Helm Security

A Look Below The Hood

SAMSUNG SDS

Raghavan “Rags” N. Srinivas

Java Developer

JavaOne Rockstar

Container/Helm Tinkerer

InfoQ contributor

Matt Farina

Engineer at Samsung SDS
@mattfarina

Helm: Maintainer

Other Open Source: Kubernetes SIG Chair, maintainer of numerous libraries, former OpenStacker, former Drupaler, and more

Work: Cloud Native Computing Team @ Samsung SDS

SAMSUNG SDS

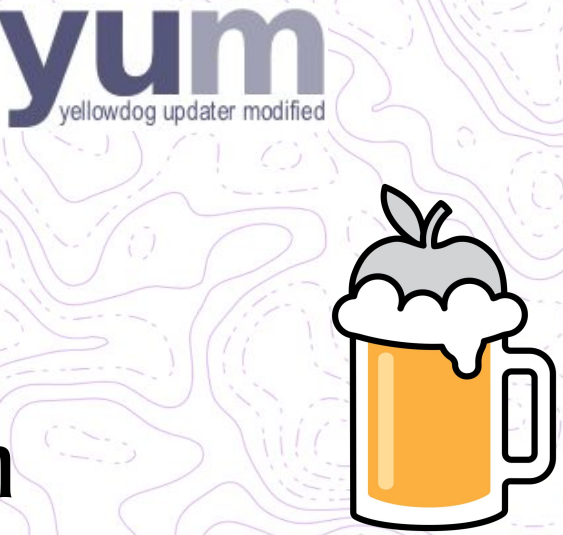
Hayley Denbraver

- *Twitter: **@haylendenb***
- *Developer Advocate at Snyk*
  - *Security education*
  - *Python Security*
  - *New to CNCF ecosystem but happy to be here*

Helm Is The Package Manager For Kubernetes

≈

debian
Apt

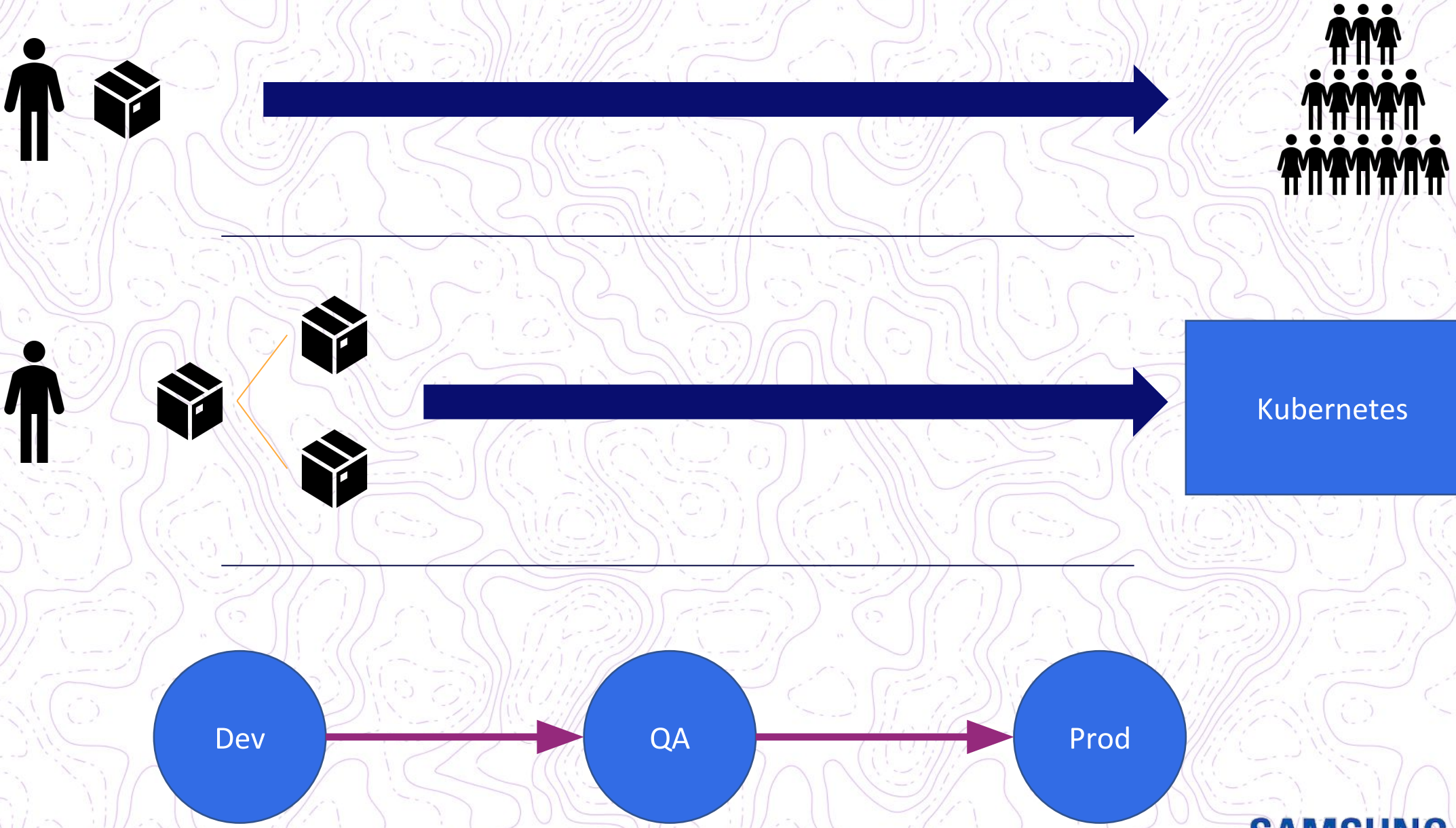


SAMSUNG SDS

What Is A Package Manager?

