74FTE3AEXAMPLE
```

Saída:

```
{
  "ETag": "E2QWRUHEXAMPLE",
  "CloudFrontOriginAccessIdentity": {
    "Id": "E74FTE3AEXAMPLE",
    "S3CanonicalUserId":
"cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
    "CloudFrontOriginAccessIdentityConfig": {
      "CallerReference": "cli-example",
      "Comment": "Example OAI"
    }
  }
}
```

- Para obter detalhes da API, consulte [GetCloudFrontOriginAccessIdentity](#) na Referência de comandos da AWS CLI.

PowerShell

Ferramentas para PowerShell

Exemplo 1: este exemplo retorna uma identidade do acesso de origem específica do Amazon CloudFront, especificada pelo parâmetro `-Id`. Embora o parâmetro `-Id` não seja obrigatório, se você não o especificar, nenhum resultado será retornado.

```
Get-CFCloudFrontOriginAccessIdentity -Id E3XXXXXXXXXXRT
```

Saída:

```
CloudFrontOriginAccessIdentityConfig    Id
-----
S3CanonicalUserId
```

```
-----  
-----  
Amazon.CloudFront.Model.CloudFrontOriginAccessIdentityConfigResponse  
4b6e...
```

- Para obter detalhes da API, consulte [GetCloudFrontOriginAccessIdentity](#) na AWS Tools for PowerShell Cmdlet Reference.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar `GetCloudFrontOriginAccessIdentityConfig` com uma CLI

Os exemplos de código a seguir mostram como usar o `GetCloudFrontOriginAccessIdentityConfig`.

CLI

AWS CLI

Como obter uma configuração de identidade do acesso de origem do CloudFront

O exemplo a seguir obtém metadados sobre a identidade do acesso de origem (OAI) do CloudFront com o ID `E74FTE3AEXAMPLE`, incluindo a ETag. O ID da OAI é retornado na saída dos comandos `create-cloud-front-origin-access-identity` e `list-cloud-front-origin-access-identities`.

```
aws cloudfront get-cloud-front-origin-access-identity-config --id E74FTE3AEXAMPLE
```

Saída:

```
{  
  "ETag": "E2QWRUHEXAMPLE",  
  "CloudFrontOriginAccessIdentityConfig": {  
    "CallerReference": "cli-example",  
    "Comment": "Example OAI"  
  }  
}
```

- Para obter detalhes da API, consulte [GetCloudFrontOriginAccessIdentityConfig](#) na Referência de comandos da AWS CLI.

PowerShell

Ferramentas para PowerShell

Exemplo 1: este exemplo retorna informações de configuração sobre uma única identidade do acesso de origem do Amazon CloudFront, especificada pelo parâmetro `-Id`. Ocorrem erros se nenhum parâmetro `-Id` for especificado.

```
Get-CFCloudFrontOriginAccessIdentityConfig -Id E3XXXXXXXXXXRT
```

Saída:

```

CallerReference                               Comment
-----
mycallerreference: 2/1/2011 1:16:32 PM         Caller
reference: 2/1/2011 1:16:32 PM

```

- Para obter detalhes da API, consulte [GetCloudFrontOriginAccessIdentityConfig](#) na AWS Tools for PowerShell Cmdlet Reference.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar **GetDistribution** com uma CLI

Os exemplos de código a seguir mostram como usar o `GetDistribution`.

CLI

AWS CLI

Como obter uma distribuição do CloudFront

O exemplo a seguir obtém a distribuição do CloudFront com o ID `EDFDVBD6EXAMPLE`, incluindo a `ETag`. O ID de distribuição é retornado nos comandos `create-distribution` e `list-distributions`.

```
aws cloudfront get-distribution --id EDFDVBD6EXAMPLE
```

Saída:

```
{
  "ETag": "E2QWRUHEXAMPLE",
  "Distribution": {
    "Id": "EDFDVBD6EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
    "Status": "Deployed",
    "LastModifiedTime": "2019-12-04T23:35:41.433Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d111111abcdef8.cloudfront.net",
    "ActiveTrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
    "DistributionConfig": {
      "CallerReference": "cli-example",
      "Aliases": {
        "Quantity": 0
      },
      "DefaultRootObject": "index.html",
      "Origins": {
        "Quantity": 1,
        "Items": [
          {
            "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
            "DomainName": "awsexamplebucket.s3.amazonaws.com",
            "OriginPath": "",
            "CustomHeaders": {
              "Quantity": 0
            },
            "S3OriginConfig": {
              "OriginAccessIdentity": ""
            }
          }
        ]
      },
      "OriginGroups": {
        "Quantity": 0
      },
      "DefaultCacheBehavior": {
```



```
example",
  "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-
example",
  "ForwardedValues": {
    "QueryString": false,
    "Cookies": {
      "Forward": "none"
    },
    "Headers": {
      "Quantity": 0
    },
    "QueryStringCacheKeys": {
      "Quantity": 0
    }
  },
  "TrustedSigners": {
    "Enabled": false,
    "Quantity": 0
  },
  "ViewerProtocolPolicy": "allow-all",
  "MinTTL": 0,
  "AllowedMethods": {
    "Quantity": 2,
    "Items": [
      "HEAD",
      "GET"
    ],
    "CachedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ]
    }
  },
  "SmoothStreaming": false,
  "DefaultTTL": 86400,
  "MaxTTL": 31536000,
  "Compress": false,
  "LambdaFunctionAssociations": {
    "Quantity": 0
  },
  "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
```

```
        "Quantity": 0
    },
    "CustomErrorResponses": {
        "Quantity": 0
    },
    "Comment": "",
    "Logging": {
        "Enabled": false,
        "IncludeCookies": false,
        "Bucket": "",
        "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
    }
}
```

- Para obter detalhes da API, consulte [GetDistribution](#) na Referência de comandos da AWS CLI.

PowerShell

Ferramentas para PowerShell

Exemplo 1: recupera as informações de uma distribuição específica.

```
Get-CFDistribution -Id EXAMPLE0000ID
```

- Para obter detalhes da API, consulte [GetDistribution](#) na AWS Tools for PowerShell Cmdlet Reference.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar o **GetDistributionConfig** com um AWS SDK ou a CLI

Os exemplos de código a seguir mostram como usar o `GetDistributionConfig`.

CLI

AWS CLI

Como obter uma configuração de distribuição do CloudFront

O exemplo a seguir obtém metadados sobre a distribuição do CloudFront com o ID EDFDVBD6EXAMPLE, incluindo o ETag. O ID de distribuição é retornado nos comandos `create-distribution` e `list-distributions`.

```
aws cloudfront get-distribution-config --id EDFDVBD6EXAMPLE
```

Saída:

```
{
  "ETag": "E2QWRUHEXAMPLE",
  "DistributionConfig": {
    "CallerReference": "cli-example",
    "Aliases": {
      "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
      "Quantity": 1,
      "Items": [
        {
          "Id": "awsexamplebucket.s3.amazonaws.com-cli-example",
          "DomainName": "awsexamplebucket.s3.amazonaws.com",
          "OriginPath": "",
          "CustomHeaders": {
            "Quantity": 0
          }
        }
      ]
    }
  }
}
```

```
        },
        "S3OriginConfig": {
            "OriginAccessIdentity": ""
        }
    }
]
},
"OriginGroups": {
    "Quantity": 0
},
"DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-example",
    "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
            "Forward": "none"
        },
        "Headers": {
            "Quantity": 0
        },
        "QueryStringCacheKeys": {
            "Quantity": 0
        }
    },
    "TrustedSigners": {
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    }
},
```

```
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
      "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
  },
  "CacheBehaviors": {
    "Quantity": 0
  },
  "CustomErrorResponses": {
    "Quantity": 0
  },
  "Comment": "",
  "Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
  },
  "PriceClass": "PriceClass_All",
  "Enabled": true,
  "ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
  },
  "Restrictions": {
    "GeoRestriction": {
      "RestrictionType": "none",
      "Quantity": 0
    }
  },
  "WebACLId": "",
  "HttpVersion": "http2",
  "IsIPV6Enabled": true
}
}
```

- Para obter detalhes da API, consulte [GetDistributionConfig](#) na Referência de comandos da AWS CLI.

PowerShell

Ferramentas para PowerShell

Exemplo 1: recupera a configuração de uma distribuição específica.

```
Get-CFDistributionConfig -Id EXAMPLE0000ID
```

- Para obter detalhes da API, consulte [GetDistributionConfig](#) na AWS Tools for PowerShell Cmdlet Reference.

Python

SDK para Python (Boto3).

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

```
class CloudFrontWrapper:
    """Encapsulates Amazon CloudFront operations."""

    def __init__(self, cloudfront_client):
        """
        :param cloudfront_client: A Boto3 CloudFront client
        """
        self.cloudfront_client = cloudfront_client

    def update_distribution(self):
        distribution_id = input(
            "This script updates the comment for a CloudFront distribution.\n"
            "Enter a CloudFront distribution ID: "
        )

        distribution_config_response =
self.cloudfront_client.get_distribution_config(
    Id=distribution_id
)
```

```
distribution_config = distribution_config_response["DistributionConfig"]
distribution_etag = distribution_config_response["ETag"]

distribution_config["Comment"] = input(
    f"\n\nThe current comment for distribution {distribution_id} is "
    f"'{distribution_config['Comment']}'.\n\n"
    f"Enter a new comment: "
)
self.cloudfront_client.update_distribution(
    DistributionConfig=distribution_config,
    Id=distribution_id,
    IfMatch=distribution_etag,
)
print("Done!")
```

- Para obter detalhes da API, consulte [GetDistributionConfig](#) na Referência da API AWS SDK para Python (Boto3).

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar **ListCloudFrontOriginAccessIdentities** com uma CLI

Os exemplos de código a seguir mostram como usar o `ListCloudFrontOriginAccessIdentities`.

CLI

AWS CLI

Como listar as identidades do acesso de origem do CloudFront

O exemplo a seguir obtém uma lista das identidades do acesso de origem (OAIs) do CloudFront na sua conta da AWS:

