(19)日本国特許庁(JP)　(12)公表特許公報(A)　(11)特許出願公表番号
特表2010-508594
(P2010-508594A)
(43)公表日　平成22年3月18日(2010.3.18)

(51)Int.Cl.		F I			テーマコード（参考）
G06F 21/22	(2006.01)	G06F	9/06	660G	5B276
G06F 21/20	(2006.01)	G06F	15/00	330A	5B285
H04L 9/32	(2006.01)	H04L	9/00	675B	5J104
G09C 1/00	(2006.01)	G09C	1/00	660D	

審査請求 未請求　予備審査請求 未請求　（全 28 頁）

(21)出願番号　特願2009-534953 (P2009-534953)
(86)(22)出願日　平成19年11月1日 (2007.11.1)
(85)翻訳文提出日　平成21年4月1日 (2009.4.1)
(86)国際出願番号　PCT/US2007/083390
(87)国際公開番号　WO2008/057970
(87)国際公開日　平成20年5月15日 (2008.5.15)
(31)優先権主張番号　11/555,610
(32)優先日　平成18年11月1日 (2006.11.1)
(33)優先権主張国　米国 (US)

(71)出願人　500046438
マイクロソフト　コーポレーション
アメリカ合衆国　ワシントン州　９８０５２－６３９９　レッドモンド　ワン　マイクロソフト　ウェイ
(74)代理人　100077481
弁理士　谷　義一
(74)代理人　100088915
弁理士　阿部　和夫
(72)発明者　ブレア　ビー．ディラウェイ
アメリカ合衆国　９８０５２　ワシントン州　レッドモンド　ワン　マイクロソフト　ウェイ　マイクロソフト　コーポレーション　インターナショナル　パテンツ内

Ｆターム(参考)　5B276 FB02

最終頁に続く

(54)【発明の名称】分散されたアプリケーション情報配信のセキュリティ保護

(57)【要約】
例示的な実施形態では、データ構造は、安全なアプリケーション命令プロトコルに適合する。データ構造は、第１のアプリケーションレベル要求および第２のアプリケーションレベル要求を含む。第１のアプリケーションレベル要求は、リクエスタからのアプリケーション固有の命令、およびリクエスタからのそのアプリケーション固有の命令に対するリクエスタ署名を有する。第２のアプリケーションレベル要求は、中間体からのアプリケーション固有の命令、および中間体からの少なくともアプリケーション固有の命令に対する中間体署名を有する。

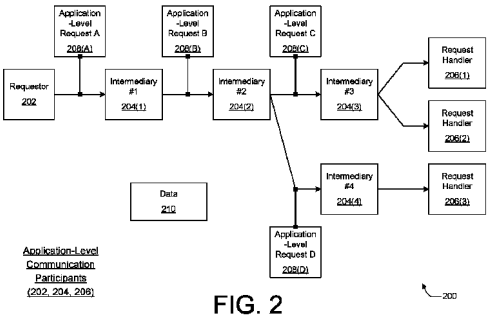


FIG. 2

【特許請求の範囲】
【請求項１】
安全なアプリケーション命令プロトコルのためのデータ構造（２０８）を備えるプロセッサ実行可能命令（７１０）を含む１つまたは複数のプロセッサでアクセス可能な媒体（７０８）であって、前記データ構造は、
リクエスタからのアプリケーション固有の命令（３０２）、および前記リクエスタからの前記アプリケーション固有の命令に対するリクエスタ署名（３０４）を含む第１のアプリケーションレベル要求（２０８（Ａ））と、
中間体からのアプリケーション固有の命令（３０６）、および前記中間体からの少なくとも前記アプリケーション固有の命令に対する中間体署名（３０８）を含む第２のアプリケーションレベル要求（２０８（Ｂ））と
を含むことを特徴とする１つまたは複数のプロセッサでアクセス可能な媒体。
【請求項２】
前記第１のアプリケーションレベル要求が、前記第２のアプリケーションレベル要求内にネストされることを特徴とする請求項１に記載の１つまたは複数のプロセッサでアクセス可能な媒体。
【請求項３】
前記第１のアプリケーションレベル要求はさらに、データにアクセスするための、前記リクエスタからの権利の委任を含むことを特徴とする請求項１に記載の１つまたは複数のプロセッサでアクセス可能な媒体。
【請求項４】
前記第２のアプリケーションレベルは要求さらに、データにアクセスするための、前記中間体からの権利の委任を含むことを特徴とする請求項１に記載の１つまたは複数のプロセッサでアクセス可能な媒体。
【請求項５】
前記中間体署名がまた、前記中間体からの権利の前記委任に対して行われることを特徴とする請求項４に記載の１つまたは複数のプロセッサでアクセス可能な媒体。
【請求項６】
前記中間体署名がまた、前記第１のアプリケーションレベル要求に対して行われることを特徴とする請求項１に記載の１つまたは複数のプロセッサでアクセス可能な媒体。
【請求項７】
前記第１のアプリケーションレベル要求はさらに、前記リクエスタからの、データアクセスのための権利の委任を含み、また前記第２のアプリケーションレベル要求はさらに、前記中間体からの、データアクセスのための権利の委任を含み、
前記第１のアプリケーションレベル要求が、前記第２のアプリケーションレベル要求内にネストされており、
前記データ構造はさらに、
他の中間体からのアプリケーション固有の命令、前記他の中間体からの少なくとも前記アプリケーション固有の命令に対する他の中間体署名、および前記他の中間体からのデータアクセスのための権利の委任を含む第３のアプリケーションレベル要求を含み、また前記第１のアプリケーションレベル要求および前記第２のアプリケーションレベル要求が、前記第３のアプリケーションレベル要求内にネストされることを特徴とする請求項１に記載の１つまたは複数のプロセッサでアクセス可能な媒体。
【請求項８】
データアクセスのための権利の前記委任のそれぞれは、委任情報を含み、また前記情報が暗号化されることを特徴とする請求項７に記載の１つまたは複数のプロセッサでアクセス可能な媒体。
【請求項９】
安全なアプリケーション命令プロトコルを実施するアプリケーション（５０２）を含む装置（７０２）であって、前記アプリケーションは、

着信するアプリケーションレベル要求（２０８）をカプセル化するための、また後続するエンティティに対するアプリケーション固有の命令（３０６）を追加するためのメッセージ増補器（５１４）と、
前記後続するエンティティに対する少なくとも前記アプリケーション固有の命令にデジタル的に署名するための、また送出される中間体署名（３０８）を追加するためのメッセージ署名器（５１６）と
を備えることを特徴とする装置。
【請求項１０】
前記着信するアプリケーションレベル要求は、リクエスタからのアプリケーション固有の命令、および前記リクエスからの前記アプリケーション固有の命令に対するリクエスタ署名を含むことを特徴とする請求項９に記載の装置。
【請求項１１】
前記着信するアプリケーションレベル要求はさらに、中間体からのアプリケーション固有の命令、および前記中間体からの前記アプリケーション固有の命令に対する中間体署名を含むことを特徴とする請求項１０に記載の装置。
【請求項１２】
前記着信するアプリケーションレベル要求から、前記リクエスタからの前記アプリケーション固有の命令、および前記中間体からの前記アプリケーション固有の命令を抽出するためのアプリケーション固有の情報抽出器と、
前記リクエスタからの前記アプリケーション固有の命令、および前記中間体からの前記アプリケーション固有の命令を分析して、前記後続するエンティティに対する識別を決定し、また前記後続するエンティティに対するアプリケーション固有の命令を決定するためのアプリケーション固有の情報分析器と
をさらに備えることを特徴とする請求項１１に記載の装置。
【請求項１３】
前記リクエスタ署名を用いて、前記リクエスタからの前記アプリケーション固有の命令が、本当に前記リクエスタから生成されたこと、および前記中間体署名を用いて、前記中間体からの前記アプリケーション固有の命令が、本当に前記中間体から生成されたことを認証するためのメッセージ参加者認証器をさらに備える特徴とする請求項１１に記載の装置。
【請求項１４】
前記リクエスタ署名を用いて、前記リクエスタからの前記アプリケーション固有の命令が、前記リクエスタにより署名された後に変更されていないこと、および前記中間体署名を用いて、前記中間体からの前記アプリケーション固有の命令が、前記中間体により署名された後に変更されていないことを検証するための、メッセージ情報の完全性検証器をさらに備えることを特徴とする請求項１１に記載の装置。
【請求項１５】
前記メッセージ増補器はさらに、前記後続するエンティティが、データにアクセスできるように、または前記データにアクセスするために権利をさらに委任できるように、権利の委任情報を追加することを特徴とする請求項９に記載の装置。
【請求項１６】
前記メッセージ署名器は、前記着信するアプリケーションレベル要求および前記追加された権利の委任情報にデジタル的に署名して、前記送出される中間体署名を有する送出アプリケーションレベル要求を作成し、また前記装置は、前記送出アプリケーションレベル要求を前記後続するエンティティに向けて転送することを特徴とする請求項１５に記載の装置。
【請求項１７】
アプリケーションレベルで、リクエスタによりデジタル的に署名された要求を有する着信メッセージを受信するステップ（６０２）と、
後続するエンティティに対するアプリケーション固有の命令を追加することにより、前

記要求を増補するステップ（６１２）と、
前記アプリケーション固有の命令にデジタル的に署名して（６１４）、前記要求、前記アプリケーション固有の命令、および前記アプリケーション固有の命令に対するデジタル署名を含む送出メッセージを作成するステップと、
前記後続するエンティティに向けて、前記送出メッセージを送信するステップ（６１６）と
を含むことを特徴とする方法。
【請求項１８】
前記増補するステップは、
前記後続するエンティティに対する権利の委任情報を追加することにより前記要求を増補するステップを含み、権利の前記委任情報は、前記後続するエンティティが、前記要求と関連するデータにアクセスすることを可能にし、または前記データへの権利をさらに委任することを可能にすることを特徴とする請求項１７に記載の方法。
【請求項１９】
前記要求から、前の中間体により生成されたアプリケーション固有の中間体命令を抽出するステップと、
前記要求から、前記リクエスタにより生成されたアプリケーション固有のリクエスタ命令を抽出するステップと、
前記アプリケーション固有の中間体命令、および前記アプリケーション固有のリクエスタ命令を分析して、前記後続するエンティティに対する識別を決定するステップと
をさらに含むことを特徴とする請求項１７に記載の方法。
【請求項２０】
前記要求から、前記リクエスタにより生成された前記アプリケーション固有のリクエスタ命令、および前記リクエスタ署名を抽出するステップと、
前記リクエスタ署名を用いて、前記アプリケーション固有のリクエスタ命令が、前記リクエスタにより生成されたことを認証するステップと、
前記リクエスタ署名を用いて、前記アプリケーション固有のリクエスタ命令の完全性を検証するステップと
をさらに含むこと特徴とする請求項１７に記載の方法。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、分散されたアプリケーション情報配信のセキュリティ保護に関する。
【背景技術】
【０００２】
最近の分散コンピューティングシステムでは、１群のユーザによる１つまたは複数の共用された計算資源の使用を最適化することがますます重要になってきている。この現象の一例が、グリッドコンピューティングシステムである。典型的なグリッドコンピューティング環境内では、いくつかの計算装置へのアクセスは、１組のジョブ管理システムにより制御される。ジョブ管理システムは、サブミットされたジョブに対する計算資源の割振りを決定し、これらのジョブのスケジューリングを決定するが、課金アカウント、セキュリティ資格証明書、並列に実行されるジョブ活動のロケーションなど、ジョブの実行状況の諸側面を決定することも時々あり得る。ジョブ管理システム（複数可）の目的は、グリッド環境のユーザからジョブ要求を受け取ること、および計算資源の全体の使用を最適化することである。コンピュータ資源は、スーパーコンピュータ、コンピューティングクラスタ、アプリケーションサーバ、デスクトップワークステーションなどを含むことができる。
【０００３】
グリッドシステムがその一例である分散コンピューティングシステムは、多数のユーザおよびコンピュータをサポートする、アプリケーションおよび資源管理システムの階層を