Package Managers Are Totally Useful

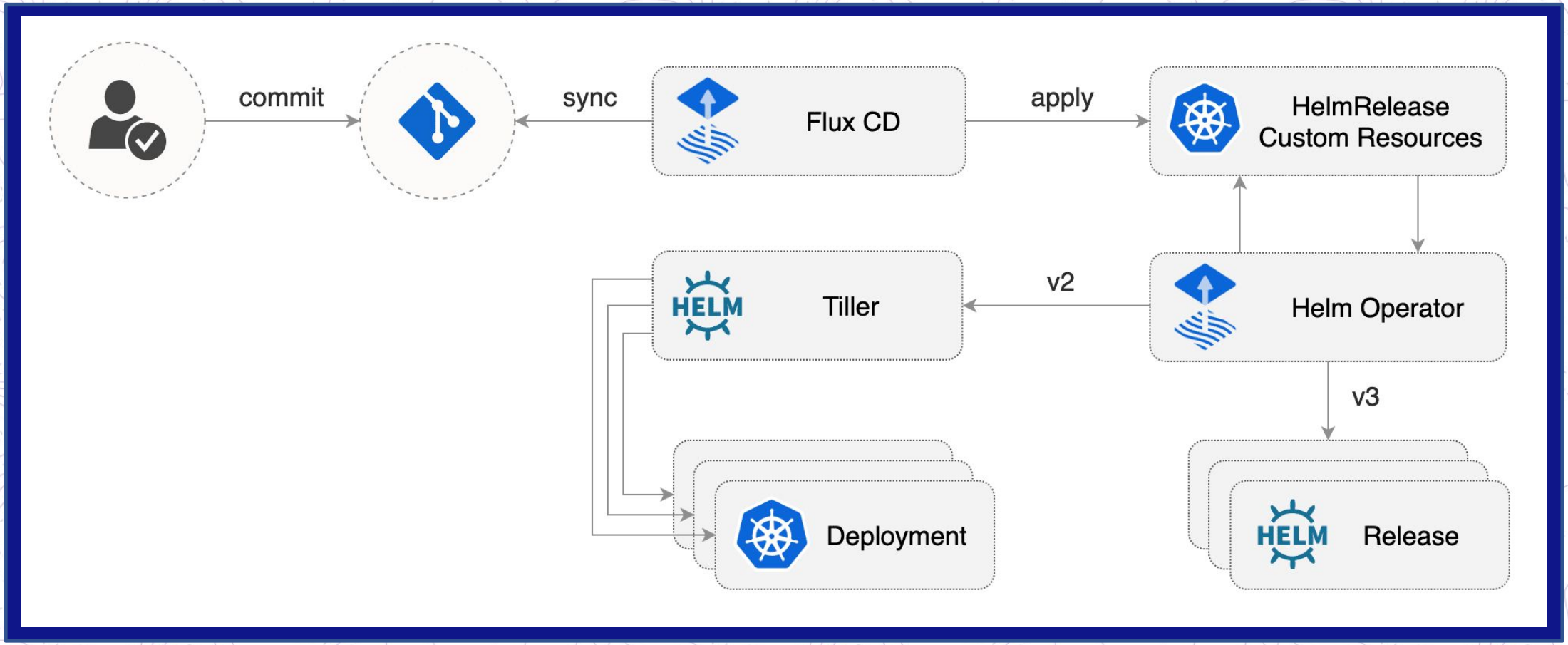
Package Management: Tooling that enables someone who has knowledge of an application and a platform to package up an application so that someone else who has neither extensive knowledge of the application or the way it needs to be run on the platform can use it.



SAMSUNG SDS

A Building Block: Integrated Into Other Tools

An example is https://fluxcd.io/



What About *helm template* And Other Cases?

We know people stretch Helm beyond its design

```
$ helm template foo .
---
# Source: foo/templates/serviceaccount.yaml
apiVersion: v1
kind: ServiceAccount
metadata:
  name: foo
  labels:
    helm.sh/chart: foo-0.1.0
    app.kubernetes.io/name: foo
    app.kubernetes.io/instance: foo
    app.kubernetes.io/version: "1.16.0"
    app.kubernetes.io/managed-by: Helm
---
# Source: foo/templates/service.yaml
apiVersion: v1
kind: Service
…
```

This is very useful but not the core of what Helm provides. When we use the core package Management features, we can use some security features.

SAMSUNG SDS

Level Set: Hash / Digest

"A cryptographic hash function is a hash function which takes an input (or 'message') and returns a fixed-size string of bytes. The string is called the 'hash value', 'message digest', 'digital fingerprint', 'digest' or 'checksum'."

– Wikipedia

In practical terms: Lets you see if the content you have is what you expected.

Hash / Digest Example

v1.17.3

06ad960

Compare

v1.17.3

k8s-release-robot released this 3 days ago · 2 commits to release-1.17 since this release

See kubernetes-announce@ and CHANGELOG-1.17.md for details.

SHA256 for `kubernetes.tar.gz` :
`9a7c074340cde46c9fd0fa9a500951fa5568081473ba3b0195952d3631fa32da`
SHA512 for `kubernetes.tar.gz` :
`63e54488630e41488f7153583b3c536df766a623c9eb41634e09a113e2ffdaf973c85ddb5d13adc2727fcf262895ce2552507bdeaf2646c00097f4e24f2b9937`

Additional binary downloads are linked in the CHANGELOG-1.17.md.

Assets 3

kubernetes.tar.gz	452 KB
Source code (zip)	
Source code (tar.gz)	

Justin Cappos Pop Quiz: What Do All Of These Organizations Have In Common?

SAMSUNG SDS

Level Set: Provenance

"***Within computer science, informatics uses the term "provenance" to mean the lineage of data***, as per data provenance, with research in the last decade extending the conceptual model of causality and relation to include processes that act on data and agents that are responsible for those processes."

– Wikipedia (emphasis added)

In practical terms: Lets you see if the content you received is from who you expect it to be from.

SAMSUNG SDS

Provenance Example

Protocol	Location
HTTP	https://www.kernel.org/pub/
GIT	https://git.kernel.org/
RSYNC	rsync://rsync.kernel.org/pub/

Latest Stable Kernel: 5.5.3

mainline:	**5.6-rc1**	2020-02-10	[tarball]		[patch]		[view diff]	[browse]	
stable:	**5.5.3**	2020-02-11	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**5.4.19**	2020-02-11	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**4.19.103**	2020-02-11	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**4.14.170**	2020-02-05	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**4.9.213**	2020-02-05	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**4.4.213**	2020-02-05	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**3.16.82**	2020-02-11	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse]	[changelog]
linux-next:	**next-20200214**	2020-02-14						[browse]	


```
-----BEGIN PGP SIGNATURE-----
Comment: This signature is for the .tar version of the archive
Comment: git archive --format tar --prefix=linux-5.5.3/ v5.5.3
Comment: git version 2.25.0

iQIzBAABCAAdFiEEZH8oZUiU471FcZm+ONu9yGCSaT4FAl5CoCcACgkQONu9yGCS
aT4GFhAAjG4QkuJ0NCOh0orMZM4bQEqIkVNrvir33DWiN28rVzOs+DtVtv/gruuL
FelojL008IxTuVdgOKLhOF1rssPoGQLIthPsR/sJZenPCWVMtuNKizeBHgxT5dQ5
XCpAK9bNIy038FVNtt2tR6PcXorMgJCwglIic7/i5cXvSuRXbTJ1gHirWq9uNfr5
rnUX73JGZLNkS13I4/tvi3rRSN6h/74VWkTlRvxmT15+DwcjrIo0vO4AJ2nINfmU
5iurnHYtG6ccBY6DDwHBn4Pmyd3fS7T/fNlXI/F3WxBgZVz8FXpy4wGiOBAjZb9L
0Meix9f+ZSjXJ2xvu/Y+jNPb3SSwlO++aCrlLAsR7YrQSPJoGp8alcAI2FWMKLwY
LkRrwkvK+t9HWUiMx/hAb1j3KAXbv9wNYkxcNP3nERv3ELduMKPmkUvZI16XZ14U
y4tUSC77wjh8TlSkq9M36q7rjlJh+7IfTLoAxSaRkAtuJMXqnhb0nnX5JggzJMRR
aOB8Y+NgFnuMD5V4dnEXwwq6WOzOa0BLUK9le76NiT+JjHLXF6vTlya1jBVJ02OK
ukBqgEsAFbnNTN1yqpiFSW7dpIdIDT79GoFoQ7B5gVu8qrYBoaIUbHfqQKtYj4LL
midD7NIP1F/GIe9p7a4tlbpgnT5OTNuJFOxJyHD6kkmQKJkchUA=
=5coV
-----END PGP SIGNATURE-----
```


SAMSUNG SDS

What Does This Mean to Helm and Containers?