```
aws cloudfront list-cloud-front-origin-access-identities
```

Saída:

```
{
  "CloudFrontOriginAccessIdentityList": {
    "Items": [
      {
        "Id": "E74FTE3AEXAMPLE",
        "S3CanonicalUserId":
"cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
        "Comment": "Example OAI"
      },
      {
        "Id": "EH1HDMBEXAMPLE",
        "S3CanonicalUserId":
"1489f6f2e6faacaae7ff64c4c3e6956c24f78788abfc1718c3527c263bf7a17EXAMPLE",
        "Comment": "Test OAI"
      },
      {
        "Id": "E2X2C9TEXAMPLE",
        "S3CanonicalUserId":
"cbfeebb915a64749f9be546a45b3fcfd3a31c779673c13c4dd460911ae402c2EXAMPLE",
        "Comment": "Example OAI #2"
      }
    ]
  }
}
```

- Para obter detalhes da API, consulte [ListCloudFrontOriginAccessIdentities](#) na Referência de comandos da AWS CLI.

PowerShell

Ferramentas para PowerShell

Exemplo 1: este exemplo retorna uma lista de identidades do acesso de origem do Amazon CloudFront. Como o parâmetro `-MaxItem` especifica um valor de 2, os resultados incluem duas identidades.

```
Get-CFCloudFrontOriginAccessIdentityList -MaxItem 2
```

Saída:

```
IsTruncated : True
```



```
Items      : {E326XXXXXXXXXT, E1YWXXXXXXXX9B}
Marker     :
MaxItems   : 2
NextMarker : E1YXXXXXXXXXX9B
Quantity   : 2
```

- Para obter detalhes da API, consulte [ListCloudFrontOriginAccessIdentities](#) na AWS Tools for PowerShell Cmdlet Reference.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar o **ListDistributions** com um AWS SDK ou a CLI

Os exemplos de código a seguir mostram como usar o ListDistributions.

CLI

AWS CLI

Como listar as distribuições do CloudFront

O seguinte exemplo obtém uma lista das distribuições do CloudFront em sua conta da AWS:

```
aws cloudfront list-distributions
```

Saída:

```
{
  "DistributionList": {
    "Items": [
      {
        "Id": "EMLARXS9EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/
EMLARXS9EXAMPLE",
        "Status": "InProgress",
        "LastModifiedTime": "2019-11-22T00:55:15.705Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
          "Enabled": false,
```

```
        "Quantity": 0
    },
    "DistributionConfig": {
        "CallerReference": "cli-example",
        "Aliases": {
            "Quantity": 0
        },
        "DefaultRootObject": "index.html",
        "Origins": {
            "Quantity": 1,
            "Items": [
                {
                    "Id": "awsexamplebucket.s3.amazonaws.com-cli-
example",
                    "DomainName":
"awsexamplebucket.s3.amazonaws.com",
                    "OriginPath": "",
                    "CustomHeaders": {
                        "Quantity": 0
                    },
                    "S3OriginConfig": {
                        "OriginAccessIdentity": ""
                    }
                }
            ]
        },
        "OriginGroups": {
            "Quantity": 0
        },
        "DefaultCacheBehavior": {
            "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-cli-
example",
            "ForwardedValues": {
                "QueryString": false,
                "Cookies": {
                    "Forward": "none"
                },
                "Headers": {
                    "Quantity": 0
                },
                "QueryStringCacheKeys": {
                    "Quantity": 0
                }
            }
        }
    },
```

```
    "TrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ],
      "CachedMethods": {
        "Quantity": 2,
        "Items": [
          "HEAD",
          "GET"
        ]
      }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
      "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
  },
  "CacheBehaviors": {
    "Quantity": 0
  },
  "CustomErrorResponses": {
    "Quantity": 0
  },
  "Comment": "",
  "Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
  },
  "PriceClass": "PriceClass_All",
  "Enabled": true,
```

```

    "ViewerCertificate": {
      "CloudFrontDefaultCertificate": true,
      "MinimumProtocolVersion": "TLSv1",
      "CertificateSource": "cloudfront"
    },
    "Restrictions": {
      "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
  }
},
{
  "Id": "EDFDVBD6EXAMPLE",
  "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
  "Status": "InProgress",
  "LastModifiedTime": "2019-12-04T23:35:41.433Z",
  "InProgressInvalidationBatches": 0,
  "DomainName": "d930174dauwrn8.cloudfront.net",
  "ActiveTrustedSigners": {
    "Enabled": false,
    "Quantity": 0
  },
  "DistributionConfig": {
    "CallerReference": "cli-example",
    "Aliases": {
      "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
      "Quantity": 1,
      "Items": [
        {
          "Id": "awsexamplebucket1.s3.amazonaws.com-cli-example",
          "DomainName":
            "awsexamplebucket1.s3.amazonaws.com",
          "OriginPath": "",
          "CustomHeaders": {

```

```
        "Quantity": 0
      },
      "S3OriginConfig": {
        "OriginAccessIdentity": ""
      }
    ]
  },
  "OriginGroups": {
    "Quantity": 0
  },
  "DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket1.s3.amazonaws.com-
cli-example",
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      },
      "Headers": {
        "Quantity": 0
      },
      "QueryStringCacheKeys": {
        "Quantity": 0
      }
    },
    "TrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ],
      "CachedMethods": {
        "Quantity": 2,
        "Items": [
          "HEAD",
          "GET"
        ]
      }
    }
  }
}
```

```

        }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
},
"Restrictions": {
    "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
    }
},
"WebACLId": "",
"HttpVersion": "http2",
"IsIPV6Enabled": true
}
},
{
    "Id": "E1X5IZQEXAMPLE",

```

```
    "ARN": "arn:aws:cloudfront::123456789012:distribution/
E1X5IZQEXAMPLE",
    "Status": "Deployed",
    "LastModifiedTime": "2019-11-06T21:31:48.864Z",
    "DomainName": "d2e04y12345678.cloudfront.net",
    "Aliases": {
      "Quantity": 0
    },
    "Origins": {
      "Quantity": 1,
      "Items": [
        {
          "Id": "awsexamplebucket2",
          "DomainName": "awsexamplebucket2.s3.us-
west-2.amazonaws.com",
          "OriginPath": "",
          "CustomHeaders": {
            "Quantity": 0
          },
          "S3OriginConfig": {
            "OriginAccessIdentity": ""
          }
        }
      ]
    },
    "OriginGroups": {
      "Quantity": 0
    },
    "DefaultCacheBehavior": {
      "TargetOriginId": "awsexamplebucket2",
      "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
          "Forward": "none"
        },
        "Headers": {
          "Quantity": 0
        },
        "QueryStringCacheKeys": {
          "Quantity": 0
        }
      },
      "TrustedSigners": {
        "Enabled": false,
```

```
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
},
"Restrictions": {
    "GeoRestriction": {
        "RestrictionType": "none",
```



```
        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "HTTP1_1",
    "IsIPV6Enabled": true
  }
]
}
```

- Para obter detalhes da API, consulte [ListDistributions](#) na Referência de comandos da AWS CLI.

PowerShell

Ferramentas para PowerShell

Exemplo 1: retorno de distribuições.

```
Get-CFDistributionList
```

- Para obter detalhes da API, consulte [ListDistributions](#) na AWS Tools for PowerShell Cmdlet Reference.

Python

SDK para Python (Boto3).

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

```
class CloudFrontWrapper:
    """Encapsulates Amazon CloudFront operations."""

    def __init__(self, cloudfront_client):
```

```
"""
:param cloudfront_client: A Boto3 CloudFront client
"""
self.cloudfront_client = cloudfront_client

def list_distributions(self):
    print("CloudFront distributions:\n")
    distributions = self.cloudfront_client.list_distributions()
    if distributions["DistributionList"]["Quantity"] > 0:
        for distribution in distributions["DistributionList"]["Items"]:
            print(f"Domain: {distribution['DomainName']}")
            print(f"Distribution Id: {distribution['Id']}")
            print(
                f"Certificate Source: "
                f"{distribution['ViewerCertificate']['CertificateSource']}"
            )
            if distribution["ViewerCertificate"]["CertificateSource"] ==
"acm":
                print(
                    f"Certificate: {distribution['ViewerCertificate']
['Certificate']}"
                )
            print("")
        else:
            print("No CloudFront distributions detected.")
```

- Para obter detalhes da API, consulte [ListDistributions](#) na Referência da API AWS SDK para Python (Boto3).

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar **UpdateDistribution** com o AWS SDK ou a CLI

Os exemplos de código a seguir mostram como usar o UpdateDistribution.

CLI

AWS CLI

Como atualizar o objeto raiz padrão de uma distribuição do CloudFront

O seguinte exemplo atualiza o objeto raiz padrão para `index.html` para a distribuição do CloudFront com o ID `EDFDVBD6EXAMPLE`:

```
aws cloudfront update-distribution --id EDFDVBD6EXAMPLE \  
  --default-root-object index.html
```

Saída:

```
{  
  "ETag": "E2QWRUHEXAMPLE",  
  "Distribution": {  
    "Id": "EDFDVBD6EXAMPLE",  
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",  
    "Status": "InProgress",  
    "LastModifiedTime": "2019-12-06T18:55:39.870Z",  
    "InProgressInvalidationBatches": 0,  
    "DomainName": "d1111111abcdef8.cloudfront.net",  
    "ActiveTrustedSigners": {  
      "Enabled": false,  
      "Quantity": 0  
    },  
    "DistributionConfig": {  
      "CallerReference": "6b10378d-49be-4c4b-a642-419ccaf8f3b5",  
      "Aliases": {  
        "Quantity": 0  
      },  
      "DefaultRootObject": "index.html",  
      "Origins": {  
        "Quantity": 1,  
        "Items": [  
          {  
            "Id": "example-website",  
            "DomainName": "www.example.com",  
            "OriginPath": "",  
            "CustomHeaders": {  
              "Quantity": 0  
            }  
          },  
        ]  
      },  
    }  
  }  
}
```