含むことができる。例えば、ユーザは、集中化したジョブマネジャに、アプリケーションを動作させるように依頼することができる。中央マネジャは、次に、計算クラスタの集合体を担当する補助的なジョブマネジャにアプリケーションを動作させるように依頼することができる。補助的なマネジャは、そのアプリケーションに最も適切な特定のコンピューティング資源を決定し、次いで、その計算クラスタのジョブマネジャにユーザのアプリケーションを動作させるように要求する。

【先行技術文献】

【非特許文献】

【０００４】

【非特許文献１】http://www.w3.org/2001/04/xmlenc#aes128-cbc

【非特許文献２】http://www.w3.org/2001/04/xmlenc#rsa-oaep-mgflp

【非特許文献３】http://www.w3.org/2001/04/xmlenc#kw-aes128

【発明の概要】

【発明が解決しようとする課題】

【０００５】

このような階層的に管理された分散システムでは、ユーザのアプリケーション、および任意の補助的なマネジャを含む割り当てられた計算クラスタを担当する一連のジョブマネジャは、分散環境の全体の状態に基づいて動的に決定することができる。ジョブ要求がサブミットされた時点で、ユーザは、そのアプリケーションが最終的にどこで実行されるかの細部を知らない可能性があるので、アプリケーションの実行時に必要ないくつかの情報は、ジョブ要求を処理する１つまたは複数のジョブマネジャにより提供されなくてはならない、または提供されるのが適切である可能性が高い。既存のシステムは、このような情報のセキュリティに対して効率的で十分な保護を提供することに失敗している。

【課題を解決するための手段】

【０００６】

例示的な実施形態では、データ構造は、安全なアプリケーション命令プロトコルに適合する。データ構造は、第１のアプリケーションレベル要求および第２のアプリケーションレベル要求を含む。第１のアプリケーションレベル要求は、リクエスタからのアプリケーション固有の命令、およびそのリクエスタからのアプリケーション固有の命令に対するリクエスタ署名を有する。第２のアプリケーションレベル要求は、中間体からのアプリケーション固有の命令、およびその中間体からの少なくともアプリケーション固有の命令に対する中間体署名を有する。

【０００７】

この要約は、以下の詳細な説明でさらに説明する諸概念の選択を、簡略化した形で導入するために提供されている。この要約は、特許請求される保護対象の重要な特徴または本質的な特徴を特定するようには意図されておらず、特許請求される保護対象の範囲を決定する一助として使用されることも意図されていない。さらに、他の方法、システム、スキーム、装置、デバイス、媒体、手順、ＡＰＩ、構成などの実施形態が本明細書で説明される。

【０００８】

同様のおよび／または対応する態様、特徴、およびコンポーネントを参照するために、諸図面を通して同じ番号が使用される。

【図面の簡単な説明】

【０００９】

【図１】分散されたアプリケーション情報配信のセキュリティ保護を実施できる例示的な分散コンピューティング環境のブロック図である。

【図２】送信されるアプリケーションレベル要求に対して、分散されたアプリケーション情報配信のセキュリティ保護が実施され得る一般的な例のコンピューティング環境を示すブロック図である。

【図３Ａ】図２の例示的なコンピューティング環境で示されるものなど、例示的なアプリ

ケーションレベル要求を示すブロック図である。
【図３Ｂ】図２の例示的なコンピューティング環境で示されるものなど、例示的なアプリケーションレベル要求を示すブロック図である。
【図３Ｃ】図２の例示的なコンピューティング環境で示されるものなど、例示的なアプリケーションレベル要求を示すブロック図である。
【図４】データにアクセスするための権利の委任を含む例示的なアプリケーションレベル要求を示すブロック図である。
【図５】図２の例示的なコンピューティング環境で示されるものなど、アプリケーションレベル通信の参加者上で実行することができる例示的なアプリケーションを示すブロック図である。
【図６】アプリケーションレベルで要求情報を安全に通信するための方法の例を示す流れ図である。
【図７】分散されたアプリケーション情報配信のセキュリティ保護を実施するために使用され得る例示的な装置のブロック図である。
【発明を実施するための形態】
【００１０】
分散されたアプリケーション情報配信のセキュリティ保護への導入
本明細書で前述したように、いくつかの分散コンピューティングシステムは、多数のユーザおよびコンピューティング装置をサポートできるアプリケーションおよび資源管理システムの階層を含む。例のために過ぎないが、ユーザは、集中化したマネジャに、アプリケーションを動作させるように依頼することができる。中央マネジャは、次に、コンピューティング資源クラスタの集合体を担当する補助的なマネジャにアプリケーションを動作させるように依頼することができる。補助的なマネジャは、そのアプリケーションに最も適切な特定のコンピューティング資源を決定し、次いで、その特定のコンピューティング資源のマネジャに、そのアプリケーションを動作させるように要求する。
【００１１】
このようなシステムでは、アプリケーションを動作させる必要のあるユーザと、アプリケーションを実際に動作させるコンピューティング資源との間に、通常、複数のマネジャタイプの中間体がある。前述のジョブマネジャは、このような環境中に存在するマネジャの１つのタイプである。存在することができ、ユーザのジョブ要求の処理を助けることのできる他のタイプの処理マネジャは、これだけに限らないが、メッセージ経路指定マネジャ、監査マネジャなどを含む。これらのマネジャは、総称的に、要求を処理する中間体と呼ぶことができる。要求を処理する中間体は、スケジュールされている途中の、または実行途中の他のアプリケーションに依存しているため、概して、アプリオリに決定することができない。これらの他のアプリケーションは、通常、いずれの単一の要求ユーザにも知られていない。
【００１２】
既存の手法を用いる場合、アプリケーションの実行サイトで、複数のエンティティにより行われてきた可能性のある、アプリケーション情報（例えば、アプリケーション固有の命令）を認証し、および／またはその完全性を保証することは困難である。不適正な命令を使用することは、誤った課金、データのセキュリティ違反、誤った計算、サービスの拒否などの結果となるおそれがあるため、このようなアプリケーション情報のセキュリティは非常に重要になり得る。
【００１３】
既存の手法は、概して、要求を行うエンティティが、要求の完全性を保証し、その要求を最終的に処理することになるエンティティにセキュリティで保護されたメッセージを送ることによって認証を実施できることを想定している。これらの既存の手法は、元のリクエスタ、および各中間のプロセッサが、最終的に誰がその要求を処理することになるかを知ることができないため、前述の分散コンピューティング環境中に存在する問題を完全には対処していない。したがって、それらは、従来の手法を用いてその要求を最終的に処理

することになるエンティティに対して、セキュリティで保護されたメッセージを適正に形成することができない。それに代えて、その要求は、１組の独立したメッセージを用いて処理され通信されるが、メッセージのセキュリティは、ポイントツーポイントベースでコンテンツを保護するに過ぎない。

【００１４】

図１は、分散されたアプリケーション情報配信のセキュリティ保護が実施され得る、例示的な分散コンピューティング環境１００のブロック図である。図示のように、分散コンピューティング環境１００は、「ｕ」個のユーザ１０２、「ｐ」個の処理マネジャ１０４、「ｃ」個のコンピューティング資源１０６、要求１０８、およびデータ１１０を含む。分散コンピューティング環境１００は、ジョブ活動が、少なくとも１つのコンピューティング資源１０６上でデータ１１０を用いて実施され得るように、ユーザ１０２からの要求１０８が、１つまたは複数の処理マネジャ１０４を介して通信される環境を表している。アプリケーションレベル通信の参加者または分散コンピューティング環境１００のエンティティは、ユーザ１０２、処理マネジャ１０４、およびコンピューティング資源１０６を含む。

【００１５】

より具体的には、分散コンピューティング環境１００は、ユーザ１０２（１）、ユーザ１０２（２）、ユーザ１０２（３）・・・ユーザ１０２（ｕ）を含み、「ｕ」は何らかの正の整数である。それはまた、処理マネジャ１０４（１）、処理マネジャ１０４（２）、処理マネジャ１０４（３）、処理マネジャ１０４（４）・・・処理マネジャ１０４（ｐ）を含み、「ｐ」は何らかの正の整数である。さらに、分散コンピューティング環境１００は、コンピューティング資源１０６（１）、コンピューティング資源１０６（２）、コンピューティング資源１０６（３）・・・コンピューティング資源１０６（ｃ）を含み、「ｃ」は何らかの正の整数である。

【００１６】

前述の実施形態では、各ユーザ１０２は、何らかのタスクがデータ１１０に対して実施されるように依頼するタスク要求１０８を送信することができる。１つのデータエレメント１１０だけが図示されているが、各ユーザ１０２は、それ自体の各データ、共用データ、複数単位のデータなどと関連付けることができる。処理マネジャ１０４の階層に関しては、要求されたタスクを実施するようにコンピューティング資源１０６に直接依頼できるまで、要求１０８は、１つの処理マネジャ１０４から他の処理マネジャ１０４へと転送される。各コンピューティング資源１０６は、単一のコンピューティング装置、コンピューティング装置のクラスタ、コンピューティング装置のクラスタの一部などとすることができる。

【００１７】

例示的なタスク要求が、分散コンピューティング環境１００の一部として示されている。この例示的なタスク要求では、ユーザ１０２（２）は、データ１１０に対して動作される、または実行される必要のあるアプリケーションを有する。ユーザ１０２（２）は、１つの参加エンティティから次のエンティティへと伝達されるメッセージとして実現される要求１０８を作成する。要求１０８は、ユーザ１０２（２）から処理マネジャ１０４（１）に送信される。処理マネジャ１０４（１）は、要求１０８を処理マネジャ１０４（２）に転送する。処理マネジャ１０４（２）は、要求１０８を処理マネジャ１０４（３）に転送する。処理マネジャ１０４（３）は、要求１０８をコンピューティング資源１０６（２）に最終的に転送し、それが、要求されたタスクを実際に実施する。要求１０８は、その転送に先立って各処理マネジャ１０４により修正され得る。

【００１８】

従来の見方からすると、図１に示すコンピュータシステム環境のタイプは、２つの潜在的なセキュリティ問題を生ずる。第１に、各中間のマネジャ（例えば、処理マネジャ１０４）および最終的な計算装置（複数可）（例えば、コンピューティング資源１０６）は、仕事をそのためにすることになる特定のユーザ（例えば、ユーザ１０２）および中間マネ

ジャを規定するアクセス制御ポリシーを有する可能性が高い。ユーザ、マネジャ、および計算装置が、既存の大規模なコンピューティンググリッドおよび複数の会社を含む協働的ビジネスシステムにおいて典型的な制限されたクロスドメインの信頼関係を有する別々の管理ドメイン中に存在する場合、特にそれは正しい。したがって、中間体、または最終の計算装置に対する各要求（例えば、要求１０８）は、要求するユーザにより、またアプリケーション要求を処理した前の各中間体により提供される命令に関する認証情報を提供することが可能でなくてはならない。