Downloading Helm

Installation and Upgrading

Download Helm 3.1. The common platform binaries are here:

- MacOS amd64 (checksum / aacb6ce8ffa08eebc4e4a570226675f53963c86feb8386d46abf4b8871066c92)
- Linux amd64 (checksum / f0fd9fe2b0e09dc9ed190239fce892a468cbb0a2a8fffb9fe846f893c8fd09de)
- Linux arm (checksum / cb2824c01860196fab8cd6ececd04ff78e9c6606d175e6cd5f41e7d99881795b)
- Linux arm64 (checksum / 1ba32db0600db61d8ace7a3afbaf7b045e16c0aab2a054dd9ec8e02755c07674)
- Linux i386 (checksum / 9bb03099968f16c20298773fe5e466fa66206bf0f125ea656e1722cd86f32439)
- Linux ppc64le (checksum / 6ba4a2a6690c0224ab513971e6c56243c60ffbcc3ba4f68960a693b070bd5f71)
- Linux s390x (checksum / de50a26e7ec79702a073797d66c16555e2c74372e653c2e4141c1a3f3c6d38e1)
- Windows amd64 (checksum / f6a6ee20bd216beb2f0195f083b03d43b8801e885a483011807824bd0915b835)

The Quickstart Guide will get you going from there. For **upgrade instructions** or detailed installation notes, check the install guide. You can also use a script to install on any system with `bash`.

This release was signed with `4614 49C2 5E36 B98E` and can be found at **@mattfarina**'s keybase account. Please use the attached signatures for verifying this release using `gpg`.

```
$ wget https://get.helm.sh/helm-v3.1.0-linux-amd64.tar.gz
--2020-02-14 12:00:34--  https://get.helm.sh/helm-v3.1.0-linux-amd64.tar.gz
Resolving get.helm.sh... 152.195.19.97
Connecting to get.helm.sh|152.195.19.97|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 12267464 (12M) [application/x-tar]
Saving to: ‘helm-v3.1.0-linux-amd64.tar.gz’

helm-v3.1.0-linux-amd64.t 100%[===================================>]  11.70M  11.2MB/s    in 1.0s

2020-02-14 12:00:36 (11.2 MB/s) - ‘helm-v3.1.0-linux-amd64.tar.gz’ saved [12267464/12267464]

$ wget https://get.helm.sh/helm-v3.1.0-linux-amd64.tar.gz.sha256sum
--2020-02-14 12:00:41--  https://get.helm.sh/helm-v3.1.0-linux-amd64.tar.gz.sha256sum
Resolving get.helm.sh... 152.195.19.97
Connecting to get.helm.sh|152.195.19.97|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 97 [application/octet-stream]
Saving to: ‘helm-v3.1.0-linux-amd64.tar.gz.sha256sum’

helm-v3.1.0-linux-amd64.t 100%[===================================>]      97  --.-KB/s    in 0s

2020-02-14 12:00:41 (1.36 MB/s) - ‘helm-v3.1.0-linux-amd64.tar.gz.sha256sum’ saved [97/97]

$ shasum -a 256 -c helm-v3.1.0-linux-amd64.tar.gz.sha256sum
helm-v3.1.0-linux-amd64.tar.gz: OK
```

Failure Example

```
$ shasum -a 256 -c helm-v3.1.0-linux-amd64.tar.gz.sha256sum
helm-v3.1.0-linux-amd64.tar.gz: FAILED
shasum: WARNING: 1 computed checksum did NOT match

$ echo $?
1
```

SAMSUNG SDS

Downloading Helm Signatures

Assets 26

helm-v3.1.0-darwin-amd64.tar.gz.asc	833 Bytes
helm-v3.1.0-darwin-amd64.tar.gz.sha256.asc	833 Bytes
helm-v3.1.0-darwin-amd64.tar.gz.sha256sum.asc	833 Bytes
helm-v3.1.0-linux-386.tar.gz.asc	833 Bytes
helm-v3.1.0-linux-386.tar.gz.sha256.asc	833 Bytes
helm-v3.1.0-linux-386.tar.gz.sha256sum.asc	833 Bytes
helm-v3.1.0-linux-amd64.tar.gz.asc	833 Bytes
helm-v3.1.0-linux-amd64.tar.gz.sha256.asc	833 Bytes
helm-v3.1.0-linux-amd64.tar.gz.sha256sum.asc	833 Bytes
helm-v3.1.0-linux-arm.tar.gz.asc	833 Bytes
helm-v3.1.0-linux-arm.tar.gz.sha256.asc	833 Bytes
helm-v3.1.0-linux-arm.tar.gz.sha256sum.asc	833 Bytes
helm-v3.1.0-linux-arm64.tar.gz.asc	833 Bytes
helm-v3.1.0-linux-arm64.tar.gz.sha256.asc	833 Bytes
helm-v3.1.0-linux-arm64.tar.gz.sha256sum.asc	833 Bytes
helm-v3.1.0-linux-ppc64le.tar.gz.asc	833 Bytes
helm-v3.1.0-linux-ppc64le.tar.gz.sha256.asc	833 Bytes
helm-v3.1.0-linux-ppc64le.tar.gz.sha256sum.asc	833 Bytes
helm-v3.1.0-linux-s390x.tar.gz.asc	833 Bytes
helm-v3.1.0-linux-s390x.tar.gz.sha256.asc	833 Bytes
helm-v3.1.0-linux-s390x.tar.gz.sha256sum.asc	833 Bytes
helm-v3.1.0-windows-amd64.zip.asc	833 Bytes
helm-v3.1.0-windows-amd64.zip.sha256.asc	833 Bytes
helm-v3.1.0-windows-amd64.zip.sha256sum.asc	833 Bytes
Source code (zip)	
Source code (tar.gz)	

```
$ curl https://keybase.io/mattfarina/pgp_keys.asc | gpg --import

$ curl -OL https://get.helm.sh/helm-v3.1.0-linux-amd64.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 11.6M  100 11.6M    0     0  8373k      0  0:00:01  0:00:01 --:--:-- 8377k

$ curl -OL https://github.com/helm/helm/releases/download/v3.1.0/helm-v3.1.0-linux-amd64.tar.gz.asc
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   624  100   624    0     0   1585      0 --:--:-- --:--:-- --:--:--  1583
100   833  100   833    0     0    698      0  0:00:01  0:00:01 --:--:--  7642

$ gpg --verify helm-v3.1.0-linux-amd64.tar.gz.asc
gpg: assuming signed data in 'helm-v3.1.0-linux-amd64.tar.gz'
gpg: Signature made Thu Feb 13 11:40:29 2020 EST
gpg:                using RSA key 711F28D510E1E0BCBD5F6BFE9436E80BFBA46909
gpg: Good signature from "Matthew Farina <matt@mattfarina.com>" [ultimate]
```

SAMSUNG SDS

```
$ gpg --verify helm-v3.1.0-linux-amd64.tar.gz.asc
gpg: assuming signed data in 'helm-v3.1.0-linux-amd64.tar.gz'
gpg: Signature made Thu Feb 13 11:40:29 2020 EST
gpg:                using RSA key 711F28D510E1E0BCBD5F6BFE9436E80BFBA46909
gpg: BAD signature from "Matthew Farina <matt@mattfarina.com>" [ultimate]

$ echo $?
1
```

What happens if the verification fails? It tells you and you get a non-0 exit code.