```
        "CustomOriginConfig": {
            "HTTPPort": 80,
            "HTTPSPort": 443,
            "OriginProtocolPolicy": "match-viewer",
            "OriginSslProtocols": {
                "Quantity": 2,
                "Items": [
                    "SSLv3",
                    "TLSv1"
                ]
            },
            "OriginReadTimeout": 30,
            "OriginKeepaliveTimeout": 5
        }
    ]
},
"OriginGroups": {
    "Quantity": 0
},
"DefaultCacheBehavior": {
    "TargetOriginId": "example-website",
    "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
            "Forward": "none"
        },
        "Headers": {
            "Quantity": 1,
            "Items": [
                "*"
            ]
        },
        "QueryStringCacheKeys": {
            "Quantity": 0
        }
    },
    "TrustedSigners": {
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
```

```
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
        "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
    "Quantity": 0
},
"CustomErrorResponses": {
    "Quantity": 0
},
"Comment": "",
"Logging": {
    "Enabled": false,
    "IncludeCookies": false,
    "Bucket": "",
    "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": true,
"ViewerCertificate": {
    "CloudFrontDefaultCertificate": true,
    "MinimumProtocolVersion": "TLSv1",
    "CertificateSource": "cloudfront"
},
"Restrictions": {
    "GeoRestriction": {
```

```

        "RestrictionType": "none",
        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "http1.1",
    "IsIPV6Enabled": true
  }
}

```

Como atualizar uma distribuição do CloudFront

O exemplo a seguir desabilita a distribuição do CloudFront com o ID EMLARXS9EXAMPLE fornecendo a configuração de distribuição em um arquivo JSON chamado `dist-config-disable.json`. Para atualizar uma distribuição, use a opção `--if-match` para fornecer o ETag da distribuição. Use o comando `get-distribution` ou `get-distribution-config` para obter o ETag.

Depois de usar o exemplo a seguir para desabilitar uma distribuição, use o comando `delete-distribution` para excluí-la.

```

aws cloudfront update-distribution \
  --id EMLARXS9EXAMPLE \
  --if-match E2QWRUHEXAMPLE \
  --distribution-config file://dist-config-disable.json

```

O arquivo `dist-config-disable.json` é um documento JSON na pasta atual que contém o seguinte. Observe que o campo `Enabled` está definido como `false`:

```

{
  "CallerReference": "cli-1574382155-496510",
  "Aliases": {
    "Quantity": 0
  },
  "DefaultRootObject": "index.html",
  "Origins": {
    "Quantity": 1,
    "Items": [
      {
        "Id": "awsexamplebucket.s3.amazonaws.com-1574382155-273939",
        "DomainName": "awsexamplebucket.s3.amazonaws.com",

```

```
        "OriginPath": "",
        "CustomHeaders": {
            "Quantity": 0
        },
        "S3OriginConfig": {
            "OriginAccessIdentity": ""
        }
    }
]
},
"OriginGroups": {
    "Quantity": 0
},
"DefaultCacheBehavior": {
    "TargetOriginId": "awsexamplebucket.s3.amazonaws.com-1574382155-273939",
    "ForwardedValues": {
        "QueryString": false,
        "Cookies": {
            "Forward": "none"
        },
        "Headers": {
            "Quantity": 0
        },
        "QueryStringCacheKeys": {
            "Quantity": 0
        }
    },
    "TrustedSigners": {
        "Enabled": false,
        "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
        "Quantity": 2,
        "Items": [
            "HEAD",
            "GET"
        ],
        "CachedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ]
        }
    }
}
```

```
    ]
  }
},
"SmoothStreaming": false,
"DefaultTTL": 86400,
"MaxTTL": 31536000,
"Compress": false,
"LambdaFunctionAssociations": {
  "Quantity": 0
},
"FieldLevelEncryptionId": ""
},
"CacheBehaviors": {
  "Quantity": 0
},
"CustomErrorResponses": {
  "Quantity": 0
},
"Comment": "",
"Logging": {
  "Enabled": false,
  "IncludeCookies": false,
  "Bucket": "",
  "Prefix": ""
},
"PriceClass": "PriceClass_All",
"Enabled": false,
"ViewerCertificate": {
  "CloudFrontDefaultCertificate": true,
  "MinimumProtocolVersion": "TLSv1",
  "CertificateSource": "cloudfront"
},
"Restrictions": {
  "GeoRestriction": {
    "RestrictionType": "none",
    "Quantity": 0
  }
},
"WebACLId": "",
"HttpVersion": "http2",
"IsIPV6Enabled": true
}
```


Saída:

```
{
  "ETag": "E9LHASXEXAMPLE",
  "Distribution": {
    "Id": "EMLARXS9EXAMPLE",
    "ARN": "arn:aws:cloudfront::123456789012:distribution/EMLARXS9EXAMPLE",
    "Status": "InProgress",
    "LastModifiedTime": "2019-12-06T18:32:35.553Z",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d1111111abcdef8.cloudfront.net",
    "ActiveTrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
  },
  "DistributionConfig": {
    "CallerReference": "cli-1574382155-496510",
    "Aliases": {
      "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
      "Quantity": 1,
      "Items": [
        {
          "Id":
"awsexamplebucket.s3.amazonaws.com-1574382155-273939",
          "DomainName": "awsexamplebucket.s3.amazonaws.com",
          "OriginPath": "",
          "CustomHeaders": {
            "Quantity": 0
          },
          "S3OriginConfig": {
            "OriginAccessIdentity": ""
          }
        }
      ]
    },
    "OriginGroups": {
      "Quantity": 0
    },
    "DefaultCacheBehavior": {
      "TargetOriginId":
"awsexamplebucket.s3.amazonaws.com-1574382155-273939",
```

```
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      },
      "Headers": {
        "Quantity": 0
      },
      "QueryStringCacheKeys": {
        "Quantity": 0
      }
    },
    "TrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
    "ViewerProtocolPolicy": "allow-all",
    "MinTTL": 0,
    "AllowedMethods": {
      "Quantity": 2,
      "Items": [
        "HEAD",
        "GET"
      ],
      "CachedMethods": {
        "Quantity": 2,
        "Items": [
          "HEAD",
          "GET"
        ]
      }
    },
    "SmoothStreaming": false,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000,
    "Compress": false,
    "LambdaFunctionAssociations": {
      "Quantity": 0
    },
    "FieldLevelEncryptionId": ""
  },
  "CacheBehaviors": {
    "Quantity": 0
  },
}
```

```
    "CustomErrorResponses": {
      "Quantity": 0
    },
    "Comment": "",
    "Logging": {
      "Enabled": false,
      "IncludeCookies": false,
      "Bucket": "",
      "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": false,
    "ViewerCertificate": {
      "CloudFrontDefaultCertificate": true,
      "MinimumProtocolVersion": "TLSv1",
      "CertificateSource": "cloudfront"
    },
    "Restrictions": {
      "GeoRestriction": {
        "RestrictionType": "none",
        "Quantity": 0
      }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
  }
}
```

- Para obter detalhes da API, consulte [UpdateDistribution](#) na Referência de comandos da AWS CLI.

Java

SDK para Java 2.x

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import software.amazon.awssdk.services.cloudfront.model.GetDistributionRequest;
import software.amazon.awssdk.services.cloudfront.model.GetDistributionResponse;
import software.amazon.awssdk.services.cloudfront.model.Distribution;
import software.amazon.awssdk.services.cloudfront.model.DistributionConfig;
import
    software.amazon.awssdk.services.cloudfront.model.UpdateDistributionRequest;
import software.amazon.awssdk.services.cloudfront.model.CloudFrontException;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-started.html
 */
public class ModifyDistribution {
    public static void main(String[] args) {
        final String usage = ""

            Usage:
                <id>\s

            Where:
                id - the id value of the distribution.\s
            """;

        if (args.length != 1) {
            System.out.println(usage);
            System.exit(1);
        }

        String id = args[0];
        CloudFrontClient cloudFrontClient = CloudFrontClient.builder()
            .region(Region.AWS_GLOBAL)
            .build();

        modDistribution(cloudFrontClient, id);
        cloudFrontClient.close();
    }
}
```

```
public static void modDistribution(CloudFrontClient cloudFrontClient, String
idVal) {
    try {
        // Get the Distribution to modify.
        GetDistributionRequest disRequest = GetDistributionRequest.builder()
            .id(idVal)
            .build();

        GetDistributionResponse response =
cloudFrontClient.getDistribution(disRequest);
        Distribution disObject = response.distribution();
        DistributionConfig config = disObject.distributionConfig();

        // Create a new DistributionConfig object and add new values to
comment and
        // aliases
        DistributionConfig config1 = DistributionConfig.builder()
            .aliases(config.aliases()) // You can pass in new values here
            .comment("New Comment")
            .cacheBehaviors(config.cacheBehaviors())
            .priceClass(config.priceClass())
            .defaultCacheBehavior(config.defaultCacheBehavior())
            .enabled(config.enabled())
            .callerReference(config.callerReference())
            .logging(config.logging())
            .originGroups(config.originGroups())
            .origins(config.origins())
            .restrictions(config.restrictions())
            .defaultRootObject(config.defaultRootObject())
            .webACLId(config.webACLId())
            .httpVersion(config.httpVersion())
            .viewerCertificate(config.viewerCertificate())
            .customErrorResponses(config.customErrorResponses())
            .build();

        UpdateDistributionRequest updateDistributionRequest =
UpdateDistributionRequest.builder()
            .distributionConfig(config1)
            .id(disObject.id())
            .ifMatch(response.eTag())
            .build();

        cloudFrontClient.updateDistribution(updateDistributionRequest);
    }
}
```

```
        } catch (CloudFrontException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
}
```

- Para obter detalhes da API, consulte [UpdateDistribution](#) na Referência da API AWS SDK for Java 2.x.