【００１９】

第２に、アプリケーションを動作させる計算装置（複数可）は、通常、ユーザにより特定されたデータ資源（例えば、データ１１０）へのアクセスを必要とする。このデータを保持するリポジトリは、そのデータに対して操作することのできる者を制限するアクセス制御ポリシーを有する可能性が高い。したがって、ユーザおよび／または処理する中間体は、データリポジトリが、データへのアクセスの許可を有効であるとして受け入れるためのセキュリティ資格証明書を有する実行アプリケーション（例えば、コンピューティング資源１０６で）を提供する何らかの機構を必要とする。

【００２０】

これらの問題に対する既存の手法は、どのようにすれば、一連の動的に決定される中間のプロセッサにより徐々に増えて生成される１組のアプリケーション命令を、アプリケーション命令の各組の完全性を検証し、また各中間のプロセッサの認証を実施できるように通信できるかについて対処することに失敗しているので不十分である。したがって、このようなコンピューティングシステム全体のセキュリティがしばしば低下することになり、それはまた、有用性に悪影響を与えるおそれもある。

【００２１】

第１のセキュリティ問題に関しては、既存のプロトコルは、参加エンティティの増分的な発見を伴うこの多段処理タイプを扱うように設計されていない。多くのプロトコルは、ポイントツーポイント用途（例えば、ＩＰｓｅｃ、ＳＳＬ、ＤＣＥ／ＲＰＣなど）のために設計されている。これらのポイントツーポイントプロトコルは、２つの知られたエンドポイント間で（すなわち、宛先エンドポイントは、メッセージの送信に先立って知られている必要がある）セキュリティで保護されたメッセージを送ることを可能にするが、２つのエンドポイント間のいずれの中間体も、不透明な２進データを見るだけである。前述した通信参加者間のメッセージフローをセキュリティで保護するために、このようなポイントツーポイントのプロトコルを使用することができるが、残念ながら、第１の参加者における着信要求に対するセキュリティと、次の参加者に向けた送出される要求に対するセキュリティとの間で定義された関係は何もない。

【００２２】

いくつかの他のプロトコル（例えば、ＳＯＡＰ　Ｍｅｓｓａｇｅ　Ｓｅｃｕｒｉｔｙ）は、知られたエンドポイントへのメッセージのセキュリティを処理するように設計されているが、中間のプロセッサに対するセキュリティは、別々に対処する。しかし、これらのプロトコルは、なお、そのエンドポイントが、アプリオリに知られていること、および経路指定の挙動により、動的に発見され得るのは中間体だけであることを仮定している。その結果、既存のシステムは、知られたエンドポイントで、メッセージの認証および完全性の保護が使用されるように意図されているポイントツーポイントのメッセージセキュリティを使用する傾向がある。

【００２３】

既存のプロトコルを用いた安全なメッセージに依存することは、処理する中間体および計算装置のすべてが、同じ管理ドメイン中にある場合、またはその他の形で完全に互いに信頼する場合に、前述の分散コンピューティング環境に対して適切に実施することが可能になる。このような場合、このようなプロトコルが、推移的な信頼モデルの存在を仮定する（すなわち、そのプロトコルは、要求の受信者が送信者を信頼し、また含意（implication）によりその要求を送信者に送る者は誰でも信頼する、信頼モデルの存在を仮定する

）ことは許容できる。しかし、完全に信頼することを正当化できない場合、この仮定は、敵意のある中間体が、実行すべきアプリケーションに影響を与える成功裡の間接的攻撃を開始させるおそれがある。さらに、完全に信頼するシナリオは、要求送信者の識別と、要求送信者がその要求中に符号化した情報に基づいて、粗粒度のアクセス制御が可能になるだけである。すなわち、いずれのアクセス制御も、最終的には、セキュリティで保護された要求メッセージを送ったエンティティの信頼に基づくだけである。

【００２４】

第２のセキュリティ問題に関しては、権利の委任を可能にする既存の手法は、同様な限界を有する。例えば、Ｃｏｎｄｏｒなどのいくつかのグリッドジョブ管理システムは、マッチメーキング（matchmaking）モードで動作する。Ｃｏｎｄｏｒマネジャ（複数可）は、アプリケーションを動作させるための資源（複数可）を突き止め、これらの資源を予約し、次いで、要求発信者に、どの資源を使用してよいかを通知する。この予約手法は、リクエスタが、データ委任のセキュリティ資格証明書を、要求発信者のアプリケーションを動作させる実際の計算装置に渡す必要があるだけなので、委任の観点から許容することができる。

【００２５】

しかし、この予約手法は、いくつかの負の側面を有する。第１に、要求発信者は、処理マネジャが適切な計算装置を発見するのにどれくらい時間がかかるかを知らないため、利用可能な状態に留まっている必要がある。おそらく、計算装置は、実際のアプリケーション、ならびにその関連する命令および委任の資格証明書が、合理的な時間量中に提供されない場合、資源の予約を取り消すことになる。第２に、リクエスタ発信者は、その計算装置に対して直接的なアクセスを必要とするが、これは、いくつかの複雑なシステムでは、ネットワークトポロジ、ファイアウォールなどにより実際的ではなく、可能でさえないこともあり得る。第３に、マネジャは、おそらく、どのような委任が必要であるかに気付くことがなく、したがって、適切な計算装置を選択するために、委任情報を使用することができない。

【００２６】

いくつかの他の実施形態では、（例えば、ＭｙＰｒｏｘｙサーバを介して）要求発信者を代理する（proxy）ために使用され得る名前－パスワード資格証明書が、セキュリティ資格証明書へのアクセスを許可するために使用される。これらの名前－パスワード資格証明書は、通常、データとして中間マネジャに渡され、最終的に計算装置に渡される。それらは、暗号化されたメッセージで搬送することができるが、各中間マネジャおよび各計算装置では、平文データとして示される。その資格証明書にアクセスでき、次いでそれらを送った中間マネジャの完全な組を追跡することを可能にするプロトコルサポートは存在しない。このような追跡情報は、例えば、誰がセキュリティ資格証明書または他の情報にアクセスできる可能性があったかを監査するために、また何らかの予測しないアクセスが生じた場合に法的な調査を行うために重要なものとなり得る。要するに、既存の手法を用いると、要求発信者とアプリケーションを実際に動作させる最終的なコンピューティング資源との間で、アプリケーション固有の情報のセキュリティを保護するための適切な機構を提供することが困難である。

【００２７】

対照的に、本明細書で説明するいくつかの実施形態を用いると、通信エンティティの参加者１０２、１０４、および／または１０６（図１による）に沿った送信経路は、安全なアプリケーション命令プロトコルを用いてセキュリティで保護され、および／または追跡可能にすることができる。例えば、各処理マネジャ１０４は、要求１０８の転送に先立ってアプリケーション固有の命令を追加することによって、要求１０８を増補する（augment）ように選ぶことができる。いくつかの分散コンピューティングネットワークでは、これらの増補的命令を追加することは、進んで実行しようとするすべての情報活動に対して、コンピューティング資源１０６により必要となり得る。各処理マネジャ１０４は、他の処理マネジャ１０４への転送に先立って要求１０８にデジタル的に署名することができる

。要求の増補は、アプリケーションレベルで実施される。デジタル署名は、アプリケーション固有の情報に適用される。したがって、アプリケーションレベル通信の参加者１０２、１０４、および／または１０６は、特有のアプリケーション命令を提供するエンティティを認証し、またこのアプリケーション情報の完全性を検証することができる。さらに、参加するエンティティの識別を追跡することができる。

【００２８】

より具体的には、前述の実施形態に関して、１組のアプリケーション固有の命令の送信者は、次の処理中間体が解読できる形で暗号化されたセキュリティ資格証明書情報を含めることができる。この処理中間体は、次いで、（ｉ）他の処理中間体が解読できる形で、その資格証明書を再度符号化し、また（ｉｉ）中間体が作成した中間体供給のアプリケーション固有の命令の組の中にこの再度暗号化した資格証明書を含めることができる。このプロセスは、コンピューティング資源などの最終的な要求ハンドラに達するまで、ポイントツーポイントベースで継続される。資格証明書は要求ハンドラで使用される。前段落で説明したように、アプリケーション固有の命令の各組がデジタル的に署名される場合、平文の資格証明書にアクセスした各エンティティに関する検証可能な記録が提供される。

【００２９】

分散されたアプリケーション情報配信のセキュリティ保護に関する例示的な実施形態

図２は、分散されたアプリケーション情報配信のセキュリティ保護が、アプリケーションレベル要求２０８に対して実施され得る一般的な例のコンピューティング環境２００を示すブロック図である。図示のように、コンピューティング環境２００は、リクエスタ２０２、複数の（処理する）中間体２０４、複数の要求ハンドラ２０６、アプリケーションレベル（タスク）要求２０８、およびデータ２１０を含む。コンピューティング環境２００では、アプリケーションレベル通信の参加者またはエンティティは、リクエスタ２０２、中間体２０４、および要求ハンドラ２０６を含む。

【００３０】

より具体的には、４つの処理中間体２０４（１）、２０４（２）、２０４（３）、および２０４（４）が示されている。３つの要求ハンドラ２０６（１）、２０６（２）、および２０６（３）が示されている。アプリケーションレベル要求２０８の４つのバージョンが示されている、すなわち、アプリケーションレベル要求Ａ　２０８（Ａ）、アプリケーションレベル要求Ｂ　２０８（Ｂ）、アプリケーションレベル要求Ｃ　２０８（Ｃ）、およびアプリケーションレベル要求Ｄ　２０８（Ｄ）である。アプリケーションレベル通信参加者の図示された各タイプの特定の数が、コンピューティング環境２００中に示されているが、リクエスタ２０２、中間体２０４、および／または要求ハンドラ２０６のそれぞれの任意の数を、所与のアプリケーションレベル要求通信に含むことができる。

【００３１】

前述の実施形態では、概して、リクエスタ２０２は、データ２１０と関連するアプリケーションレベル要求２０８を生成し、開始する。アプリケーションレベル要求２０８は、１つまたは複数の処理中間体２０４の間で、またその中で送信される。各処理中間体２０４は、アプリケーションレベル要求２０８を、次にどこに転送すべきかを決定する。本明細書の以下でさらに説明するように、各中間体２０４はまた、アプリケーションレベル要求２０８を、アプリケーション固有の命令をそれに追加することによって増補することができる。最終的に、中間体２０４は、アプリケーションレベル要求２０８を少なくとも１つの要求ハンドラ２０６に転送する。各要求ハンドラ２０６は、アプリケーションレベル要求２０８の一部として受信されたアプリケーションレベル命令に従って、またその関連するデータ２１０に従って、アプリケーションを実行することができる。

【００３２】

図１に関しては、より具体的な分散コンピューティング環境１００が示されている。図２の一般的なコンピューティング環境２００の状況において、リクエスタ２０２は、ユーザ１０２として実現され、処理中間体２０４は、処理マネジャ１０４として実現され、また要求ハンドラ２０６は、コンピューティング資源１０６として実現され得る。さらに、

アプリケーションレベル要求２０８は、要求１０８として実現され、またデータ２１０は、データ１１０として実現され得る。

【００３３】

コンピューティング環境２００に関して説明する実施形態では、リクエスタ２０２、中間体２０４、および要求ハンドラ２０６が１つまたは複数のネットワーク（図２では明示的に示されていない）により相互接続される。このようなネットワーク（複数可）の通信リンクを用いて、アプリケーションレベル要求２０８は、それが、要求されたタスクを実施できる少なくとも１つの要求ハンドラ２０６に提供されるまで、中間体２０４の間で、またその中で転送される。