SAMSUNG SDS

```
$ curl https://keybase.io/mattfarina/pgp_keys.asc | gpg --import

$ curl https://mattfarina.com/pgp_key.asc | gpg --import

$ curl https://github.com/helm/helm/blob/master/KEYS | gpg --import

$ gpg --locate-keys matt@mattfarina.com
```

Four ways to get a public key

```
$ gpg --fingerprint matt@mattfarina.com
pub   rsa4096/0x461449C25E36B98E 2017-11-10 [SC]
      Key fingerprint = 672C 657B E06B 4B30 969C  4A57 4614 49C2 5E36 B98E
uid                 [ultimate] Matthew Farina <matt@mattfarina.com>
sub   rsa4096/0xCCCE67689DF05738 2017-11-10 [E]
sub   rsa4096/0x9436E80BFBA46909 2017-11-10 [S] [expires: 2022-11-09]
```

Every Key Has A Unique Fingerprint

When you have a fingerprint you can get and check a key against even if you get if from many locations. The fingerprint for the person signing each release is listed in the release notes.

SAMSUNG SDS

Charts

Charts and Registries

```
$ helm package foo
Successfully packaged chart and saved it to: /Users/mfarina/Code/preso/foo-0.1.0.tgz

$ helm repo index .

$ cat index.yaml
apiVersion: v1
entries:
  foo:
  - apiVersion: v2
    appVersion: 1.16.0
    created: "2020-02-18T13:07:23.754331-05:00"
    description: A Helm chart for Kubernetes
    digest: 9f17b96c1f6519c830e91d454cbe4afc845826133fb3a26dde2aea3d4901d62a
    name: foo
    type: application
    urls:
    - foo-0.1.0.tgz
    version: 0.1.0
generated: "2020-02-18T13:07:23.748982-05:00"
```

SAMSUNG SDS

Charts and Registries

```
$ helm pull demo-repo/foo

$ ls
foo-0.1.0.tgz
```

SAMSUNG SDS

Signing Charts

```
$ helm package --sign --key matt@mattfarina.com --keyring /path/to/keyring.pgp foo
```

```
$ helm plugin install https://github.com/technosophos/helm-gpg
Installed plugin: gpg

$ helm package foo
Successfully packaged chart and saved it to:
/Users/mfarina/Code/preso/foo-0.1.0.tgz

$ helm gpg sign foo-0.1.0.tgz
Signing foo-0.1.0.tgz
```

Two Different Methods

```
$ ls
foo                 foo-0.1.0.tgz     foo-0.1.0.tgz.prov
```

Provenance Files

```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA512

apiVersion: v2
appVersion: 1.16.0
description: A Helm chart for Kubernetes
name: foo
type: application
version: 0.1.0

...
files:
  foo-0.1.0.tgz: sha256:9f17b96c1f6519c830e91d454cbe4afc845826133fb3a26dde2aea3d4901d62a
-----BEGIN PGP SIGNATURE-----

iQIzBAEBCgAdFiEEcR8o1RDh4Ly9X2v+lDboC/ukaQkFAl5MRNUACgkQlDboC/uk
aQmGCxAAoSGzz33mBQ6P9QGJPnWBiBkRfuImvH8cwqahd6StaMJ70zWwm+h/uIlt
CKzeWbsyNl/OeR9lSKne+qXskIQ0qaM8IoyX29x4NUK/4QqjQ/WDJ3TKvjV2eTzj
J2GjlboC2Y5tPIjFS0Evi52ZZoC4Wy6p/dBHGTmrRbqBRpuf+4PeyeTJJkmDNwXA
0VEVeYib+KMWiqpX0cmWLyUoovAzx3QNVrPvNavUVh8ycjC5dmC143i12KF5nuPR
mAa7n4MPMjq58hweTmXPtl+uyhl3T+xGeBhH8vBBKXVca1Ygh0hZe4AWB+eFGH60
Ydc+IPVEThOf8PGDRJlJTNwuGXihlLqfJSWDzXg7sh4tTkhzKbwKygtFMQjEfu0s
lVadsxS9e8HP2JxLNEBbKtPJQsSnDG71vcDuGoZrH+u2EZAl/qKvKO122rxKgRcH
ADnr6KKZ8lXED18AUvoZz7XE4OzSNFcDOAAIGtahYCUvDA+0NOZQXSeV3m3jT5qq
+zD4MHfEORv4mG0rcTaZbbsA6y8eriIS2DNwjubCOd/Z6XIJVtDNYbX4WQYUsaLL
+iCiX+aqeif8MXgVXcM47Hz9XkqK0yPW/1DLfyrZg3r7dSGRejPI+VfyF4Aj6eBr
vT/fmGKUfGJz+jJJG1jDgsqS8eI++nof7C4LlS75m1BHrlV7uCc=
=F1Tf
-----END PGP SIGNATURE-----
```

Metadata from Chart.yaml

Hash of chart archive

PGP Signature of archive

SAMSUNG SDS

Verifying Charts

```
$ helm pull --prov demo-repo/foo

$ ls
foo-0.1.0.tgz    foo-0.1.0.tgz.prov

$ helm gpg verify foo-0.1.0.tgz
gpg: Signature made Tue Feb 18 15:35:42 2020 EST
gpg:                using RSA key 711F28D510E1E0BCBD5F6BFE9436E80BFBA46909
gpg: Good signature from "Matthew Farina <matt@mattfarina.com>" [ultimate]
plugin: Chart SHA verified.
sha256:f8a4e6e3ebf8a3fac95396b698f5517d43e0f5275bac3c484b6f38428b90b578
```

```
$ gpg --output keyring.pgp --export matt@mattfarina.com
```

```
$ helm pull --verify --keyring /path/to/keyring.pgp demo-repo/foo
```

Security in Charts

Resource Quotas

kubernetes.io/docs/concepts/policy/resource-quotas/

kubernetes

Documentation Blog Partners Community Case Studies English v1.17

Compute Resource Quota

You can limit the total sum of compute resources that can be requested in a given namespace.

The following resource types are supported:

Resource Name	Description
`limits.cpu`	Across all pods in a non-terminal state, the sum of CPU limits cannot exceed this value.
`limits.memory`	Across all pods in a non-terminal state, the sum of memory limits cannot exceed this value.
`requests.cpu`	Across all pods in a non-terminal state, the sum of CPU requests cannot exceed this value.
`requests.memory`	Across all pods in a non-terminal state, the sum of memory requests cannot exceed this value.

Resource Quota For Extended Resources

In addition to the resources mentioned above, in release 1.10, quota support for extended resources is added.

As overcommit is not allowed for extended resources, it makes no sense to specify both `requests` and `limits` for the same extended resource in a quota. So for extended resources, only quota items with prefix `requests.` is allowed for now.

Take the GPU resource as an example, if the resource name is `nvidia.com/gpu`, and you want to limit the total number of GPUs requested in a namespace to 4, you can define a quota as follows:

- `requests.nvidia.com/gpu: 4`

Limit resource usage to normal needs

SAMSUNG SDS

Secrets

kubernetes.io/docs/tasks/inject-data-application/distribute-credentials

kubernetes

Documentation Blog Partners Community Case Studies English v1.17

Tasks

HOME GETTING STARTED CONCEPTS TASKS TUTORIALS REFERENCE CONTRIBUTE

Search

Tasks

- Install Tools
- Administer a Cluster
- Configure Pods and Containers
- Manage Kubernetes Objects
- Inject Data Into Applications
  - Define a Command and Arguments for a Container
  - Define Environment Variables for a Container
  - Expose Pod Information to Containers Through Environment Variables
  - Expose Pod Information to Containers Through Files
  - Distribute Credentials Securely Using Secrets
  - Inject Information into Pods Using a PodPreset
- Run Applications
- Run Jobs
- Access Applications in a Cluster
- Monitoring, Logging, and Debugging
- Extend Kubernetes
- TLS

Distribute Credentials Securely Using Secrets

This page shows how to securely inject sensitive data, such as passwords and encryption keys, into Pods.