Python

SDK para Python (Boto3).

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

```
class CloudFrontWrapper:
    """Encapsulates Amazon CloudFront operations."""

    def __init__(self, cloudfront_client):
        """
        :param cloudfront_client: A Boto3 CloudFront client
        """
        self.cloudfront_client = cloudfront_client

    def update_distribution(self):
        distribution_id = input(
            "This script updates the comment for a CloudFront distribution.\n"
            "Enter a CloudFront distribution ID: "
        )

        distribution_config_response =
self.cloudfront_client.get_distribution_config(
    Id=distribution_id
```

```
)
distribution_config = distribution_config_response["DistributionConfig"]
distribution_etag = distribution_config_response["ETag"]

distribution_config["Comment"] = input(
    f"\n\nThe current comment for distribution {distribution_id} is "
    f"'{distribution_config['Comment']}'.\n\n"
    f"Enter a new comment: "
)
self.cloudfront_client.update_distribution(
    DistributionConfig=distribution_config,
    Id=distribution_id,
    IfMatch=distribution_etag,
)
print("Done!")
```

- Para obter detalhes da API, consulte [UpdateDistribution](#) na Referência da API AWS SDK para Python (Boto3).

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Cenários para o CloudFront usando SDKs da AWS

Os exemplos de código a seguir mostram como implementar cenários comuns no CloudFront com os SDKs da AWS. Esses casos mostram como realizar tarefas específicas chamando várias funções no CloudFront ou combinadas com outros Serviços da AWS. Cada cenário inclui um link para o código-fonte completo, em que você pode encontrar instruções sobre como configurar e executar o código.

Os cenários têm como alvo um nível intermediário de experiência para ajudá-lo a entender as ações de serviço no contexto.

Exemplos

- [Exclua recursos de assinatura do CloudFront usando o AWS SDK](#)
- [Criar URLs e cookies assinados usando um AWS SDK](#)

Exclua recursos de assinatura do CloudFront usando o AWS SDK

O exemplo de código a seguir mostra como excluir recursos que são usados para obter acesso a conteúdo restrito em um bucket do Amazon Simple Storage Service (Amazon S3).

Java

SDK para Java 2.x

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import software.amazon.awssdk.services.cloudfront.model.DeleteKeyGroupResponse;
import
    software.amazon.awssdk.services.cloudfront.model.DeleteOriginAccessControlResponse;
import software.amazon.awssdk.services.cloudfront.model.DeletePublicKeyResponse;
import software.amazon.awssdk.services.cloudfront.model.GetKeyGroupResponse;
import
    software.amazon.awssdk.services.cloudfront.model.GetOriginAccessControlResponse;
import software.amazon.awssdk.services.cloudfront.model.GetPublicKeyResponse;

public class DeleteSigningResources {
    private static final Logger logger =
        LoggerFactory.getLogger(DeleteSigningResources.class);

    public static void deleteOriginAccessControl(final CloudFrontClient
        cloudFrontClient,
        final String originAccessControlId) {
        GetOriginAccessControlResponse getResponse = cloudFrontClient
            .getOriginAccessControl(b -> b.id(originAccessControlId));
        DeleteOriginAccessControlResponse deleteResponse =
            cloudFrontClient.deleteOriginAccessControl(builder -> builder
                .id(originAccessControlId)
                .ifMatch(getResponse.eTag()));
        if (deleteResponse.sdkHttpResponse().isSuccessful()) {
```



```
        logger.info("Successfully deleted Origin Access Control [{}]",
originAccessControlId);
    }
}

    public static void deleteKeyGroup(final CloudFrontClient cloudFrontClient,
final String keyGroupId) {

        GetKeyGroupResponse getResponse = cloudFrontClient.getKeyGroup(b ->
b.id(keyGroupId));
        DeleteKeyGroupResponse deleteResponse =
cloudFrontClient.deleteKeyGroup(builder -> builder
            .id(keyGroupId)
            .ifMatch(getResponse.eTag()));
        if (deleteResponse.sdkHttpResponse().isSuccessful()) {
            logger.info("Successfully deleted Key Group [{}]", keyGroupId);
        }
    }

    public static void deletePublicKey(final CloudFrontClient cloudFrontClient,
final String publicKeyId) {
        GetPublicKeyResponse getResponse = cloudFrontClient.getPublicKey(b ->
b.id(publicKeyId));

        DeletePublicKeyResponse deleteResponse =
cloudFrontClient.deletePublicKey(builder -> builder
            .id(publicKeyId)
            .ifMatch(getResponse.eTag()));

        if (deleteResponse.sdkHttpResponse().isSuccessful()) {
            logger.info("Successfully deleted Public Key [{}]", publicKeyId);
        }
    }
}
```

- Para obter detalhes da API, consulte os tópicos a seguir na Referência da API AWS SDK for Java 2.x.
 - [DeleteKeyGroup](#)
 - [DeleteOriginAccessControl](#)
 - [DeletePublicKey](#)

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Criar URLs e cookies assinados usando um AWS SDK

O exemplo de código a seguir mostra como criar URLs e cookies assinados que permitem acesso a recursos restritos.

Java

SDK para Java 2.x

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [Repositório de exemplos de código da AWS](#).

Usar a classe [CannedSignerRequest](#) para assinar URLs ou cookies com uma política predefinida.

```
import software.amazon.awssdk.services.cloudfront.model.CannedSignerRequest;

import java.net.URL;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.time.Instant;
import java.time.temporal.ChronoUnit;

public class CreateCannedPolicyRequest {

    public static CannedSignerRequest createRequestForCannedPolicy(String
distributionDomainName,
        String fileNameToUpload,
        String privateKeyFullPath, String publicKeyId) throws Exception {
        String protocol = "https";
        String resourcePath = "/" + fileNameToUpload;

        String cloudFrontUrl = new URL(protocol, distributionDomainName,
resourcePath).toString();
        Instant expirationDate = Instant.now().plus(7, ChronoUnit.DAYS);
```

```
    Path path = Paths.get(privateKeyFullPath);

    return CannedSignerRequest.builder()
        .resourceUrl(cloudFrontUrl)
        .privateKey(path)
        .keyPairId(publicKeyId)
        .expirationDate(expirationDate)
        .build();
}
}
```

Usar a classe [CustomSignerRequest](#) para assinar URLs ou cookies com uma política personalizada. O `activeDate` e `ipRange` são métodos opcionais.

```
import software.amazon.awssdk.services.cloudfront.model.CustomSignerRequest;

import java.net.URL;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.time.Instant;
import java.time.temporal.ChronoUnit;

public class CreateCustomPolicyRequest {

    public static CustomSignerRequest createRequestForCustomPolicy(String
distributionDomainName,
        String fileNameToUpload,
        String privateKeyFullPath, String publicKeyId) throws Exception {
        String protocol = "https";
        String resourcePath = "/" + fileNameToUpload;

        String cloudFrontUrl = new URL(protocol, distributionDomainName,
resourcePath).toString();
        Instant expireDate = Instant.now().plus(7, ChronoUnit.DAYS);
        // URL will be accessible tomorrow using the signed URL.
        Instant activeDate = Instant.now().plus(1, ChronoUnit.DAYS);
        Path path = Paths.get(privateKeyFullPath);

        return CustomSignerRequest.builder()
            .resourceUrl(cloudFrontUrl)
            .privateKey(path)
            .keyPairId(publicKeyId)
```

```
        .expirationDate(expireDate)
        .activeDate(activeDate) // Optional.
        // .ipRange("192.168.0.1/24") // Optional.
        .build();
    }
}
```

O exemplo a seguir demonstra o uso da classe [CloudFrontUtilities](#) para produzir cookies e URLs assinados. [Veja](#) esse exemplo de código no GitHub.

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import software.amazon.awssdk.services.cloudfront.CloudFrontUtilities;
import software.amazon.awssdk.services.cloudfront.cookie.CookiesForCannedPolicy;
import software.amazon.awssdk.services.cloudfront.cookie.CookiesForCustomPolicy;
import software.amazon.awssdk.services.cloudfront.model.CannedSignerRequest;
import software.amazon.awssdk.services.cloudfront.model.CustomSignerRequest;
import software.amazon.awssdk.services.cloudfront.url.SignedUrl;

public class SigningUtilities {
    private static final Logger logger =
        LoggerFactory.getLogger(SigningUtilities.class);
    private static final CloudFrontUtilities cloudFrontUtilities =
        CloudFrontUtilities.create();

    public static SignedUrl signUrlForCannedPolicy(CannedSignerRequest
        cannedSignerRequest) {
        SignedUrl signedUrl =
            cloudFrontUtilities.getSignedUrlWithCannedPolicy(cannedSignerRequest);
        logger.info("Signed URL: [{}]", signedUrl.url());
        return signedUrl;
    }

    public static SignedUrl signUrlForCustomPolicy(CustomSignerRequest
        customSignerRequest) {
        SignedUrl signedUrl =
            cloudFrontUtilities.getSignedUrlWithCustomPolicy(customSignerRequest);
        logger.info("Signed URL: [{}]", signedUrl.url());
        return signedUrl;
    }
}
```

```
public static CookiesForCannedPolicy
getCookiesForCannedPolicy(CannedSignerRequest cannedSignerRequest) {
    CookiesForCannedPolicy cookiesForCannedPolicy = cloudFrontUtilities
        .getCookiesForCannedPolicy(cannedSignerRequest);
    logger.info("Cookie EXPIRES header [{}]",
cookiesForCannedPolicy.expiresHeaderValue());
    logger.info("Cookie KEYPAIR header [{}]",
cookiesForCannedPolicy.keyPairIdHeaderValue());
    logger.info("Cookie SIGNATURE header [{}]",
cookiesForCannedPolicy.signatureHeaderValue());
    return cookiesForCannedPolicy;
}

public static CookiesForCustomPolicy
getCookiesForCustomPolicy(CustomSignerRequest customSignerRequest) {
    CookiesForCustomPolicy cookiesForCustomPolicy = cloudFrontUtilities
        .getCookiesForCustomPolicy(customSignerRequest);
    logger.info("Cookie POLICY header [{}]",
cookiesForCustomPolicy.policyHeaderValue());
    logger.info("Cookie KEYPAIR header [{}]",
cookiesForCustomPolicy.keyPairIdHeaderValue());
    logger.info("Cookie SIGNATURE header [{}]",
cookiesForCustomPolicy.signatureHeaderValue());
    return cookiesForCustomPolicy;
}
}
```

- Para obter detalhes da API, consulte [CloudFrontUtilities](#) na Referência da API AWS SDK for Java 2.x.