【００３４】

図２で示すように、また例示のために過ぎないが、アプリケーションレベル要求Ａ　２０８（Ａ）は、リクエスタ２０２により生成される。リクエスタ２０２は、アプリケーションレベル要求Ａ　２０８（Ａ）を中間体＃１　２０４（１）に送信する。アプリケーションレベル要求２０８は、何らかのトランスポートプロトコルに従って通信されるメッセージの一部として送信され得る。利用されるトランスポートプロトコルは、アプリケーションレベル通信の参加者２０２、２０４、および２０６の間で異なる可能性がある。

【００３５】

着信する要求２０８（Ａ）の何らかの操作の後、中間体＃１　２０４（１）は、アプリケーションレベル要求Ｂ　２０８（Ｂ）を中間体＃２　２０４（２）に送信する。着信する要求２０８（Ｂ）の何らかの操作の後、中間体＃２　２０４（２）は、アプリケーションレベル要求Ｃ　２０８（Ｃ）を中間体＃３　２０４（３）に送信する。さらに、着信する要求２０８（Ｂ）の何らかの（おそらく異なる）操作の後、中間体＃２　２０４（２）はまた、アプリケーションレベル要求Ｄ　２０８（Ｄ）を中間体＃４　２０４（４）に送信する。

【００３６】

中間体＃３　２０４（３）は、アプリケーションレベル要求Ｃ　２０８（Ｃ）の要求されたタスクを２つの部分に分離する。それは、第１の部分を第１の要求ハンドラ２０６（１）に、また第２の部分を第２の要求ハンドラ２０６（２）に転送する。中間体＃４　２０４（４）は、アプリケーションレベル要求Ｄ　２０８（Ｄ）の要求されたタスクを第３の要求ハンドラ２０６（３）に転送する。したがって、要求ハンドラ２０６（１）、２０６（２）、および２０６（３）はそれぞれ、アプリケーションレベル要求Ａ　２０８（Ａ）の元の要求されたタスクを実施することになる。

【００３７】

図３Ａ、３Ｂ、および３Ｃは、図２の例示的なコンピューティング環境で示されたものなど、例示的なアプリケーションレベル要求２０８を示すブロック図である。具体的には、図３Ａは、アプリケーションレベル要求Ａ　２０８（Ａ）の例を示す。図３Ｂは、アプリケーションレベル要求Ｂ　２０８（Ｂ）の例を示す。図３Ｃは、アプリケーションレベル要求Ｃ　２０８（Ｃ）の例を示す。

【００３８】

図３Ａに関して説明する実施形態では、アプリケーションレベル要求Ａ　２０８（Ａ）は、リクエスタからのアプリケーション固有の命令３０２およびリクエスタ署名３０４を含む。リクエスタからのアプリケーション固有の命令３０２は、実行されることを要求するタスクを規定する情報である。例えば、それは、実行されるアプリケーション、およびアプリケーションが実行するデータ、ならびに任意の必要なセキュリティ資格証明書を規定することができる。より具体的には、リクエスタからのアプリケーション固有の命令３０２は、例示のためであり、これだけに限らないが、実行要件、初期化パラメータ、必要なデータソース、セキュリティコンテキスト情報などを含むことができる。リクエスタ２０２は、リクエスタからのアプリケーション固有の命令３０２に関する情報を確認し、アプリケーションレベル要求Ａ　２０８（Ａ）を作成する。

【００３９】

リクエスタ署名３０４は、リクエスタ２０２による、リクエスタからのアプリケーション固有の命令３０２に適用されるデジタル署名である。言い換えると、デジタル署名手順が、アプリケーションレベル情報に適用される。その結果、中間体＃１　２０４（１）に対応するエンティティなど、アプリケーションレベル要求Ａ　２０８（Ａ）をその後に続いて受信し、処理するエンティティは、リクエスタからのアプリケーション固有の命令３０２に対して、認証検査および完全性の検証を実施することができる。認証検査は、どのリクエスタ２０２がそのアプリケーション固有の命令３０２を生成したかを決定する。完全性の検証は、リクエスタからのアプリケーション固有の命令３０２の情報が、リクエスタ２０２により生成（また署名）された後に変更されていないことを検証する。

【００４０】

図３Ｂに関して説明する実施形態では、アプリケーションレベル要求Ｂ　２０８（Ｂ）は、中間体＃１からのアプリケーション固有の命令３０６（１）、中間体＃１の署名３０８（１）、およびアプリケーションレベル要求Ａ　２０８（Ａ）を含む。中間体＃１　２０４（１）は、リクエスタ２０２からの着信するアプリケーションレベル要求Ａ　２０８（Ａ）を受け入れ、また何らかの操作の後に、それを中間体＃２　２０４（２）に転送すべきであると決定する。中間体＃１　２０４（１）は、アプリケーションレベル要求Ａ　２０８（Ａ）をアプリケーションレベル要求Ｂ　２０８（Ｂ）中に有効にカプセル化する。

【００４１】

中間体＃１　２０４（１）は、アプリケーションレベル要求２０８を、補足的なアプリケーション固有の命令３０６をそれに追加することにより増補する。これらの補足的な命令は、中間体＃１からのアプリケーション固有の命令３０６（１）として示される。それらは、図２の例における中間体＃２　２０４（２）を含むことのできる、少なくとも１つの後続する受信者により処理されるように意図されている。中間体＃１　２０４（１）はまた、中間体デジタル署名３０８を用いて、アプリケーションレベル要求２０８にデジタル的に署名する。このデジタル署名は、中間体＃１署名３０８（１）として示されている。中間体＃１署名３０８（１）は、アプリケーションレベル要求Ｂ　２０８（Ｂ）のアプリケーション固有の情報に対する署名である。これは、例えば、中間体＃１からのアプリケーション固有の命令３０６（１）および／またはアプリケーションレベル要求Ａ　２０８（Ａ）を含むことができる。例えば、中間体＃　２０４（１）が、そのアプリケーション固有の命令およびリクエスタ２０２により提供されたものが、独立して変更されていないことを保証する理由が存在する場合、中間体＃１　２０４（１）はまた、アプリケーションレベル要求Ａ　２０８（Ａ）にデジタル的に署名することができる。

【００４２】

図３Ｃに関して説明する実施形態では、アプリケーションレベル要求Ｃ　２０８（Ｃ）は、中間体＃２からのアプリケーション固有の命令３０６（２）、中間体＃２署名３０８（２）、およびアプリケーションレベル要求Ｂ　２０８（Ｂ）を含む。中間体＃２　２０４（２）は、中間体＃１　２０４（１）から着信するアプリケーションレベル要求Ｂ　２０８（Ｂ）を受け入れ、それを、少なくとも部分的に、また何らかの操作の後に、中間体＃３　２０４（３）に転送すべきであると決定する。中間体＃２　２０４（２）は、アプリケーションレベル要求Ｂ　２０８（Ｂ）をアプリケーションレベル要求Ｃ　２０８（Ｃ）中に有効にカプセル化する。

【００４３】

中間体＃２　２０４（２）は、アプリケーションレベル要求２０８を、補足的なアプリケーション固有の命令３０６をそれに追加することにより増補する。これらの補足的な命令は、中間体＃２からのアプリケーション固有の命令３０６（２）として示されている。中間体＃２　２０４（２）はまた、中間体デジタル署名３０８を用いて、アプリケーションレベル要求２０８にデジタル的に署名する。このデジタル署名は、中間体＃２署名３０８（２）として示されている。中間体＃２署名３０８（２）は、アプリケーションレベル要求Ｃ　２０８（Ｃ）のアプリケーション固有の情報に対する署名である。これは、例え

ば、中間体＃２からのアプリケーション固有の命令３０６（２）、および／またはアプリケーションレベル要求Ｂ　２０８（Ｂ）を含むことができる。個別に示されていないが、アプリケーションレベル要求Ｄ　２０８（Ｄ）は、アプリケーションレベル要求Ｃ　２０８（Ｃ）に類似させた形で作成することができる。

【００４４】

したがって、図２および３Ａ～３Ｃで示すように、アプリケーションレベル命令プロトコルの前述の実施形態は、着信するアプリケーションレベル要求２０８を有効にカプセル化し、送出するアプリケーションレベル要求２０８を生成する。送出するアプリケーションレベル要求は、アプリケーション固有の情報の少なくとも一部に対するデジタル署名３０８を含む。それはまた、補足的なアプリケーション固有の命令３０６を含むことができる。アプリケーションレベル要求２０８が、参加するアプリケーションレベル通信のノードまたはエンティティを介して伝送されると、図３Ｃにより特に示されるように、アプリケーションレベル要求２０８のネストされた組が作成される。

【００４５】

各ネストされた要求に対するデジタル署名３０４および３０８と結合されたアプリケーションレベル要求２０８のこのネスティングは、参加者に、要求送信の連鎖を通して、アプリケーション固有の情報の認証および完全性の検証を実施することを可能にする。しかし、少なくとも最後の要求ハンドラ２０６（図２による）は、アプリケーションレベル要求２０８に従って要求されたタスクを実施するためにデータ２１０にアクセスする必要があり得る。いくつかの実施形態では、データ２１０にアクセスするには、データ２１０にアクセスするための権利が許可されている必要があり得る。したがって、このような実施形態では、要求ハンドラ２０６により要求されたタスクを実施することは、まず、データ２１０にアクセスするための権利を実施者（implementer）に許可することが必要となるはずである。前述の実施形態では、データ２１０へのこのアクセス権は、委任アクセス制御機構により許可され得る。

【００４６】

図４は、データにアクセスするための権利の委任を含む例示的なアプリケーションレベル要求Ｂ　２０８（Ｂ）＊を示すブロック図である。アプリケーションレベル要求Ｂ　２０８（Ｂ）＊は、データ委任の権利情報が含まれたアプリケーションレベル要求Ｂ　２０８（Ｂ）（図３Ｂによる）である。本明細書の以下でさらに説明するように、任意のアプリケーションレベル要求２０８は、例えば、リクエスタからの権利の委任４０２および／または中間体からの権利の委任４０４の形で、データ委任の権利情報を含むことができる。

【００４７】

図示のように、アプリケーションレベル要求Ｂ　２０８（Ｂ）＊は、中間体＃１からの権利の委任４０４（１）を含み、また要求ネスティングにより、アプリケーションレベル要求Ａ　２０８（Ａ）の一部である、リクエスタからの権利の委任４０２を有効に含む。前述の実施形態では、リクエスタ２０２が、アプリケーションレベル要求Ａ　２０８（Ａ）を作成する場合、それは、リクエスタからの権利の委任４０２を含む。リクエスタ２０２は、そのデータ２１０に対するアクセス権を有する。リクエスタ２０２は、どの要求ハンドラ（複数可）２０６が最終的にデータ２１０にアクセスするための権利が必要となるかを知らない可能性があるので、リクエスタ２０２は、データ２１０へのアクセス権を直接許可することはできないはずである。したがって、リクエスタ２０２は、データ２１０に対するアクセス権を委任するための権利を、後続するアプリケーションレベル通信の参加者（例えば、処理中間体２０４）に許可する。

【００４８】

データ委任情報（例えば、リクエスタからの権利の委任４０２、中間体からの権利の委任４０４など）は、第２の参加者へと転送される、または延長され得る、第１の参加者への委任権を許可することである。言い換えると、（アプリケーションレベル要求Ｂ　２０８（Ｂ）＊の）アプリケーションレベル要求Ａ　２０８（Ａ）の場合、リクエスタ２０２