- Before you begin
- Convert your secret data to a base-64 representation
- Create a Secret
- Create a Pod that has access to the secret data through a Volume
- Define container environment variables using Secret data
- Configure all key-value pairs in a Secret as container environment variables
- What's next

Before you begin

You need to have a Kubernetes cluster, and the kubectl command-line tool must be configured to communicate with your cluster. If you do not already have a cluster, you can create one by using Minikube, or you can use one of these Kubernetes playgrounds:

- Katacoda
- Play with Kubernetes

Convert your secret data to a base-64 representation

Secrets not ConfigMaps

SAMSUNG SDS

Secret Types

```
$ cat foo-test-secret.yaml
apiVersion: v1
kind: Secret
metadata:
  name: foo-test-secret
type: com.example.mine
stringData:
  foo: bar
  baz: qux
```

Secrets can have a type

```
$kubectl get secrets --field-selector "type=com.example.mine"
NAME              TYPE                DATA   AGE
foo-test-secret   com.example.mine    2      40s
```

You can list secrets by type

SAMSUNG SDS

Encrypting Secrets At Rest

kubernetes.io/docs/tasks/administer-cluster/encrypt-data/#providers

kubernetes

Documentation Blog Partners Community Case Studies English v1.17

Tasks

HOME GETTING STARTED CONCEPTS TASKS TUTORIALS REFERENCE CONTRIBUTE

Search

Tasks

- Install Tools
- Administer a Cluster
  - Administration with kubeadm
  - Manage Memory, CPU, and API Resources
  - Install a Network Policy Provider
  - Access Clusters Using the Kubernetes API
  - Access Services Running on Clusters
  - Advertise Extended Resources for a Node
  - Autoscale the DNS Service in a Cluster
  - Change the default StorageClass
  - Change the Reclaim Policy of a PersistentVolume
  - Cluster Management
  - Configure Multiple Schedulers
  - Configure Out of Resource Handling
  - Configure Quotas for API Objects
  - Control CPU Management Policies on the Node
  - Control Topology Management Policies on a node
  - Customizing DNS Service
  - Debugging DNS Resolution

Encrypting Secret Data at Rest

This page shows how to enable and configure encryption of secret data at rest.

- Before you begin
- Configuration and determining whether encryption at rest is already enabled
- Understanding the encryption at rest configuration.
- Encrypting your data
- Verifying that data is encrypted
- Ensure all secrets are encrypted
- Rotating a decryption key
- Decrypting all data

Before you begin

- You need to have a Kubernetes cluster, and the kubectl command-line tool must be configured to communicate with your cluster. If you do not already have a cluster, you can create one by using Minikube, or you can use one of these Kubernetes playgrounds:
- Katacoda
- Play with Kubernetes

Your Kubernetes server must be at or later than version 1.13. To check the version, enter `kubectl version`.

Cluster operators can setup clusters to encrypt secrets at rest. Real encryption. Not base64 in etcd.

SAMSUNG SDS

RBAC

kubernetes.io/docs/reference/access-authn-authz/rbac/

kubernetes

Documentation Blog Partners Community Case Studies English v1.17

- API Reference
- Setup tools reference
- Command line tools reference
- kubectl CLI

Tools

API Overview

The RBAC API declares four top-level types which will be covered in this section. Users can interact with these resources as they would with any other API resource (via `kubectl`, API calls, etc.). For instance, `kubectl apply -f (resource).yml` can be used with any of these examples, though readers who wish to follow along should review the section on bootstrapping first.

Role and ClusterRole

In the RBAC API, a role contains rules that represent a set of permissions. Permissions are purely additive (there are no "deny" rules). A role can be defined within a namespace with a `Role`, or cluster-wide with a `ClusterRole`.

A `Role` can only be used to grant access to resources within a single namespace. Here's an example `Role` in the "default" namespace that can be used to grant read access to pods:

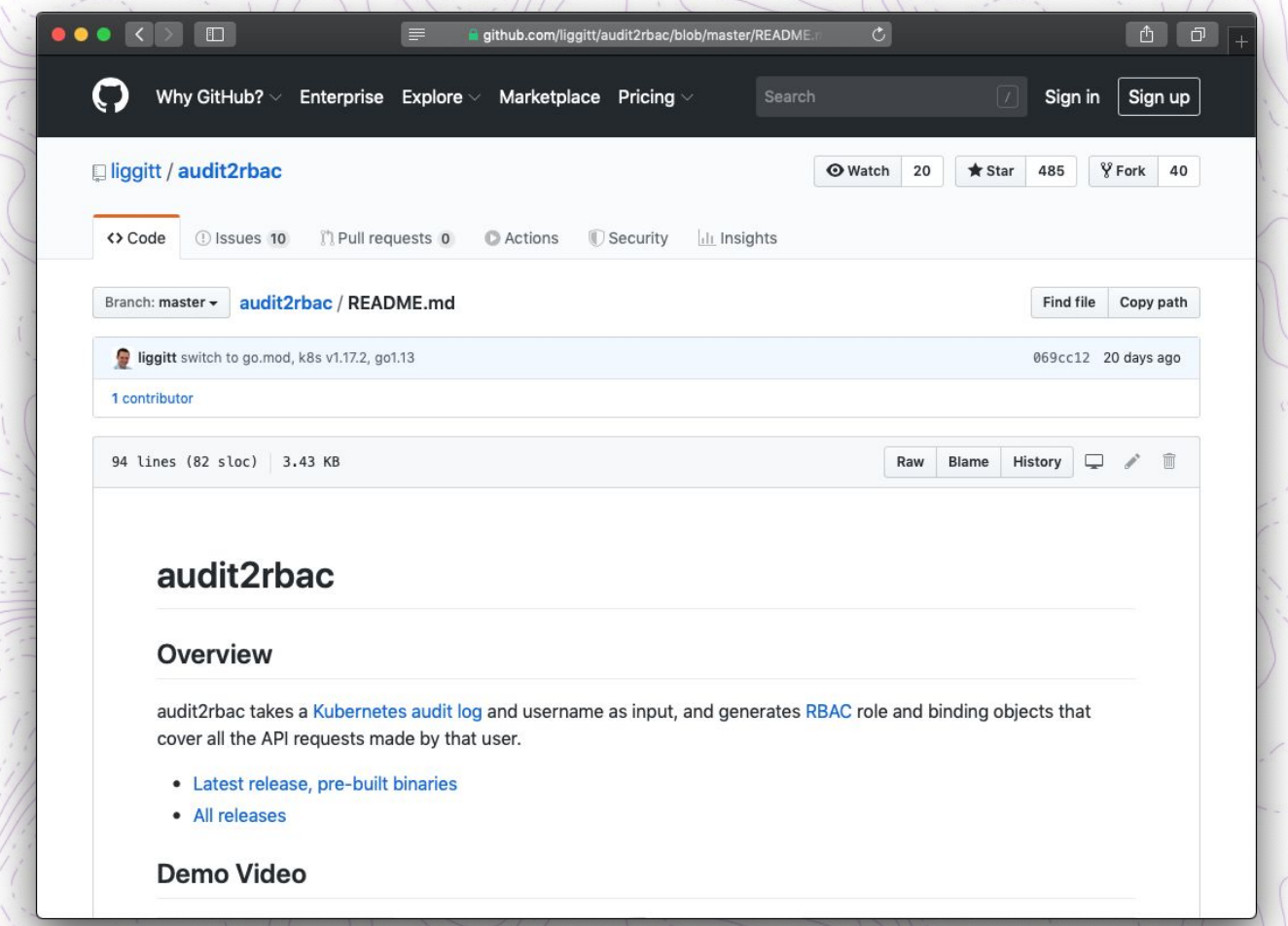
```
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  namespace: default
  name: pod-reader
rules:
- apiGroups: [""] # "" indicates the core API group
  resources: ["pods"]
  verbs: ["get", "watch", "list"]
```