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Exemplos do CloudFront Functions para o CloudFront usando SDKs da AWS

Os exemplos de código a seguir mostram como usar o CloudFront com SDKs da AWS.

Exemplos

- [Adicionar cabeçalhos de segurança HTTP a um evento de resposta do visualizador do CloudFront Functions](#)
- [Adicionar um cabeçalho de CORS a um evento de resposta do visualizador do CloudFront Functions](#)
- [Adicionar um cabeçalho de controle de cache a um evento de resposta do visualizador do CloudFront Functions](#)
- [Adicione um cabeçalho IP de cliente verdadeiro a um evento de solicitação do visualizador do CloudFront Functions](#)
- [Adicionar um cabeçalho de origem a um evento de resposta do visualizador do CloudFront Functions](#)
- [Adicionar index.html aos URLs de solicitação sem um nome de arquivo em um evento de solicitação do visualizador do CloudFront Functions](#)
- [Normalizar os parâmetros de string de consulta em uma solicitação do visualizador do CloudFront Functions](#)
- [Redirecionar para um novo URL em um evento de solicitação do visualizador do CloudFront Functions](#)
- [Reescrever um URI de solicitação com base na configuração do KeyValueStore para um evento de solicitação do visualizador do CloudFront Functions](#)
- [Usar pares de chave-valor em uma solicitação do visualizador do CloudFront Functions](#)
- [Validar um token simples em uma solicitação do visualizador do CloudFront Functions](#)

Adicionar cabeçalhos de segurança HTTP a um evento de resposta do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como adicionar cabeçalhos de segurança HTTP a um evento de resposta do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
function handler(event) {
  var response = event.response;
  var headers = response.headers;

  // Set HTTP security headers
  // Since JavaScript doesn't allow for hyphens in variable names, we use the
  dict["key"] notation
  headers['strict-transport-security'] = { value: 'max-age=63072000;
includeSubdomains; preload'};
  headers['content-security-policy'] = { value: "default-src 'none'; img-src
'self'; script-src 'self'; style-src 'self'; object-src 'none'; frame-ancestors
'none'"};
  headers['x-content-type-options'] = { value: 'nosniff'};
  headers['x-frame-options'] = {value: 'DENY'};
  headers['x-xss-protection'] = {value: '1; mode=block'};
  headers['referrer-policy'] = {value: 'same-origin'};

  // Return the response to viewers
  return response;
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Adicionar um cabeçalho de CORS a um evento de resposta do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como adicionar um cabeçalho de CORS a um evento de resposta do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
function handler(event) {
    var request = event.request;
    var response = event.response;

    // If Access-Control-Allow-Origin CORS header is missing, add it.
    // Since JavaScript doesn't allow for hyphens in variable names, we use the
    // dict["key"] notation.
    if (!response.headers['access-control-allow-origin'] &&
        request.headers['origin']) {
        response.headers['access-control-allow-origin'] = {value:
            request.headers['origin'].value};
        console.log("Access-Control-Allow-Origin was missing, adding it now.");
    }

    return response;
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Adicionar um cabeçalho de controle de cache a um evento de resposta do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como adicionar um cabeçalho de controle de cache a um evento de resposta do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
function handler(event) {
  var response = event.response;
  var headers = response.headers;

  if (response.statusCode >= 200 && response.statusCode < 400) {
    // Set the cache-control header
    headers['cache-control'] = {value: 'public, max-age=63072000'};
  }

  // Return response to viewers
  return response;
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Adicione um cabeçalho IP de cliente verdadeiro a um evento de solicitação do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como adicionar um cabeçalho de IP de cliente verdadeiro a um evento de resposta do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
function handler(event) {
  var request = event.request;
  var clientIP = event.viewer.ip;

  //Add the true-client-ip header to the incoming request
  request.headers['true-client-ip'] = {value: clientIP};

  return request;
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Adicionar um cabeçalho de origem a um evento de resposta do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como adicionar um cabeçalho de origem a um evento de resposta do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
function handler(event) {
  var request = event.request;
  var headers = request.headers;
  var host = request.headers.host.value;

  // If origin header is missing, set it equal to the host header.
  if (!headers.origin)
    headers.origin = {value: `https://${host}`};

  return request;
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Adicionar index.html aos URLs de solicitação sem um nome de arquivo em um evento de solicitação do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como adicionar index.html a URLs de solicitação sem um nome de arquivo em um evento de solicitação do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
function handler(event) {
  var request = event.request;
  var uri = request.uri;

  // Check whether the URI is missing a file name.
  if (uri.endsWith('/')) {
    request.uri += 'index.html';
  }
}
```

```
    }  
    // Check whether the URI is missing a file extension.  
    else if (!uri.includes('.')) {  
        request.uri += '/index.html';  
    }  
  
    return request;  
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Normalizar os parâmetros de string de consulta em uma solicitação do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como normalizar os parâmetros de string de consulta em uma solicitação do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
function handler(event) {  
    var qs=[];  
    for (var key in event.request.querystring) {  
        if (event.request.querystring[key].multiValue) {  
            event.request.querystring[key].multiValue.forEach((mv) =>  
{qs.push(key + "=" + mv.value)});  
        } else {  
            qs.push(key + "=" + event.request.querystring[key].value);  
        }  
    }  
};
```

```
    event.request.querystring = qs.sort().join('&');

    return event.request;
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Redirecionar para um novo URL em um evento de solicitação do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como redirecionar para um novo URL em um evento de solicitação do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
function handler(event) {
    var request = event.request;
    var headers = request.headers;
    var host = request.headers.host.value;
    var country = 'DE' // Choose a country code
    var newurl = `https://${host}/de/index.html` // Change the redirect URL to
    your choice

    if (headers['cloudfront-viewer-country']) {
        var countryCode = headers['cloudfront-viewer-country'].value;
        if (countryCode === country) {
            var response = {
                statusCode: 302,
```

```
        statusDescription: 'Found',
        headers:
            { "location": { "value": newurl } }
    }

    return response;
}
}
return request;
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Reescrever um URI de solicitação com base na configuração do KeyValueStore para um evento de solicitação do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como reescrever um URI de solicitação com base na configuração do KeyValueStore para um evento de solicitação do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
import cf from 'cloudfront';

// (Optional) Replace KVS_ID with actual KVS ID
const kvsId = "KVS_ID";
// enable stickiness by setting a cookie from origin or using another edge
function
```

```
const stickinessCookieName = "appversion";
// set to true to enable console logging
const loggingEnabled = false;

// function rewrites the request uri based on configuration in KVS
// example config in KVS in key:value format
// "latest": {"a_weightage": .8, "a_url": "v1", "b_url": "v2"}
// given above key and value in KVS the request uri will be rewritten
// for example http(s)://domain/latest/something/else will be rewritten as
// http(s)://domain/v1/something/else or http(s)://domain/v2/something/else
// depending on weightage
// if no configuration is found, then the request is returned as is
async function handler(event) {
  // NOTE: This example function is for a viewer request event trigger.
  // Choose viewer request for event trigger when you associate this function
  // with a distribution.
  const request = event.request;
  const pathSegments = request.uri.split('/');
  const key = pathSegments[1];

  // if empty path segment or if there is valid stickiness cookie
  // then skip call to KVS and let the request continue.
  if (!key || hasValidStickinessCookie(request.cookies[stickinessCookieName],
key)) {
    return event.request;
  }

  try {
    // get the prefix replacement from KVS
    const replacement = await getPathPrefixByWeightage(key);
    if (!replacement) {
      return event.request;
    }
    //Replace the first path with the replacement
    pathSegments[1] = replacement;
    log(`using prefix ${pathSegments[1]}`);
    const newUri = pathSegments.join('/');
    log(`${request.uri} -> ${newUri}`);
    request.uri = newUri;

    return request;
  } catch (err) {
    // No change to the path if the key is not found or any other error
    log(`request uri: ${request.uri}, error: ${err}`);
  }
}
```

```
    }
    // no change to path - return request
    return event.request;
}

// function to get the prefix from KVS
async function getPathPrefixByWeightage(key) {
    const kvsHandle = cf.kvs(kvsId);
    // get the weightage config from KVS
    const kvsResponse = await kvsHandle.get(key);
    const weightageConfig = JSON.parse(kvsResponse);
    // no configuration - return null
    if (!weightageConfig || !isFinite(weightageConfig.a_weightage)) {
        return null;
    }
    // return the url based on weightage
    // return null if no url is configured
    if (Math.random() <= weightageConfig.a_weightage) {
        return weightageConfig.a_url ? weightageConfig.a_url : null;
    } else {
        return weightageConfig.b_url ? weightageConfig.b_url : null;
    }
}