は、下流の通信参加者に、データ２１０へのアクセス権をさらに許可するための権利を、中間体＃１　２０４（１）に委任する。中間体＃１　２０４（１）は、中間体＃１からの権利の委任４０４（１）を追加することによって、この委任権を利用する。言い換えると、アプリケーションレベル要求Ｂ　２０８（Ｂ）＊の場合、中間体＃１　２０４（１）は、データ２１０へのアクセス権を、他の下流の通信参加者にさらに許可するための権利を中間体＃２　２０４（２）に委任する。この委任の連鎖は、選択された要求ハンドラ２０６に、データ２１０へのアクセス権が許可される（例えば、中間体＃３　２０４（３）および／または中間体＃４　２０４（４）により）まで延長され得る。この委任情報４０２および／または４０４が、許可されない参加者に開示されるべきではない秘密（例えば、パスワード、暗号化鍵など）を含む場合、このような情報は、次の処理エンティティがそれを解読できるように暗号化することができる。この処理エンティティは、次いで、次の処理エンティティのために、その秘密情報を再度暗号化することができ、また要求を増補するためのデータ委任情報をそれに含めることができる。

【００４９】

したがって、いくつかの前述の実施形態は、要求を生成するリクエスタ、要求のハンドラ、およびその要求を処理し、リクエスタと要求ハンドラの間で転送する１つまたは複数の中間体の間の対話のための一般的なアプリケーションレベルのセキュリティプロトコルを提供する。アプリケーションレベルのセキュリティプロトコルの前述の実施形態は、これらのエンティティが、要求の発行に先立って知られている状況において使用することができる。さらにいくつかの前述の実施形態は、これらのエンティティがアプリオリに知られていないが、その要求がリクエスタと最終的な要求ハンドラの間で伝わるにつれて、エンティティが徐々に確立される、より複雑な場合を扱うことができる。

【００５０】

図２における要求２０８のフローは、本明細書で説明するアプリケーションレベルのセキュリティプロトコルにより処理できる一般のメッセージフローパターンの一例を示す。前述のように、リクエスタ２０２から要求ハンドラ２０６への要求のフローは、有向グラフを形成する。いずれの中間体２０４においても、要求２０８は、潜在的に、複数のエンティティ（例えば、１つまたは複数の他の中間体２０４および／または要求ハンドラ２０６）に転送することができる。リクエスタ２０２から要求ハンドラ２０６への各フローは、論理的に別個のものとして処理され得る。図２に示すコンピューティング環境２００は、したがって、３つの論理的に別々のフローを有していると見なすことができる。１つのフローに対する要求処理を他のフローで再使用できる可能性があることにより、実施効率を改善することができるが、このような再使用は、アプリケーションレベルのセキュリティプロトコルに影響を与えないで済む。したがって、以下の本明細書の説明では、明確化のために、単一の要求フローだけを扱う。

【００５１】

前述のように、アプリケーションレベル要求２０８に対するネスティングプロセスは、デジタル署名３０４および３０８と併せて、リクエスタ２０２と、所与の要求メッセージを処理した任意の前の中間体２０４とを認証するための機構を提供する。さらに、ネスティングおよびデジタル署名は、元の要求と、中間体により追加された任意のアプリケーションレベルの処理命令とにおけるアプリケーションレベル情報の完全性を独立して検証するための機構を提供する。

【００５２】

前述の実施形態では、メッセージ構成は、ネスティングプロセスを用いて達成され、またこのネスティングプロセスは、デジタル署名技術と結合される。デジタル署名は、例えば、公開鍵暗号技術に基づくことができる。以下のテキストおよび例示的なメッセージ要求フォーマットでは、リクエスタ２０２はリクエスタＲと称し、中間体＃１～＃ｎ　２０４（１・・・ｎ）は中間体Ｍ１～Ｍｎと称し、また要求ハンドラ２０６は要求ハンドラＲＨと称する。デジタル署名とネスティングを組み合わせたこの手法を用いて、鍵K_Rを有するリクエスタＲは、最初の中間体Ｍ１に、

【００５３】
【表１】

Request

Signed by K_R .

【００５４】
を含むメッセージを送る。「要求」は、リクエスタからのアプリケーション固有の命令３０２（図３Ａ～４による）に対応する。「K_Rによる署名」は、リクエスタ署名３０４に対応する。リクエスタ署名３０４は、例えば、ＳＨＡ－１ダイジェストアルゴリズムを用いた、要求コンテンツ情報に対するＲＳＡ署名を用いて実施することができる。
【００５５】
中間体Ｍ１は、その署名を検査して要求がリクエスタＲから来ていることを認証し、送信中にそれが変更されていないことを検証することができる。鍵K_{M1}を有する中間体Ｍ１が、鍵K_{M2}を有する第２の中間体Ｍ２に要求を転送することを決定した場合、中間体Ｍ１は、
【００５６】
【表２】

M1 instructions for M2

Request

Signed by K_R

Signed by K_{M1} .

【００５７】
を含むメッセージを送る。「Ｍ２に対するＭ１命令」は、中間体＃２　２０４（２）に対して中間体＃１　２０４（１）により追加された中間体＃１からのアプリケーション固有の命令３０６（１）（図３Ｂ～４による）に対応する。「K_{M1}による署名」は、中間体＃１の署名３０８（１）に対応する。「Ｍ２に対するＭ１命令」（または中間体３０６からの任意のアプリケーション固有の命令）は、実施者Ｍ１により、さらなるアプリケーション固有の命令が何も追加されない場合は、ヌルとなり得る。
【００５８】
中間体Ｍ２は、次に、２つの署名（３０４および３０８（１））を使用して、［１］「Ｍ２に対するＭ１命令」が中間体Ｍ１から生成されたこと、［２］中間体Ｍ１が元の要求を有していたこと、［３］元の要求がリクエスタＲから来たこと、および［４］署名が適用されたため、何も変更されていないことを判定することができる。中間体Ｍ２は、次いで、この情報を使用して、中間体Ｍ１とリクエスタＲの両方のために、このような要求を進んで処理するかどうか判定することができる。
【００５９】
この手法は、最終的に、実際の要求ハンドラに到達するまで、「ｎ」個の中間体Ｍを通して継続される、ただし、「ｎ」は何らかの整数である。この実際の要求ハンドラは、

【００６０】
【表３】

Mn instructions for Request Handler

Mn-1 instructions for Mn

.....

M1 instructions for M2

Request

Signed by K_R

Signed by K_{M1}

....

Signed by K_{Mn-1}

Signed by K_{Mn} .

【００６１】
を含むメッセージを受け取る。
【００６２】
暗号化デジタル署名K_RおよびK_{M1}・・・K_{Mn}は、中間体および要求ハンドラによりアプリケーションレイヤで理解されたデータに対して行われる。これは、送信されたメッセージを認証し、完全性を保護するために、通信のセキュリティ保護に広く使用されるネットワークのセキュリティプロトコルのデジタル署名とは全く異なる。これらの広く使用されるメッセージのデジタル署名は、通常、使用される特定の低レイヤセキュリティプロトコル用に符号化されたメッセージコンテンツおよびメッセージヘッダを共にカバーする。したがって、ヘッダは、概して、諸中間体にわたって意味を有しておらず、また使用されるメッセージングプロトコルは、要求フローに参加するすべてのエンティティ間で同一ではない可能性があるため、このようなメッセージデジタル署名は、これらのアプリケーションに対して使用することができない。
【００６３】
図４を参照して前述したように、データアクセスに対する権利の委任はまた、動的に発見されたメッセージフローの参加者に即して、アプリケーションレベルのセキュリティプロトコルの使用を実行することができる。いくつかの前述の実施形態は、したがって、様々な参加エンティティ間で、委任情報を渡すための機構を提供する。このデータ委任情報

を渡すことは、フロー中のエンティティを動的に発見できることに対処する。言い換えると、データ委任情報を渡すことは、リクエスタが最初に要求を作成する場合、データのアクセス権を、どの要求ハンドラに委任すべきかをリクエスタが直接示す方法がない可能性のあることに対処している。

【００６４】

リクエスタからの権利の委任４０２および／または中間体からの権利の委任４０４など、セキュリティ資格証明書情報を暗号化することができる。各処理中間体２０４は、それらを解読し、任意の適切な分析を行い、それらをおそらく変更し、次いで、要求２０８を転送する前に再度暗号化することができる。暗号化は、任意の一般に使用される暗号（例えば、非特許文献１参照）または特別に適応された暗号を用いて適用することができる。関連する解読鍵は、一般に利用可能な技法（例えば、非特許文献２で説明されているＲＳＡ鍵トランスポート）、または特別に設計された技法を用いて通信することができる。関連する解読鍵はまた、ＡＥＳ鍵ラップ（例えば、非特許文献３）を用いて通信することもできる。他の暗号化、解読、および鍵トランスポート手法を、代替的に使用することもできる。

【００６５】

権利の委任を伴う分散されたアプリケーション情報配信のセキュリティ保護に関するいくつかの前述の実施形態は、リクエスタのデータアクセス権を要求ハンドラに委任するためにどの機構が使用されるかに関して全く分かっていない。例示的な委任機構は、例としてであり、これに限らないが、（ｉ）ＭｙＰｒｏｘｙサービスで使用される物など、委任資格証明書をアンロックするために使用される名前－パスワードの対、（ｉｉ）ＩＳＯのＲＥＬ（Ｒｉｇｈｔｓ Ｅｘｐｒｅｓｓｉｏｎ Ｌａｎｇｕａｇｅ；権利記述言語）などのポリシー言語を用いて、各処理中間体で生成される一連の明示的な委任ポリシー／資格証明書、（ｉｉｉ）Ｍｉｃｒｏｓｏｆｔ（登録商標）からのＳｅｃＰＡＬ（商標）（Ｓｅｃｕｒｉｔｙ Ｐｏｌｉｃｙ Ａｓｓｅｒｔｉｏｎ Ｌａｎｇｕａｇｅ）言語、（ｉｖ）それらのいくつかの組合せなどを含む。例示的な実施形態は、グリッドコンピューティング分散ジョブ管理のために開発されている１つまたは複数のＳＯＡＰベースのウェブのサービスプロトコルと共に使用されるＸＭＬ符号化を使用することができる。

【００６６】

前述の例示的な手法は、リクエスタ、および各処理中間体に所望の委任を符号化させて、メッセージ通信フローに参加している次のエンティティに送ることである。その機構が、名前－パスワード資格証明書およびＭｙＰｒｏｘｙサービスを含む場合、それは、暗号化されてフロー中の次のエンティティに送られる名前－パスワードの形を取ることに加えて、使用するＭｙＰｒｏｘｙサービスへの参照の形を取る。その機構が、ポリシー言語手法の１つを含む場合、それはフロー中に参加している次のエンティティが、必要なデータにアクセスする権利を有すること、および／またはこれらの権利を他の者に委任することを示す資格証明書の作成を伴う。

【００６７】

そうではあるが、このようなデータ委任資格証明書は、リクエスタまたは現在の処理中間体により「発行される」、またはデジタル的に署名される。権利の委任情報は、次いで、（図４の組合せで示されるように）前述の認証情報と組み合わされて、アプリケーションレベルのセキュリティプロトコルの前述の他の実施形態を形成することができる。例示のためだけであるが、権利の委任コンポーネントを有するアプリケーションレベルのセキュリティプロトコルの前述の実施形態は、概して、以下のようにフォーマット化することができる、ただし、「データアクセス権」は「ＤＡＲ」により表される。

【００６８】

【表４】

Mn instructions for RH + Mn delegation of DAR to RH

Mn-1 instructions for Mn + Mn-1 delegation of DAR delegation to Mn

.....