Role, ClusterRole, RoleBinding,

ClusterRoleBinding

SAMSUNG SDS

audit2rbac



https://github.com/liggitt/audit2rbac

Takes a Kubernetes audit log and builds RBAC resources for you

SAMSUNG SDS

Service Accounts

kubernetes.io/docs/tasks/configure-pod-container/configure-se

kubernetes

Documentation Blog Partners Community Case Studies English v1.17

Tasks

HOME GETTING STARTED CONCEPTS TASKS TUTORIALS REFERENCE CONTRIBUTE

Search

Tasks

- Install Tools
- Administer a Cluster
- Configure Pods and Containers
  - Assign Memory Resources to Containers and Pods
  - Assign CPU Resources to Containers and Pods
  - Configure GMSA for Windows Pods and containers
  - Configure RunAsUserName for Windows pods and containers
  - Configure Quality of Service for Pods
  - Assign Extended Resources to a Container
  - Configure a Pod to Use a Volume for Storage
  - Configure a Pod to Use a PersistentVolume for Storage

Configure Service Accounts for Pods

A service account provides an identity for processes that run in a Pod.

> **Note:** This document is a user introduction to Service Accounts and describes how service accounts behave in a cluster set up as recommended by the Kubernetes project. Your cluster administrator may have customized the behavior in your cluster, in which case this documentation may not apply.

When you (a human) access the cluster (for example, using `kubectl`), you are authenticated by the apiserver as a particular User Account (currently this is usually `admin`, unless your cluster administrator has customized your cluster). Processes in containers inside pods can also contact the apiserver. When they do, they are authenticated as a particular Service Account (for example, `default`).

- Before you begin
- Use the Default Service Account to access the API server.
- Use Multiple Service Accounts.
- eate a service account API token.

Open "https://kubernetes.io/docs/tasks/configure-pod-container/assign-memory-resource/" in a new tab

Custom application specific service accounts

SAMSUNG SDS

Pod Security Policies

kubernetes.io/docs/concepts/policy/pod-security-policy/#run-ar

kubernetes

Documentation Blog Partners Community Case Studies English v1.17

Concepts

HOME GETTING STARTED CONCEPTS TASKS TUTORIALS REFERENCE CONTRIBUTE

Search

Concepts

- Overview
- Cluster Architecture
- Containers
- Workloads
- Services, Load Balancing, and Networking
- Storage
- Configuration
- Security
- Policies
  - Limit Ranges
  - Resource Quotas
  - Pod Security Policies
- Scheduling
- Cluster Administration
- Extending Kubernetes

Pod Security Policies

FEATURE STATE: `Kubernetes v1.17` beta

Pod Security Policies enable fine-grained authorization of pod creation and updates.

What is a Pod Security Policy?

A *Pod Security Policy* is a cluster-level resource that controls security sensitive aspects of the pod specification. The PodSecurityPolicy objects define a set of conditions that a pod must run with in order to be accepted into the system, as well as defaults for the related fields. They allow an administrator to control the following:

Control Aspect	Field Names
Running of privileged containers	`privileged`
Usage of host namespaces	`hostPID`, `hostIPC`
Usage of host networking and ports	`hostNetwork`, `hostPorts`

Control features pods can access

SAMSUNG SDS

Image Security

Vulnerabilities In Container Images

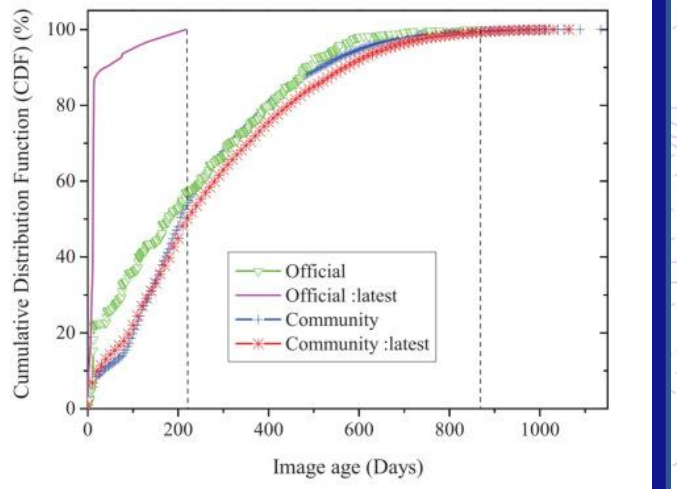


Figure 6: Cumulative distribution function (CDF) of percentage distribution of the age of images at the time of analysis.