// function to check if the stickiness cookie is valid
function hasValidStickinessCookie(stickinessCookie, pathSegment) {
    // if the value exists and it matches pathSegment
    return (stickinessCookie && stickinessCookie.value === pathSegment)
}

function log(message) {
    if (loggingEnabled) {
        console.log(message);
    }
}
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Usar pares de chave-valor em uma solicitação do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como usar pares de chave-valor em uma solicitação do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
import cf from 'cloudfront';

// This fails if there is no key value store associated with the function
const kvsHandle = cf.kvs();

// Remember to associate the KVS with your function before referencing KVS in
// your code.
// https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/kvs-with-
// functions-associate.html
async function handler(event) {
  const request = event.request;
  // Use the first segment of the pathname as key
  // For example http(s)://domain/<key>/something/else
  const pathSegments = request.uri.split('/')
  const key = pathSegments[1]
  try {
    // Replace the first path of the pathname with the value of the key
    // For example http(s)://domain/<value>/something/else
    pathSegments[1] = await kvsHandle.get(key);
    const newUri = pathSegments.join('/');
    console.log(`${request.uri} -> ${newUri}`)
    request.uri = newUri;
  } catch (err) {
    // No change to the pathname if the key is not found
    console.log(`${request.uri} | ${err}`);
  }
}
```

```
    }  
    return request;  
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Validar um token simples em uma solicitação do visualizador do CloudFront Functions

O exemplo de código a seguir mostra como validar um token simples em uma solicitação do visualizador do CloudFront Functions.

JavaScript

Runtime 2.0 do JavaScript para o CloudFront Functions

Note

Há mais no GitHub. Encontre o exemplo completo e saiba como configurar e executar no [repositório de exemplos do CloudFront Functions](#).

```
import crypto from 'crypto';  
import cf from 'cloudfront';  
  
//Response when JWT is not valid.  
const response401 = {  
  statusCode: 401,  
  statusDescription: 'Unauthorized'  
};  
  
// Remember to associate the KVS with your function before calling the const  
kvsKey = 'jwt.secret'.  
// https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/kvs-with-  
functions-associate.html  
const kvsKey = 'jwt.secret';  
// set to true to enable console logging
```

```
const loggingEnabled = false;

function jwt_decode(token, key, noVerify, algorithm) {
  // check token
  if (!token) {
    throw new Error('No token supplied');
  }
  // check segments
  const segments = token.split('.');
  if (segments.length !== 3) {
    throw new Error('Not enough or too many segments');
  }

  // All segment should be base64
  const headerSeg = segments[0];
  const payloadSeg = segments[1];
  const signatureSeg = segments[2];

  // base64 decode and parse JSON
  const payload = JSON.parse(_base64urlDecode(payloadSeg));

  if (!noVerify) {
    const signingMethod = 'sha256';
    const signingType = 'hmac';

    // Verify signature. `sign` will return base64 string.
    const signingInput = [headerSeg, payloadSeg].join('.');

    if (!_verify(signingInput, key, signingMethod, signingType,
signatureSeg)) {
      throw new Error('Signature verification failed');
    }

    // Support for nbf and exp claims.
    // According to the RFC, they should be in seconds.
    if (payload.nbf && Date.now() < payload.nbf*1000) {
      throw new Error('Token not yet active');
    }

    if (payload.exp && Date.now() > payload.exp*1000) {
      throw new Error('Token expired');
    }
  }
}
```

```
    return payload;
  }

  //Function to ensure a constant time comparison to prevent
  //timing side channels.
  function _constantTimeEquals(a, b) {
    if (a.length !== b.length) {
      return false;
    }

    let xor = 0;
    for (let i = 0; i < a.length; i++) {
      xor |= (a.charCodeAt(i) ^ b.charCodeAt(i));
    }

    return 0 === xor;
  }

  function _verify(input, key, method, type, signature) {
    if(type === "hmac") {
      return _constantTimeEquals(signature, _sign(input, key, method));
    }
    else {
      throw new Error('Algorithm type not recognized');
    }
  }

  function _sign(input, key, method) {
    return crypto.createHmac(method, key).update(input).digest('base64url');
  }

  function _base64urlDecode(str) {
    return Buffer.from(str, 'base64url')
  }

  async function handler(event) {
    let request = event.request;

    //Secret key used to verify JWT token.
    //Update with your own key.
    const secret_key = await getSecret()

    if(!secret_key) {
```

```
    return response401;
  }

  // If no JWT token, then generate HTTP redirect 401 response.
  if(!request.querystring.jwt) {
    log("Error: No JWT in the querystring");
    return response401;
  }

  const jwtToken = request.querystring.jwt.value;

  try{
    jwt_decode(jwtToken, secret_key);
  }
  catch(e) {
    log(e);
    return response401;
  }

  //Remove the JWT from the query string if valid and return.
  delete request.querystring.jwt;
  log("Valid JWT token");
  return request;
}

// get secret from key value store
async function getSecret() {
  // initialize cloudfront kv store and get the key value
  try {
    const kvsHandle = cf.kvs();
    return await kvsHandle.get(kvsKey);
  } catch (err) {
    log(`Error reading value for key: ${kvsKey}, error: ${err}`);
    return null;
  }
}

function log(message) {
  if (loggingEnabled) {
    console.log(message);
  }
}
}
```

Para ver uma lista completa dos Guias do desenvolvedor de SDK da AWS e exemplos de código, consulte [Como usar o CloudFront com um SDK da AWS](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Histórico do documento

A tabela a seguir descreve as alterações importantes feitas na documentação do CloudFront. Para receber notificações sobre atualizações, [inscreva-se no feed RSS](#).

Alteração	Descrição	Data
Avaliar com o AWS Config Rules	Avalie as configurações do CloudFront com o AWS Config Rules.	20 de setembro de 2024
Ampliação do conteúdo de solução de problemas	Ampliação do conteúdo de solução de problemas para códigos de status de resposta de erro HTTP 4xx e 5xx.	26 de agosto de 2024
Adição de novas políticas de cache gerenciadas	Adição de novas políticas de cache gerenciadas <code>UseOriginCacheControlHeaders</code> e <code>UseOriginCacheControlHeader-QueryStrings</code> .	24 de maio de 2024
Adição de compatibilidade ao controle de acesso à origem	Agora é possível criar um controle de acesso à origem (OAC) para AWS Elemental MediaPackage V2 e um URL da função do AWS Lambda.	11 de abril de 2024
Campos de log em tempo real para CMCD	Adição de 18 campos Common Media Client Data (CMCD) para registro em log em tempo real.	9 de abril de 2024
Conceitos básicos de uma distribuição básica do CloudFront	Atualização de tutorial de uma distribuição básica que usa uma origem do Amazon S3	18 de março de 2024

com um controle de acesso à origem (OAC).

[Exemplos de código para o CloudFront usando SDKs da AWS](#)

Adição de exemplos de código que mostram como usar o CloudFront com um Kit de Desenvolvimento de Software (SDK) da AWS. Os exemplos são divididos em trechos de código que mostram como chamar funções de serviço individuais e exemplos que mostram como realizar uma tarefa específica chamando várias funções dentro do mesmo serviço.

16 de fevereiro de 2024

[Atualização da política gerenciada da AWS](#)

As políticas CloudFrontReadOnlyAccess e CloudFrontFullAccess do IAM agora são compatíveis com as operações KeyValueStore .

19 de dezembro de 2023

[Runtime 2.0 do JavaScript](#)

Adição de atributos de runtime 2.0 do JavaScript para CloudFront Functions.

21 de novembro de 2023

KeyValueStore do CloudFront	O Amazon CloudFront já é compatível com o KeyValueStore do CloudFront. Esse recurso é um datastore de valor de chave seguro, global e de baixa latência que permite acesso de leitura por meio do CloudFront Functions . É possível habilitar a lógica personalizável avançada nos locais da borda do CloudFront.	21 de novembro de 2023
Lambda@Edge oferece suporte a versões de tempo de execução mais recentes	O Lambda@Edge já é compatível com funções do Lambda com o runtime do Node.js 20.	15 de novembro de 2023
Painel de segurança	O CloudFront cria um painel de segurança quando você cria uma distribuição. Habilite o AWS WAF, gerencie restrições geográficas e visualize dados gerais para solicitações, bots e logs.	8 de novembro de 2023
Classificar strings de consulta em funções	O CloudFront agora permite a classificação de strings de consulta usando o CloudFront Functions.	3 de outubro de 2023
Recomendações de segurança do AWS WAF	O Amazon CloudFront agora exibe recomendações de segurança do AWS WAF no console do CloudFront.	26 de setembro de 2023

Suporte para o fornecimento de conteúdo de cache obsoleto (expirado)	O CloudFront oferece suporte às diretivas de controle de cache <code>Stale-While-Revalidate</code> e <code>Stale-If-Error</code> .	15 de maio de 2023
Habilite proteções do AWS WAF com um clique	Um método simplificado para adicionar proteções de segurança do AWS WAF às distribuições do CloudFront.	10 de maio de 2023
Habilitar ACLs para novos buckets do S3 usados para logs padrão	Adição de notas e links para abordar a configuração da ACL padrão para novos buckets do S3.	11 de abril de 2023
Criar uma origem usando o Amazon S3 Object Lambda	É possível usar um alias de ponto de acesso do Amazon S3 Object Lambda como uma origem para sua distribuição.	31 de março de 2023
Personalizar o status HTTP e o corpo usando o CloudFront Functions	É possível usar o CloudFront Functions para atualizar o código de status da resposta do visualizador e substituir ou remover o corpo da resposta.	29 de março de 2023
Adição de opções de curinga de cabeçalhos de CORS para portas	Agora é possível incluir várias configurações de curinga para portas nos cabeçalhos de controle de acesso de CORS.	20 de março de 2023
Adição de novo link para o Guia do usuário do AWS Security Hub	Atualização de idioma e adição de link para os controles reorganizados do Amazon CloudFront no Guia do usuário do AWS Security Hub.	9 de março de 2023

[O CloudFront agora é compatível com as listas de bloqueio \(“todas exceto”\) nas políticas de solicitação de origem](#)

Use listas de bloqueio nas políticas de solicitação de origem para incluir todas as strings de consulta, cabeçalhos HTTP ou cookies, exceto para os especificados, nas solicitações que o CloudFront envia à origem.