M1 instructions for M2 + M1 delegation of DAR delegation to M2

Request + R delegation of DAR delegation to M1

Signed by K_R

Signed by K_{M1}

....

Signed by K_{Mn-1}

Signed by K_{Mn}

【００６９】

図５は、図２の例示的なコンピューティング環境中で示されたものなど、アプリケーションレベル通信の参加者上で実行することができる例示的なアプリケーション５０２を示すブロック図である。図示のように、アプリケーション５０２は、１０個のモジュール５０４～５２２を含む。これらの１０個のモジュールは、受信器５０４、アプリケーション固有の情報抽出器５０６、アプリケーション固有の情報分析器５０８、メッセージ参加者認証器５１０、メッセージ情報の完全性検証器５１２、メッセージ増補器５１４、メッセージ署名器５１６、情報暗号化器５１８、情報解読器５２０、および送信器５２２である。

【００７０】

１０個のモジュールが、アプリケーション５０２に関して示され、以下で説明されているが、アプリケーション命令に対するアプリケーションレベルのセキュリティプロトコル中で参加エンティティとして機能しているアプリケーションは、任意の数のモジュールを含むことができる。以下の説明は、主として、中間体２０４として機能しているアプリケーション５０２を対象とする。そうではあるが、アプリケーション５０２の機能は、リクエスタ２０２、要求ハンドラ２０６などの他のエンティティに対しても同様の物であり得る。しかし、いくつかの違いが存在する可能性がある。例えば、要求ハンドラ２０６は、メッセージ増補器５１４またはメッセージ署名器５１６を必要としない可能性がある。リクエスタ２０２は、要求２０８を生成するので、アプリケーション固有の情報抽出器５０６またはアプリケーション固有の情報分析器５０８を必要としない可能性がある。他方で、リクエスタ２０２のアプリケーション５０２中に両方を含むことは、リクエスタそれ自体が、要求２０８の送信経路の何らかの要求追跡の法的な分析を行うことを可能にする。

【００７１】

前述の実施形態では、受信器５０４は、アプリケーションレイヤよりも低位のコンピュータの通信スタックレイヤから、着信する要求２０８を受け入れる。同様に、送信器５２２は、低位レイヤの通信トランスポートプロトコルを用いて、他の中間体２０４または要求ハンドラ２０６に転送するために、送出する要求２０８を、アプリケーションレイヤから通信スタックの低位レイヤに送る。

【００７２】

前述の実施形態では、アプリケーション固有の情報抽出器５０６は、着信する要求２０８からアプリケーション固有の情報を抽出する。アプリケーション固有の情報の例には、これだけに限らないが、リクエスタからのアプリケーション固有の命令３０２、中間体か

らのアプリケーション固有の命令３０６、リクエスタからの権利の委任４０２、中間体からの権利の委任４０４などが含まれる。アプリケーション固有の情報分析器５０８は、抽出されたアプリケーション固有の情報を分析して、その要求を次にどこに転送すべきかを決定する。次の参加ノードは、例えば、他の中間体または要求ハンドラとすることができる。アプリケーション固有の情報分析器５０８はまた、その抽出されたアプリケーション固有の情報を分析して、必要な場合、送出する要求２０８用として、この次のノードのためにどのような追加のアプリケーション固有の命令を着信する要求２０８に追加すべきかを決定する。

【００７３】

メッセージ参加者認証器５１０は、要求２０８および／または増補的アプリケーション固有の情報の元を認証するためにデジタル署名を使用する。したがって、メッセージ参加者認証器５１０は、リクエスタ署名３０４を使用して、リクエスタからのアプリケーション固有の命令３０２を有する元の要求２０８（Ａ）が、リクエスタ２０２によって開始されたことを認証することができる。それはまた、中間体＃１の署名３０８（１）を使用して、アプリケーション固有の命令３０６（１）を有するカプセル化された要求２０８（Ｂ）が、中間体＃１　２０４（１）から転送されたことを認証することができる。

【００７４】

メッセージ情報の完全性検証器５１２は、それぞれがネストされたデジタル署名３０４および３０８を使用して、それぞれがネストされたアプリケーション固有の情報の完全性を検証する。より具体的には、メッセージ情報の完全性検証器５１２は、リクエスタ署名３０４を使用して、リクエスタからのアプリケーション固有の命令３０２および／またはリクエスタからの権利の委任４０２の完全性を検証する。メッセージ情報の完全性検証器５１２はまた、中間体署名３０８を使用して、中間体からのアプリケーション固有の命令３０６および／または中間体からの権利の委任４０４を検証することができる。

【００７５】

メッセージ増補器５１４は、任意のさらなる所望の処理命令を追加する。例えば、メッセージ増補器５１４は、次のエンティティ受信者（例えば、中間体または要求ハンドラ）に対する新しいアプリケーション固有の命令３０６を追加することができ、および／または次のエンティティ受信者に対するデータアクセス権を含むデータ委任権４０４を追加することができる。メッセージ署名器５１６は、アプリケーションレベル情報にデジタル的に署名して、中間体署名３０８を作成する。デジタル署名手順は、次のエンティティ受信者用である、中間体からのアプリケーション固有の命令３０６など、増補的情報に対して適用することができる。代替的には、デジタル署名手順はまた、要求２０８が作成され、命令が追加されている順序の検証を提供するように、ネストされたアプリケーションレベル要求２０８に対して実施することもできる。

【００７６】

暗号化および解読は、それぞれ、情報暗号化器５１８および情報解読器５２０で処理される。情報は、情報解読器５２０により解読することができる。情報は、情報暗号化器５１８により暗号化され、および／または再度暗号化され得る。情報は、例えば、セキュリティ資格証明書情報とすることができる。より一般的には、情報は、これだけに限らないが、アプリケーション固有の命令３０２および／または３０６、権利の委任４０２および／または４０４などを含む所与の任意のデータとすることができる。

【００７７】

図６は、アプリケーションレベルの要求情報を安全に通信するための方法の例を示す流れ図６００である。流れ図６００は８個のブロック６０２～６１６を含む。流れ図６００のアクションは他の環境中で、また様々なハードウェアおよびソフトウェアの組合せを用いて実施することができるが、図２～５のいくつかの態様は、流れ図６００の方法の一例を示すために使用される。例えば、流れ図６００のアクションは、処理中間体２０４により実施され得る。

【００７８】

ブロック６０２では、アプリケーションレベルで、リクエスタによりデジタル的に署名された要求を有する着信メッセージが受信される。例えば、中間体＃２　２０４（２）は、リクエスタ２０２により少なくとも部分的に署名されたアプリケーションレベル要求Ｂ　２０８（Ｂ）を有するメッセージを受信することができる。アプリケーションレベル要求Ｂ　２０８（Ｂ）は、リクエスタ署名３０４を含むアプリケーションレベル要求Ａ　２０８（Ａ）をカプセル化する。

【００７９】

ブロック６０４で、受信された要求が前の中間体からの任意のアプリケーション固有の命令を有するかどうかが判定される。例えば、アプリケーションレベル要求Ｂ　２０８（Ｂ）が、前の中間体３０６からの任意のアプリケーション固有の命令を含むかどうかを判定することができる。図３Ｂで示すように、アプリケーションレベル要求Ｂ　２０８（Ｂ）は、中間体＃１からのアプリケーション固有の命令３０６（１）を含む。

【００８０】

受信されたメッセージが、前の中間体からのアプリケーション固有の命令を有する場合、ブロック６０６で、そのアプリケーション固有の中間体命令が要求から抽出される。例えば、中間体＃１からのアプリケーション固有の命令３０６（１）を抽出することができる。

【００８１】

ブロック６０６の後、またはブロック６０４で「Ｎｏ」の判断の後、アプリケーション固有のリクエスタ命令が、ブロック６０８で、要求から抽出される。例えば、リクエスタからのアプリケーション固有の命令３０２が抽出され得る。したがって、ブロック６０６および６０８の後、いずれのアプリケーション固有の命令も、それがリクエスタから発信されたものであろうと、前の中間体からのものであろうと抽出されている。デジタル署名３０４／３０８および／または権利の委任４０２／４０４などの他のアプリケーション固有の情報もまた、抽出することができる。

【００８２】

ブロック６１０で、抽出されたアプリケーション固有の情報が分析される。例えば、アプリケーション固有のリクエスタ命令３０２および／またはアプリケーション固有の中間体命令３０６が分析されて、その要求を、他の中間体に転送すべきか、要求ハンドラに転送すべきか、またはその一方もしくは両方に転送すべきかを判定することができる。言い換えると、分析は、その要求の後続する受信者となるべき少なくとも１つのエンティティの識別を決定することができる。分析はまた、現在の中間体によって、どのようなアプリケーション固有の命令を要求に追加すべきかを決定することもできる。

【００８３】

デジタル署名保護が使用されている場合、リクエスタ署名３０４および／または中間体署名（複数可）３０８が抽出され、またそれを、アプリケーションレベル要求２０８の完全性を認証し、および／または分析するために使用することができる。データ委任情報が、アプリケーションレベル要求２０８中に含まれる場合、リクエスタからの権利の委任４０２、および／または中間体からの権利の委任４０４が抽出され、また特に、下流の参加者にデータアクセス権をさらに委任するために、分析で使用することができる。

【００８４】

ブロック６１２で、要求が、後続するエンティティに対するアプリケーションレベル命令を追加することにより増補される。例えば、ブロック６１０の分析に応じて、中間体＃２　２０４（２）は、中間体＃２からのアプリケーション固有の命令３０６（２）など、後続する中間体に対するアプリケーション固有の命令３０６を追加することができる。委任の権利が送られている場合、中間体＃２　２０４（２）はさらに、中間体＃２　からの権利の委任４０４を追加することにより要求を増補して、データアクセス権の推移的な委任連鎖を継続することができる。

【００８５】

ブロック６１４では、アプリケーションレベルで追加された命令を有する増補された要

求は、デジタル的に署名され送出メッセージが作成される。例えば、中間体＃２　２０４（２）は、追加されたアプリケーション固有の命令３０６（２）および／またはネストされたアプリケーションレベル要求２０８（ＡおよびＢ）にデジタル的に署名して、中間体＃２の署名３０８（２）を作成することができる。

【００８６】

ブロック６１６で、デジタル的に署名された送出メッセージが、後続するエンティティに向けて送信される。例えば、中間体＃２　２０４（２）は、アプリケーションレベル要求Ｃ　２０８（Ｃ）を中間体＃３　２０４（３）に向けて送信することができる。中間体＃３　２０４（３）はまた、要求ハンドラ２０６（１）および２０６（２）への後続する通信のために、それらの間の論理的関係および信頼関係に応じて、前述のアプリケーションレベルのセキュリティプロトコルの諸態様を適用することもできる。

【００８７】

図７は、分散されたアプリケーション情報配信のセキュリティ保護を実施するために使用できる例示的な装置７０２のブロック図である。複数の装置７０２は、１つまたは複数のネットワーク７１４にわたって通信することができる。ネットワーク７１４は、インターネット、イントラネット、イーサネット（登録商標）、無線ネットワーク、有線ネットワーク、公衆網、専用ネットワーク、ケーブルネットワーク、デジタル加入者回線（ＤＳＬ）ネットワーク、電話ネットワーク、ファイバネットワーク、グリッドコンピュータネットワーク、資源クラスタのネットワーク、そのいくつかの組合せなどとすることができる。