From paper published in ACM

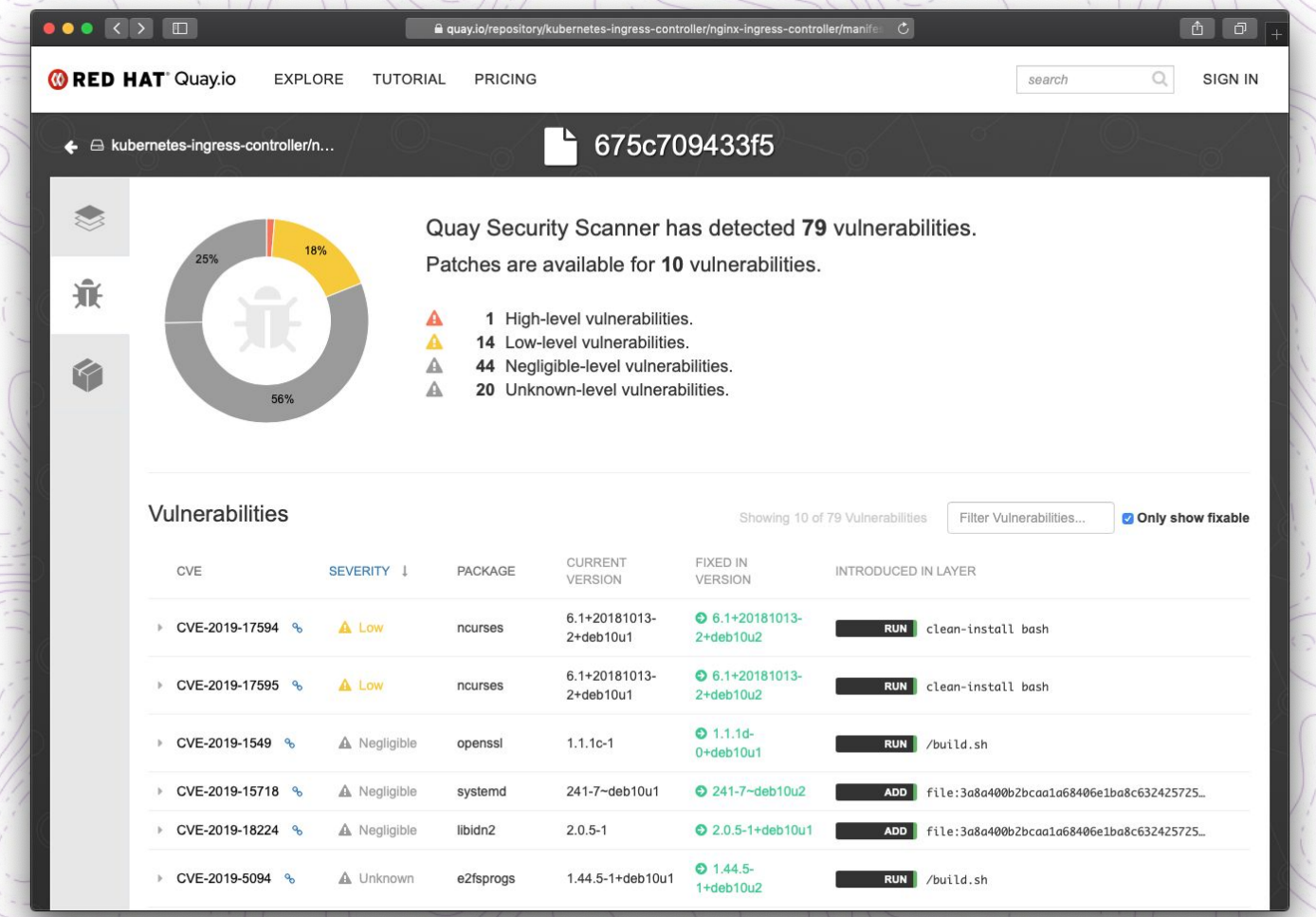
About Docker Hub circa 2017

https://blog.acolyer.org/2017/04/03/a-study-of-security-vulnerabilities-on-docker-hub/

Scan Your Container Images And Fix CVEs

SAMSUNG SDS

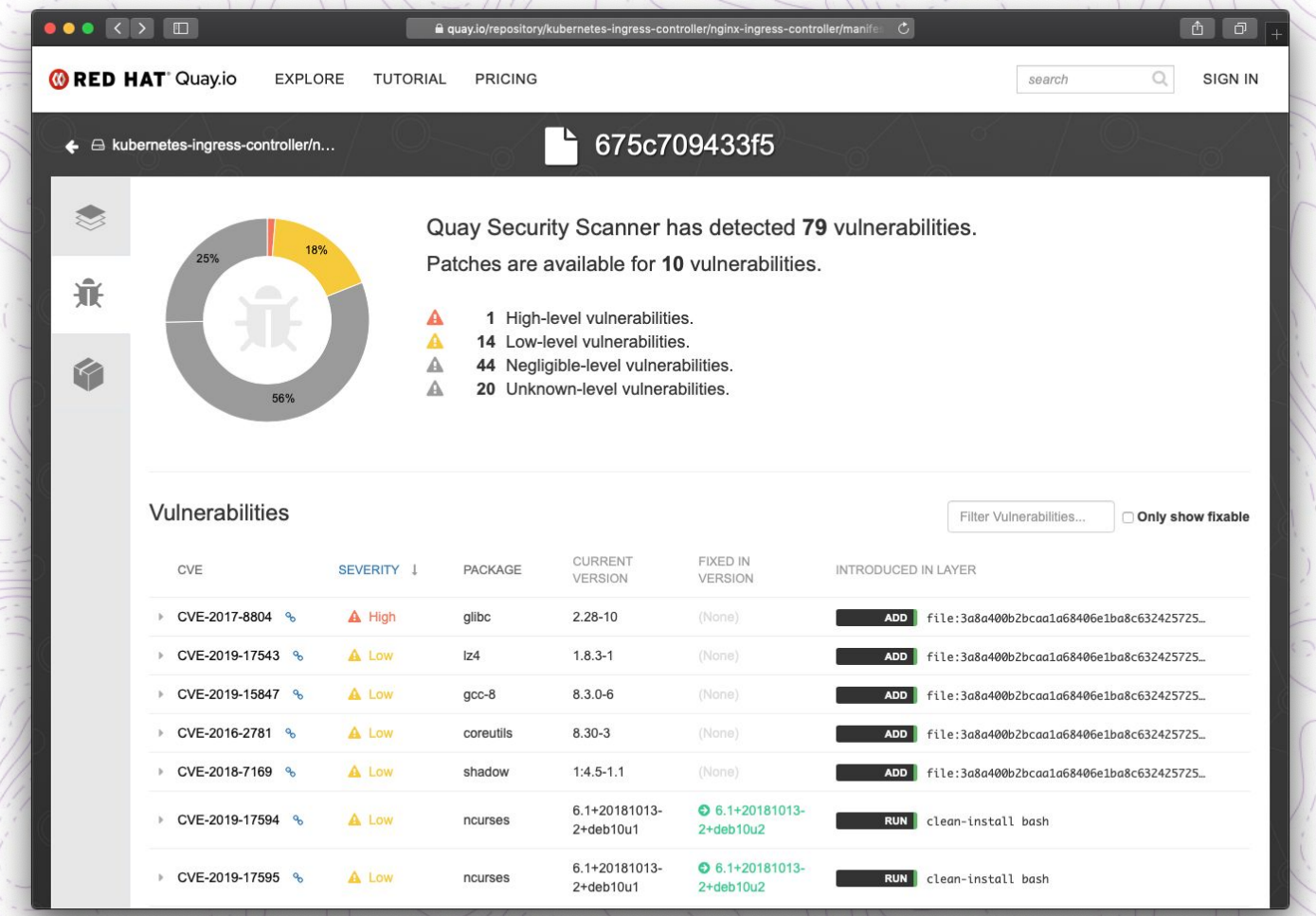
Quay (one of many options)