22 de fevereiro de 2023

[O CloudFront adiciona uma nova política de solicitação de origem gerenciada para encaminhar todos os cabeçalhos do visualizador, exceto o cabeçalho de Host](#)

Use a nova política de solicitação de origem gerenciada do CloudFront para incluir todos os cabeçalhos da solicitação do visualizador, exceto o cabeçalho de Host, nas solicitações que o CloudFront envia à origem.

22 de fevereiro de 2023

[Restrições atualizadas ao Lambda@Edge](#)

O Lambda@Edge é compatível com configurações de gerenciamento de ambiente de tempo de execução do Lambda definidas como Auto (Automáticas).

16 de fevereiro de 2023

[Orientação do IAM atualizada para o CloudFront](#)

Guia atualizado para alinhamento com as práticas recomendadas do IAM. Para obter mais informações, consulte [Práticas recomendadas de segurança no IAM](#).

15 de fevereiro de 2023

[Segurança aprimorada com controle de acesso de origem](#)

Agora é possível proteger as origens do MediaStore permitindo o acesso somente às distribuições designadas do CloudFront.

9 de fevereiro de 2023

<u>Novos cabeçalhos para determinar a estrutura de cabeçalho do visualizador</u>	Agora você pode adicionar ordem de cabeçalhos e contagem de cabeçalhos para ajudar a identificar o visualizador com base nos cabeçalhos que ele envia.	13 de janeiro de 2023
<u>Lambda@Edge oferece suporte a versões de tempo de execução mais recentes</u>	O Lambda@Edge agora oferece suporte para funções do Lambda com o tempo de execução do Node.js 18.	12 de janeiro de 2023
<u>Remover cabeçalhos de resposta usando uma política de cabeçalhos de resposta</u>	Agora é possível usar uma política de cabeçalhos de resposta do CloudFront para remover cabeçalhos que o CloudFront recebeu na resposta da origem. Os cabeçalhos especificados não são incluídos na resposta que o CloudFront envia aos visualizadores.	3 de janeiro de 2023
<u>Implantação contínua para testar com segurança as alterações de configuração</u>	Agora você pode implantar alterações em sua configuração de CDN por meio de testes com um subconjunto do tráfego de produção.	18 de novembro de 2022
<u>Versão do cabeçalho CloudFront-Viewer-JA3-Fingerprint</u>	Agora você pode usar a impressão digital JA3 para ajudar a determinar se a solicitação vem de um cliente conhecido.	16 de novembro de 2022

Adição de opções de curinga de cabeçalhos de CORS	Agora é possível usar várias configurações de curinga em alguns cabeçalhos de controle de acesso de CORS.	11 de novembro de 2022
Métricas adicionais para distribuições do CloudFront	Suporte para Monitorin gSubscription na API do CloudFront e no AWS CloudFormation.	3 de outubro de 2022
Segurança aprimorada com controle de acesso de origem	Agora você pode proteger as origens do Amazon S3 permitindo o acesso somente às distribuições designadas do CloudFront.	24 de agosto de 2022
Suporte a HTTP/3 para distribuições do CloudFront	Agora é possível escolher HTTP/3 para sua distribuição do CloudFront.	15 de agosto de 2022
Adicionar detalhes de handshake ao cabeçalho CloudFront-Viewer-TLS	Agora é possível visualiza r informações sobre o handshake SSL/TLS usado.	27 de junho de 2022
Nova métrica em cabeçalho Server-Timing	Adição da nova métrica cdn-downstream-fb1 a cabeçalhos Server-Ti ming .	13 de junho de 2022
Novo cabeçalho para obter informações sobre a versão TLS e a cifra	Agora é possível usar o CloudFront-Viewer-TLS para obter informaçõ es sobre a versão do TLS (ou SSL) e a criptografia usada para a conexão entre o visualizador e o CloudFront.	23 de maio de 2022

[Nova métrica FunctionThrottles para CloudFront Functions](#)

Com o Amazon CloudWatch, agora é possível monitorar o número de vezes que uma CloudFront Function foi limitada em determinado período.

4 de maio de 2022

[CloudFront oferece suporte a URLs de função do Lambda](#)

Se criar uma aplicação web sem servidor usando funções do Lambda com URLs da função, você poderá adicionar o CloudFront para obter diversos benefícios.

6 de abril de 2022

[Cabeçalho Server-Timing em respostas HTTP](#)

Agora, você pode habilitar o cabeçalho Server-Timing em respostas HTTP enviadas do CloudFront para visualizar métricas que podem ajudar a obter insights sobre o comportamento e a performance do CloudFront.

30 de março de 2022

[Usar lista de prefixos gerenciados pela AWS para limitar o tráfego de entrada](#)

Agora é possível limitar o tráfego HTTP e HTTPS de entrada às suas origens apenas dos endereços IP que pertencem aos servidores voltados para a origem do CloudFront.

7 de fevereiro de 2022

[Novo atributo](#)

O CloudFront adiciona suporte para políticas de cabeçalhos de resposta, que permitem especificar os cabeçalhos de HTTP que o CloudFront adiciona às respostas HTTP que ele envia aos visualizadores (navegadores da Web ou outros clientes). Você pode especificar os cabeçalhos desejados (e seus valores) sem fazer alterações na origem ou escrever nenhum código. Consulte mais informações em [Adicionar ou remover cabeçalhos HTTP em respostas do CloudFront](#).

2 de novembro de 2021

[Novo cabeçalho da solicitação CloudFront-Viewer-Address](#)

O CloudFront adiciona suporte a um novo cabeçalho , CloudFront-Viewer-Address , que contém o endereço IP do visualizador que enviou a solicitação de HTTP para o CloudFront. Consulte mais informações em [Adicionar cabeçalhos de solicitação do CloudFront](#).

25 de outubro de 2021

[Lambda@Edge compatível com nova versão de runtime](#)

O Lambda@Edge agora é compatível com funções Lambda com o tempo de execução Python 3.9. Consulte mais informações em [Tempos de execução compatíveis](#).

22 de setembro de 2021

[Atualização da política gerenciada da AWS](#)

CloudFront atualiza a política CloudFrontReadOnlyAccess. Consulte mais informações em [Atualizações do CloudFront em políticas gerenciadas pela AWS](#).

8 de setembro de 2021

[Novo atributo](#)

O CloudFront agora oferece suporte a certificados ECDSA para conexões HTTPS voltadas para o visualizador. Consulte mais informações em [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#) e em [Requisitos para usar certificados SSL/TLS com o CloudFront](#).

14 de julho de 2021

[Novo atributo](#)

O CloudFront agora oferece suporte a mais formas de mover um nome de domínio alternativo de uma distribuição para outra, sem que seja necessário entrar em contato com o AWS Support. Para ter mais informações, consulte [Move an alternate domain name to a different distribution](#).

7 de julho de 2021

[Nova política de segurança](#)

O CloudFront agora é compatível com uma nova política de segurança, TLSv1.2_2021, com um conjunto menor de criptogramas compatíveis. Consulte mais informações em [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

23 de junho de 2021

[Novo atributo](#)

O Amazon CloudFront agora oferece suporte ao CloudFront Functions, um recurso nativo do CloudFront que permite escrever funções leves em JavaScript para personalizações do CDN de alta escala e e sensíveis à latência. Consulte mais informações em [Como personalizar a borda com o CloudFront Functions](#).

3 de maio de 2021

[O Lambda@Edge é compatível com as versões de runtime mais recentes](#)

O Lambda@Edge agora oferece suporte para funções do Lambda com o tempo de execução do Node.js 14. Consulte mais informações em [Tempos de execução compatíveis](#).

29 de abril de 2021

[Remover documentação para distribuições RTMP](#)

[Em 31 de dezembro de 2020, o Amazon CloudFront tornou as distribuições RTMP obsoletas.](#) Agora a documentação para distribuições RTMP foi removida do Guia do desenvolvedor do Amazon CloudFront.

10 de fevereiro de 2021

[Nova opção de preço](#)

O Amazon CloudFront apresenta o pacote CloudFront Security Savings, uma maneira simples de economizar até 30% nas cobranças do CloudFront em sua fatura da AWS. Para ter mais informações, consulte as [Perguntas frequentes](#) do Pacote Savings.

sexta-feira, 5 de fevereiro de 2021

[Novo tutorial](#)

Agora o Guia do desenvolvedor do Amazon CloudFront inclui um tutorial para usar o Amazon CloudFront a fim de restringir o acesso a um Application Load Balancer no Elastic Load Balancing. Consulte mais informações em [Restringir o acesso aos Application Load Balancers.](#)

18 de dezembro de 2020

[Nova opção de gerenciamento de chaves públicas](#)

O CloudFront agora é compatível com o gerenciamento de chaves públicas para URLs assinados e cookies assinados por meio do console e da API do CloudFront, sem exigir acesso ao usuário-raiz da Conta da AWS. Consulte mais informações em [Especificar os assinantes que podem criar signed URLs e cookies](#).

22 de outubro de 2020

[Novo recurso: Origin Shield](#)

O CloudFront agora é compatível com o CloudFront Origin Shield, uma camada adicional na infraestrutura de armazenamento em cache do CloudFront que ajuda a minimizar a carga da origem, melhorar a disponibilidade e reduzir os custos operacionais. Consulte mais informações em [Usar o Amazon CloudFront Origin Shield](#).