【００８８】

図示のように、２つの装置７０２（１）および７０２（ｎ）は、アプリケーションレベル要求２０８の転送など、ネットワーク７１４を介するメッセージ通信の送信に従事することができる。２つの装置７０２が、具体的に示されているが、実施形態に応じて、１つまたは２を超える装置７０２を使用することもできる。リクエスタ２０２、中間体２０４、要求ハンドラ２０６などが、装置７０２として実現され得る。

【００８９】

概して、装置７０２は、任意のコンピュータ、あるいはサーバ装置、ワークステーションもしくは他の一般のコンピュータ装置、データ記憶リポジトリ装置、携帯情報端末（ＰＤＡ）、移動電話、ゲームプラットフォーム、娯楽装置、ルータコンピューティングノード、またはそれらの何らかの組合せなどの処理可能な装置を表すことができる。図示のように、装置７０２は、１つまたは複数の入力／出力（Ｉ／Ｏ）インターフェース７０４、少なくとも１つのプロセッサ７０６、および１つまたは複数の媒体７０８を含む。媒体７０８は、プロセッサ実行可能命令７１０を含む。

【００９０】

装置７０２の前述の実施形態では、Ｉ／Ｏインターフェース７０４は、（ｉ）ネットワーク７１４にわたり通信するためのネットワークインターフェース、（ｉｉ）表示画面上に情報を表示するための表示装置インターフェース、（ｉｉｉ）１つまたは複数のマンマシンインターフェースなどを含むことができる。（ｉ）のネットワークインターフェースの例は、ネットワークカード、モデム、１つまたは複数のポート、ネットワーク通信スタックなどを含む。（ｉｉ）の表示装置インターフェースの例は、グラフィックスドライバ、グラフィックスカード、画面またはモニタ用のハードウェアまたはソフトウェアドライバなどを含む。（ｉｉｉ）のマンマシンインターフェースの例は、有線でまたは無線でマンマシンインターフェース装置７１２（例えば、キーボード、リモート、マウス、または他のグラフィカルなポインティング装置など）に通信する物を含む。

【００９１】

概して、プロセッサ７０６は、プロセッサ実行可能命令７１０などのプロセッサ実行可能命令を実行し、実施し、および／またはその他の形で実現することができる。媒体７０８は、１つまたは複数のプロセッサアクセス可能媒体からなる。言い換えると、媒体７０８は、装置７０２による機能の性能を実現するために、プロセッサ７０６により実行可能

なプロセッサ実行可能命令７１０を含むことができる。

【００９２】

したがって、分散されたアプリケーション情報配信のセキュリティ保護の実現は、プロセッサ実行可能命令の一般的なコンテキストで記述され得る。概して、プロセッサ実行可能命令は、特定のタスクを実施する、および／または可能にする、および／または特定の抽象型を実施するルーチン、プログラム、アプリケーション、コーディング、モジュール、プロトコル、オブジェクト、コンポーネント、メタデータおよびその定義、データ構造、アプリケーションプログラミングインターフェース（ＡＰＩ）などを含む。プロセッサ実行可能命令は、別個の記憶媒体中に位置し、異なるプロセッサにより実行され、および／または様々な送信媒体を介して伝播され、またはその媒体上に存在することができる。

【００９３】

プロセッサ（複数可）７０６は、任意の適用可能な、処理可能技術を用いて実施することができる。媒体７０８は、装置７０２の一部として含まれる、および／または装置７０２によりアクセス可能な任意の利用可能な媒体とすることができる。それは揮発性および不揮発性媒体、取外し可能および取外し不能媒体、記憶および送信媒体（例えば、無線のまたは有線の通信チャネル）を含む。例えば、媒体７０８は、プロセッサ実行可能命令７１０の長期間の大容量記憶のためのディスクアレイ、現在実行されている、および／またはそのほかの形で処理されている命令の短期間記憶のためのランダムアクセスメモリ（ＲＡＭ）、通信を送信するためのネットワーク７１４上のリンク（複数可）などを含むことができる。

【００９４】

具体的に示したように、媒体７０８は、少なくともプロセッサ実行可能命令７１０を含む。概して、プロセッサ実行可能命令７１０は、プロセッサ７０６によって実行された場合、装置７０２に、本明細書で説明した様々な機能を実施させることができる。このような機能は、これだけに限らないが、（ｉ）図２に示されたアプリケーションレベル通信の参加者を実現すること、（ｉｉ）流れ図６００（図６による）で示されたこれらのアクションを実施すること、（ｉｉｉ）図３Ａ、３Ｂ、３Ｃ、および４で示されたこれらのデータ構造（複数可）２０８を実施すること、（ｉｖ）図５で示されたアプリケーション５０２を実現することなどを含む。例のために過ぎないが、プロセッサ実行可能命令７１０は、１つまたは複数のアプリケーションレベル要求２０８、アプリケーション５０２、それらの何らかの組合せなどを含むことができる。

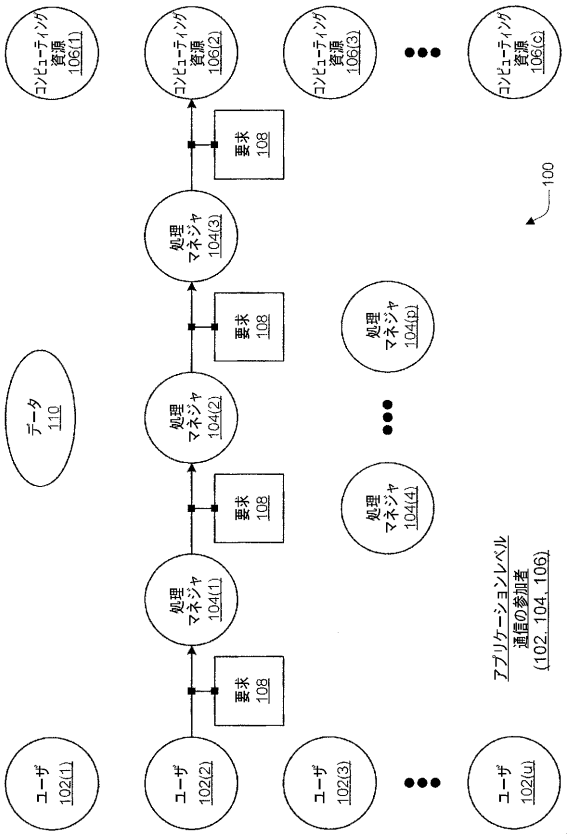
【００９５】

図１～７の装置、アクション、態様、特徴、機能、手順、モジュール、データ構造、プロトコル、コンピューティングシステム、コンポーネントなどは、複数のブロックに分割された図で示されている。しかし、図１～７が説明され、および／または示された順序、相互接続、相互関係、レイアウトなどは、限定的に解釈されることを意図しておらず、分散されたアプリケーション情報配信のセキュリティ保護のために、１つまたは複数のシステム、方法、装置、手順、媒体、装置、ＡＰＩ、構成などを実施するように任意の方法で、任意の数のブロックを変更し、組み合わせ、再配置し、増補し、除外することができる。

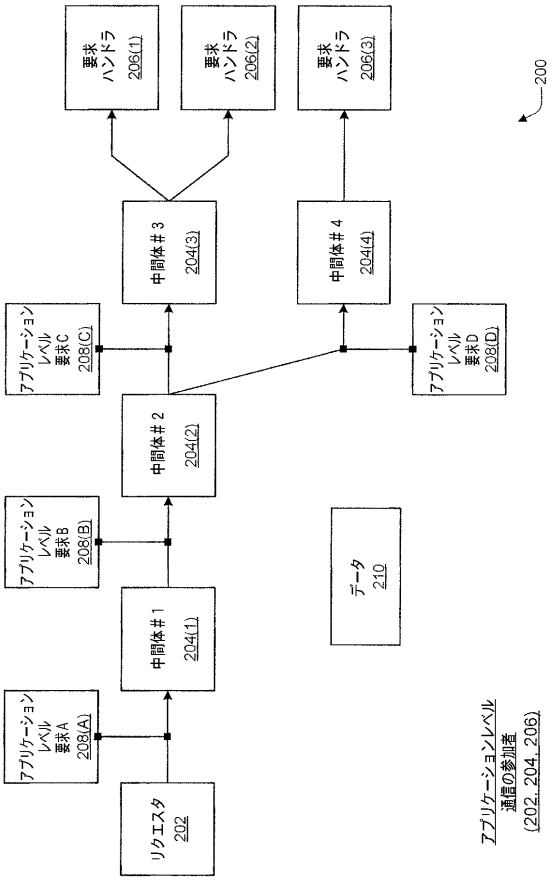
【００９６】

システム、媒体、デバイス、方法、手順、装置、機構、スキーム、手法、プロセス、構成、および他の実施形態が、構造的、論理的、アルゴリズム的、および機能的な特徴および／または図に固有の言語で記述されているが、添付の特許請求の範囲で定義される本発明は、前述の特定の機能または行為に必ずしも限定されないことを理解されたい。そうではなくて、前述の特定の特徴および行為は、特許請求の範囲を実施する例示的な形態として開示される。

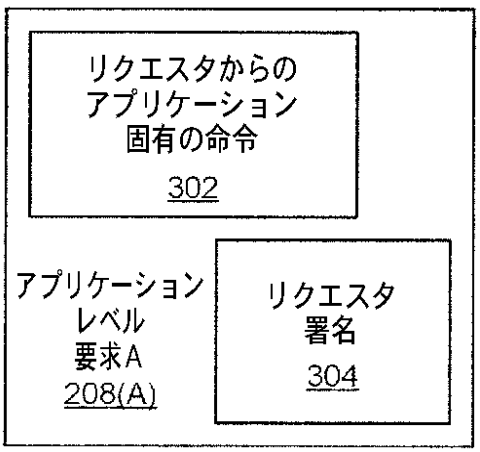
【図１】


コンピューティング資源 106(1)
コンピューティング資源 106(2)
コンピューティング資源 106(3)
コンピューティング資源 106(c)
要求 108
処理マネジャ 104(3)
処理マネジャ 104(p)
処理マネジャ 104(2)
処理マネジャ 104(4)
処理マネジャ 104(1)
データ 110
ユーザ 102(1)
ユーザ 102(2)
ユーザ 102(3)
ユーザ 102(u)
100
アプリケーションレベル通信の参加者 (102, 104, 106)

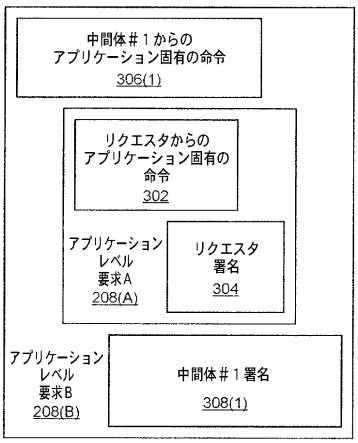
【図２】


要求ハンドラ 206(1)
要求ハンドラ 206(2)
要求ハンドラ 206(3)
200
中間体＃3 204(3)
中間体＃4 204(4)
アプリケーションレベル要求C 208(C)
アプリケーションレベル要求D 208(D)
中間体＃2 204(2)
アプリケーションレベル要求B 208(B)
データ 210
中間体＃1 204(1)
アプリケーションレベル要求A 208(A)
リクエスタ 202
アプリケーションレベル通信の参加者 (202, 204, 206)