20 de outubro de 2020

[Novo formato de compactação](#)

O CloudFront agora é compatível com a compactação Brotli quando você configura o CloudFront para compactar objetos nos pontos de presença do CloudFront. Também é possível configurar o CloudFront para armazenar objetos Brotli usando um cabeçalho Accept-Encoding normalizado. Consulte mais informações em [Fornecer arquivos compactados](#) e em [Suporte à compactação](#).

14 de setembro de 2020

[Novo protocolo TLS](#)

O CloudFront agora é compatível com o protocolo TLS 1.3 para conexões HTTPS entre visualizadores e distribuições do CloudFront. O TLS 1.3 está habilitado por padrão em todas as políticas de segurança do CloudFront. Consulte mais informações em [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

3 de setembro de 2020

[Novos logs em tempo real](#)

O CloudFront agora é compatível com logs configuráveis em tempo real. Com logs em tempo real, é possível obter informações sobre solicitações feitas para uma distribuição em tempo real. É possível usar os logs em tempo real para monitorar, analisar e tomar ações com base na performance da entrega de conteúdo. Consulte mais informações em [Logs em tempo real](#).

31 de agosto de 2020

[Suporte da API para métricas adicionais](#)

O CloudFront agora é compatível com a ativação de oito métricas adicionais em tempo real com a API do CloudFront. Consulte mais informações em [Ativar métricas adicionais](#).

28 de agosto de 2020

[Novos cabeçalhos HTTP do CloudFront](#)

O CloudFront adicionou mais cabeçalhos HTTP para determinar informações sobre o visualizador, como o tipo de dispositivo, a localização geográfica e muito mais. Consulte mais informações em [Adicionar cabeçalhos de solicitação do CloudFront](#).

23 de julho de 2020

[Novo atributo](#)

O CloudFront agora oferece suporte a políticas de cache e políticas de solicitação de origem, o que proporciona um controle mais granular sobre a chave de cache e as solicitações de origem para as suas distribuições do CloudFront. Para ter mais informações, consulte [Controlar a chave de cache](#) e [Controlar solicitações de origem](#).

22 de julho de 2020

[Nova política de segurança](#)

O CloudFront agora é compatível com uma nova política de segurança, TLSv1.2_2019, com um conjunto menor de criptografias compatíveis. Consulte mais informações em [Protocolos e cifras compatíveis entre visualizadores e o CloudFront](#).

8 de julho de 2020

[Novas configurações para controlar tempos limite e tentativas da origem](#)

O CloudFront adicionou novas configurações que controlam os tempos limite e as tentativas da origem. Consulte mais informações em [Controlar tempos limite e tentativas da origem](#).

5 de junho de 2020

[Nova documentação de conceitos básicos do CloudFront ao criar um site estático seguro](#)

Comece a usar o CloudFront criando um site estático seguro usando o Amazon S3, o CloudFront, o Lambda@Edge e muito mais, tudo implantado com o AWS CloudFormation. Para obter mais informações, consulte [Conceitos básicos de site estático seguro](#).

2 de junho de 2020

[O Lambda@Edge é compatível com as versões de runtime mais recentes](#)

O Lambda@Edge agora oferece suporte para funções do Lambda com os tempos de execução do Node.js 12 e do Python 3.8. Consulte mais informações em [Tempos de execução compatíveis](#).

27 de fevereiro de 2020

[Novas métricas em tempo real no CloudWatch](#)

Agora o Amazon CloudFront oferece oito métricas adicionais em tempo real no Amazon CloudWatch. Consulte mais informações em [Visualizar métricas adicionais de distribuição do CloudFront](#).

19 de dezembro de 2019

[Novos campos em logs de acesso](#)

O CloudFront adiciona sete novos campos aos logs de acesso. Consulte mais informações em [Campos padrão de arquivo de log](#).

12 de dezembro de 2019

[Plug-in do WordPress da AWS](#)

É possível usar o plug-in do WordPress da AWS para fornecer aos visitantes do site do WordPress uma experiência de visualização acelerada por meio do CloudFront. (Atualização: desde 30 de setembro de 2022, o plug-in do WordPress da AWS foi desativado.)

30 de outubro de 2019

[Políticas de permissão do IAM em nível de recurso e baseadas em tags](#)

O CloudFront agora é compatível com duas maneiras adicionais de especificar políticas de permissões do IAM: baseadas em tags e em nível de recurso. Para obter mais informações, consulte [Gerenciar o acesso aos recursos](#).

8 de agosto de 2019

[Suporte à linguagem de programação Python](#)

Agora, você pode usar a linguagem de programação Python, além do Node.js, para desenvolver funções no Lambda@Edge. Para obter exemplos de funções que abrangem várias situações, consulte [Funções de exemplo do Lambda@Edge](#).

1 de agosto de 2019

[Gráficos de monitoramento atualizados](#)

Atualizações de conteúdo que descrevem novas maneiras de você monitorar as funções Lambda associadas às suas distribuições do CloudFront diretamente no console do CloudFront, para monitoramento mais fácil e depuração de erros. Para obter mais informações, consulte [Monitorar o CloudFront](#).

20 de junho de 2019

[Conteúdo de segurança consolidado](#)

Um novo capítulo de segurança consolida as informações sobre os recursos do CloudFront e a implementação de proteção de dados, IAM, registro em log, conformidade e muito mais. Para obter mais informações, consulte [Segurança](#).

24 de maio de 2019

[Agora é necessária a validação do domínio](#)

O CloudFront agora requer que você use um certificado do SSL para verificar se tem permissão para usar um nome de domínio alternativo com uma distribuição. Para obter mais informações, consulte [Usar nomes de domínio alternativos e HTTPS](#).

9 de abril de 2019

[Atualização do nome do arquivo PDF](#)

O novo nome de arquivo do Guia do desenvolvedor do Amazon CloudFront é: AmazonCloudFront_DeveloperGuide. O nome anterior era: cf-dg.

7 de janeiro de 2019

[Novos atributos](#)

O CloudFront agora é compatível com o uso do WebSocket, um protocolo baseado em TCP que é útil quando você precisa de conexões de longa duração entre clientes e servidores. Você também pode configurar o CloudFront com o failover de origem para cenários que exigem alta disponibilidade. Para obter mais informações, consulte [Usar o WebSocket com distribuições do CloudFront](#) e [Otimizar alta disponibilidade com o failover da origem do CloudFront](#).

20 de novembro de 2018

Novo atributo

O CloudFront agora é compatível com o registro detalhado de erros em log para solicitações HTTP que executam as funções do Lambda. Você pode armazenar os logs no CloudWatch e usá-los para ajudar a solucionar erros HTTP 5xx quando sua função retornar uma resposta inválida. Para obter mais informações, consulte [Métricas do CloudWatch e o CloudWatch Logs para funções do Lambda](#).

8 de outubro de 2018

Novo atributo

Agora, você pode optar por fazer com que o Lambda@Edge expor o corpo em uma solicitação para métodos HTTP graváveis (POST, PUT, DELETE e assim por diante), para que você possa acessá-lo na sua função do Lambda. Você pode escolher acesso somente leitura ou especificar que substituirá o corpo. Para obter mais informações, consulte [Acessar o corpo da solicitação escolhendo a opção para incluir corpo](#).

14 de agosto de 2018

Novo atributo

O CloudFront agora pode fornecer conteúdo comprimido usando brotli ou outros algoritmos de compactação, além de gzip ou em vez dele. Para obter mais informações, consulte [Fornecer arquivos compactados](#).

25 de julho de 2018

Reorganização

O Guia do desenvolvedor do Amazon CloudFront foi reorganizado para simplificar a localização de conteúdo relacionado e melhorar a capacidade de verificação e de navegação.

28 de junho de 2018

Novo recurso

O Lambda@Edge agora permite personalizar ainda mais a entrega de conteúdo armazenado em um bucket do Amazon S3, permitindo que você acesse cabeçalhos adicionais, inclusive cabeçalhos personalizados, em eventos voltados para a origem. Para obter mais informações, consulte estes exemplos que mostram a personalização de conteúdo com base na [localização do visualizador](#) e no [tipo de dispositivo do visualizador](#).

20 de março de 2018

[Novo recurso](#)

Agora é possível usar o Amazon CloudFront para negociar conexões HTTPS com as origens usando o Elliptic Curve Digital Signature Algorithm (ECDSA, Algoritmo de assinatura digital de curvas elípticas). O ECDSA usa chaves menores que são mais rápidas, mas tão seguras quanto o algoritmo RSA mais antigo. Para obter mais informações, consulte [Protocolos e criptografias SSL/TLS compatíveis para comunicação entre o CloudFront e a origem](#) e [Sobre as criptografias RSA e ECDSA](#).

15 de março de 2018

[Novo recurso](#)

O Lambda@Edge permite personalizar as mensagens de erro da origem, possibilitando a execução de funções do Lambda em resposta a erros HTTP que o Amazon CloudFront recebe da origem. Para obter mais informações, consulte estes exemplos que mostram [redirecionamentos para outro local](#) e a [geração de respostas com o código de status 200 \(OK\)](#).

21 de dezembro de 2017

[Novo recurso](#)

Um novo recurso do CloudFront, a criptografia em nível de arquivo, ajuda a melhorar ainda mais a segurança dos dados confidenciais, como números de cartões de crédito ou informações de identificação pessoal (PII), como números de CPF. Consulte mais informações em [Usar a criptografia no nível de campo para ajudar a proteger dados sigilosos.](#)

14 de dezembro de 2017

[Histórico de documentos arquivados](#)

O histórico de documentos mais antigos foi arquivado.

1.º de dezembro de 2017