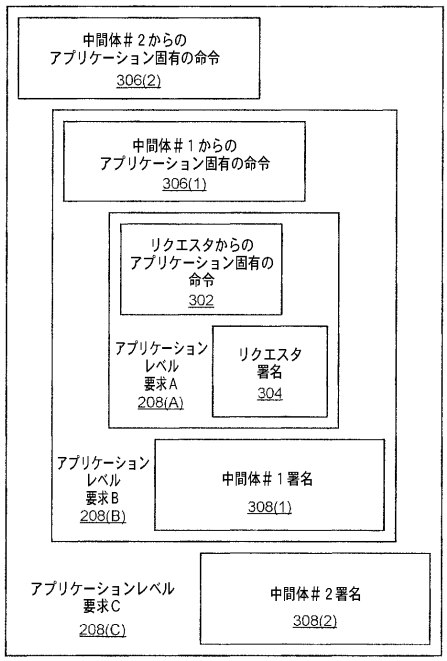
【図３Ａ】


リクエスタからのアプリケーション固有の命令
302
アプリケーションレベル要求A
208(A)
リクエスタ署名
304

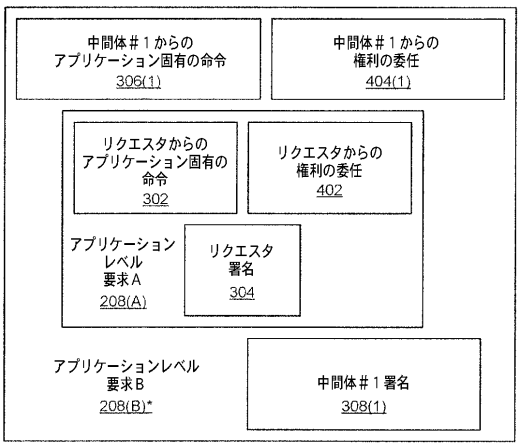
【図３Ｂ】


中間体＃1からのアプリケーション固有の命令
306(1)
リクエスタからのアプリケーション固有の命令
302
アプリケーションレベル要求A
208(A)
リクエスタ署名
304
アプリケーションレベル要求B
208(B)
中間体＃1署名
308(1)

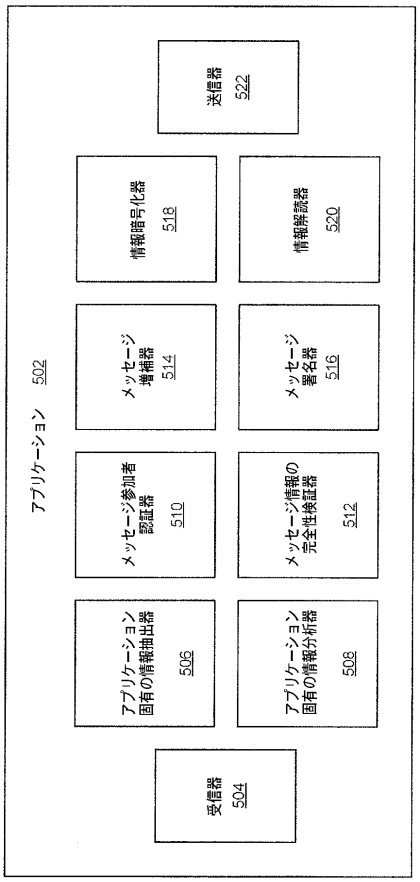
【図３Ｃ】


中間体＃２からの
アプリケーション固有の命令
306(2)
中間体＃１からの
アプリケーション固有の命令
306(1)
リクエスタからの
アプリケーション固有の
命令
302
アプリケーション
レベル
要求Ａ
208(A)
リクエスタ
署名
304
アプリケーション
レベル
要求Ｂ
208(B)
中間体＃１署名
308(1)
アプリケーションレベル
要求Ｃ
208(C)
中間体＃２署名
308(2)

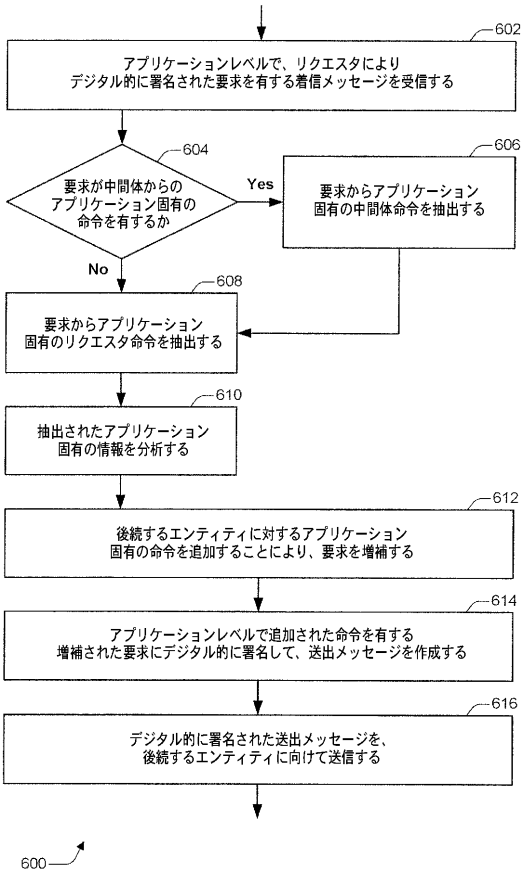
【図４】


中間体＃１からの
アプリケーション固有の命令
306(1)
中間体＃１からの
権利の委任
404(1)
リクエスタからの
アプリケーション固有の
命令
302
リクエスタからの
権利の委任
402
アプリケーション
レベル
要求Ａ
208(A)
リクエスタ
署名
304
アプリケーションレベル
要求Ｂ
208(B)*
中間体＃１署名
308(1)

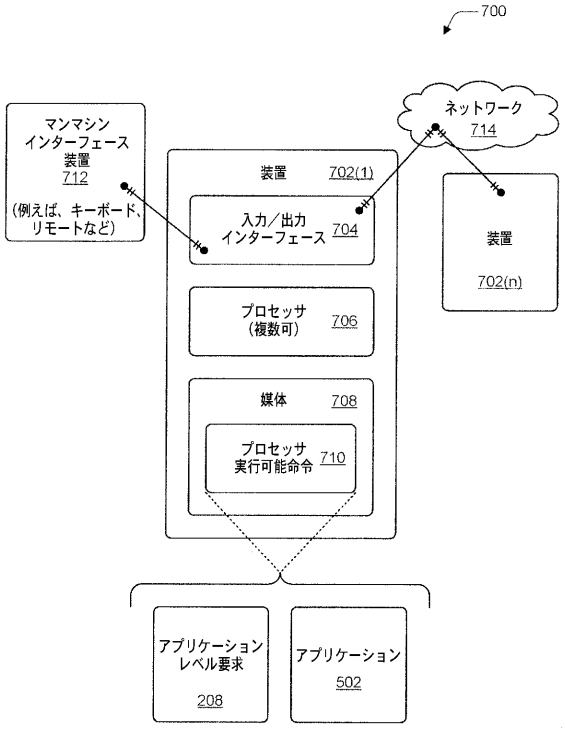
【図５】


アプリケーション 502
送信器 522
情報暗号化器 518
情報解読器 520
メッセージ増補器 514
メッセージ署名器 516
メッセージ参加者認証器 510
メッセージ情報の完全性検証器 512
アプリケーション固有の情報抽出器 506
アプリケーション固有の情報分析器 508
受信器 504

【図６】


602
アプリケーションレベルで、リクエスタにより
デジタル的に署名された要求を有する着信メッセージを受信する
604
要求が中間体からの
アプリケーション固有の
命令を有するか
Yes
606
要求からアプリケーション
固有の中間体命令を抽出する
No
608
要求からアプリケーション
固有のリクエスタ命令を抽出する
610
抽出されたアプリケーション
固有の情報を分析する
612
後続するエンティティに対するアプリケーション
固有の命令を追加することにより、要求を増補する
614
アプリケーションレベルで追加された命令を有する
増補された要求にデジタル的に署名して、送出メッセージを作成する
616
デジタル的に署名された送出メッセージを、
後続するエンティティに向けて送信する
600

【図７】


700
ネットワーク
714
マンマシン
インターフェース
装置
712
(例えば、キーボード、
リモートなど)
装置
702(1)
入力／出力
インターフェース
704
プロセッサ
(複数可)
706
媒体
708
プロセッサ
実行可能命令
710
装置
702(n)
アプリケーション
レベル要求
208
アプリケーション
502

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2007/083390

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/06(2006.01)i, G06F 9/54(2006.01)i, G06F 15/16(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 8 G06F, H04Q

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean Utility models and applications for Utility models since 1975
Japanese Utility models and applications for Utility models since 1975

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKIPASS(KIPO internal) "Keywords: grid computing, message structure, secure protocol, signature and similar terms"

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5,630,129 A (WHEAT et al.) 13 May 1997 See the abstract and figures 1a-1b.	1-20
A	US 2006/150158 A1 (FELLENSTEIN et al.) 06 July 2006 See the abstract, figures 2-5, and paragraphs [0014-0018].	1-20
A	KR 10-2006-0096979 A (IBM CORP.) 13 September 2006 See the abstract, figures 2b-5, and pages 3-9.	1-20
A	US 2005/155033 A1 (LUOFFO et al.) 14 July 2005 See the abstract, figure 5, and paragraphs [0015-0017].	1-20
A	US 2005/166041 A1 (BROWN et al.) 28 July 2005 See the abstract and figure 2.	1-20

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
21 APRIL 2008 (21.04.2008)	**21 APRIL 2008 (21.04.2008)**

Name and mailing address of the ISA/KR
Korean Intellectual Property Office
Government Complex-Daejeon, 139 Seonsa-ro, Seo-gu, Daejeon 302-701, Republic of Korea
Facsimile No. 82-42-472-7140

Authorized officer
NHO, Ji Myong
Telephone No. 82-42-481-8528

Form PCT/ISA/210 (second sheet) (April 2007)

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/US2007/083390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 05630129 A	13.05.1997	None	
US 2006150158 A1	06.07.2006	WO 2006072547 A1	13.07.2006
KR 1020060096979 A	13.09.2006	CN 1701295 A	23.11.2005
		US 2005021956 A1	27.01.2005
		WO 2005003934 A1	13.01.2005
US 2005155033 A1	14.07.2005	CN 1902588 A	24.01.2007
		JP 2007518169 T2	05.07.2007
		WO 2005069138 A1	28.07.2005
US 2005166041 A1	28.07.2005	None	

フロントページの続き

(81)指定国 AP(BW,GH,GM,KE,LS,MW,MZ,NA,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HU,IE,IS,IT,LT,LU,LV,MC,MT,NL,PL,PT,RO,SE,SI,SK,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AT,AU,AZ,BA,BB,BG,BH,BR,BW,BY,BZ,CA,CH,CN,CO,CR,CU,CZ,DE,DK,DM,DO,DZ,EC,EE,EG,ES,FI,GB,GD,GE,GH,GM,GT,HN,HR,HU,ID,IL,IN,IS,JP,KE,KG,KM,KN,KP,KR,KZ,LA,LC,LK,LR,LS,LT,LU,LY,MA,MD,ME,MG,MK,MN,MW,MX,MY,MZ,NA,NG,NI,NO,NZ,OM,PG,PH,PL,PT,RO,RS,RU,SC,SD,SE,SG,SK,SL,SM,SV,SY,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VC,VN,ZA,ZM,ZW

Ｆターム(参考) 5B285 AA01 BA11 CA14 CA43 CA44 CB47
5J104 AA08 AA09 AA16 AA32 AA37 EA04 EA08 JA21 LA06 NA02
NA37 PA14