An old version with vulnerabilities found

SAMSUNG SDS

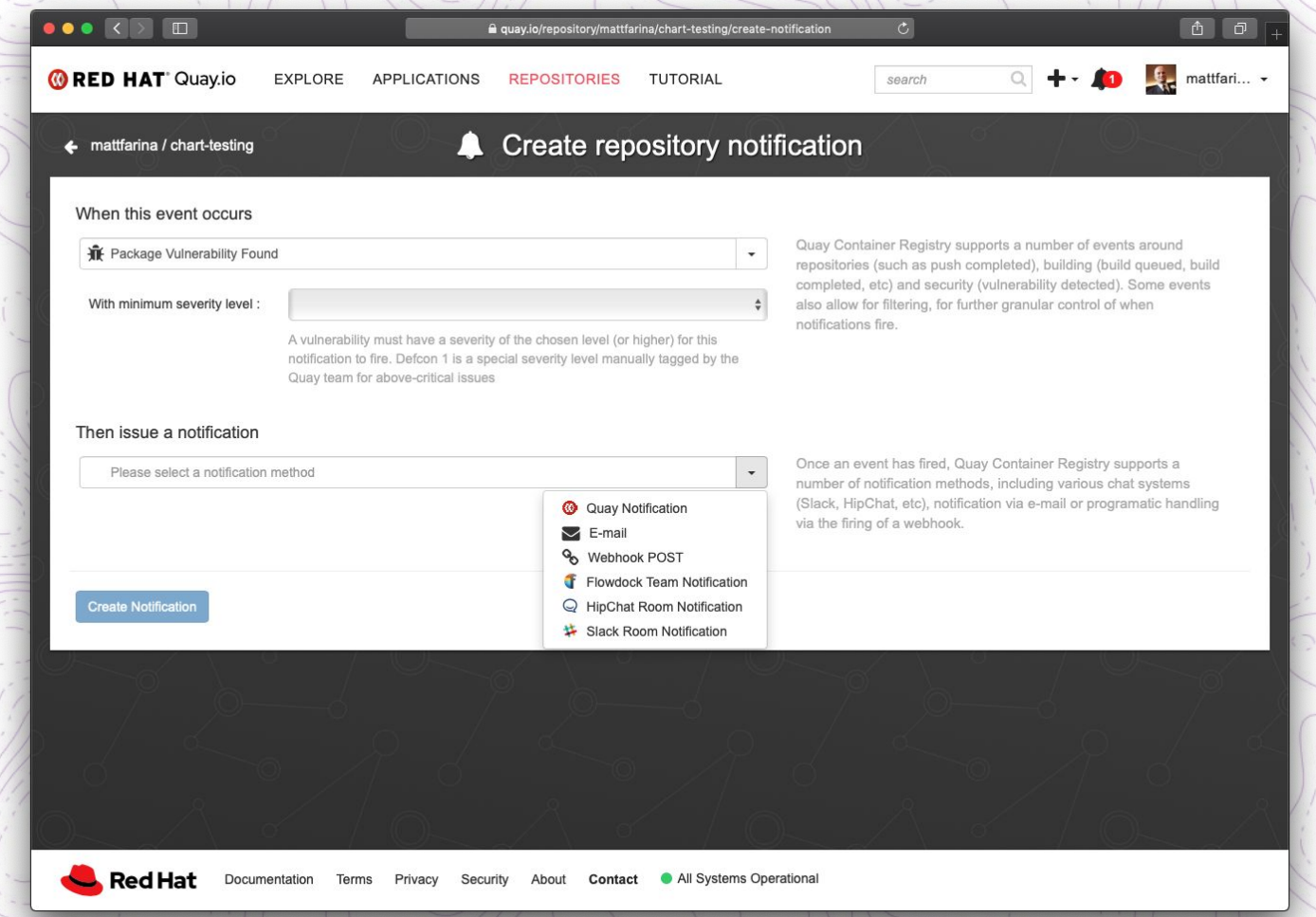
Quay (the adventure continues)



Many unfixable issues! Turns out there were fixes. Latest image is CVE free. Fixes may not be obvious.

SAMSUNG SDS

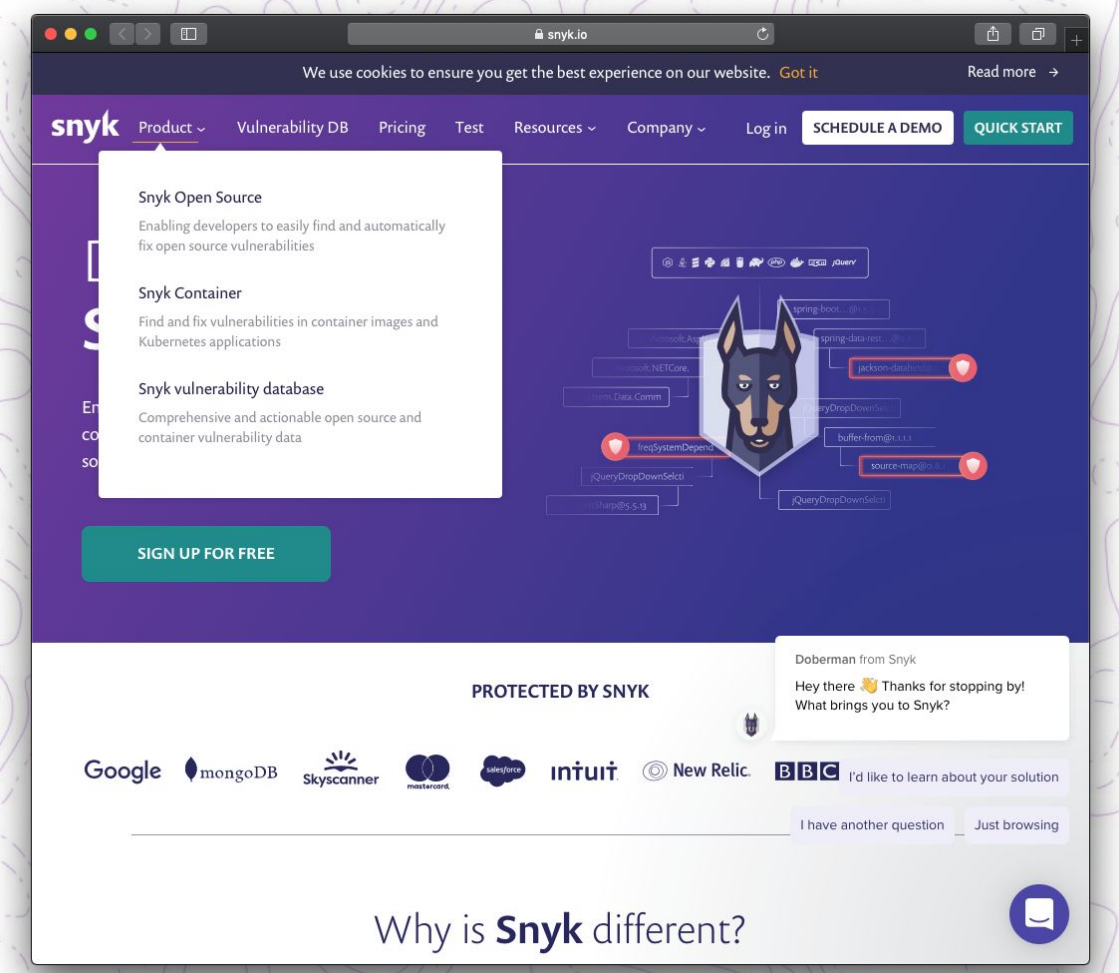
Quay (event notifications!)



Quay provides the ability to notify you if one of YOUR images has a vulnerability found. Does not tell you about others images.

SAMSUNG SDS

Snyk



Snyk provides the ability to scan container images in your development workflow (CI)

SAMSUNG SDS

Snyk

github.com/marketplace/actions/snyk

Search or jump to... / Pull requests Issues Marketplace Explore

Marketplace / Actions / Snyk

GitHub Action

Snyk

0.1.1 Pre-release

Use latest version

Snyk GitHub Actions

Generate Snyk GitHub Actions passing

A set of GitHub Action for using Snyk to check for vulnerabilities in your GitHub projects. A different action is required depending on which language or build tool you are using. We currently support:

- CocoaPods
- DotNet
- Golang
- Gradle
- Maven
- Node
- PHP
- Python
- Ruby
- Scala
- Docker

Here's an example of using one of the Actions, in this case to test a Node.js project:

```
name: Example workflow using Snyk
on: push
```

Verified creator

GitHub has verified that this action was created by **snyk**.

Learn more about verified Actions.

Stars

Star 18

Contributors

Categories

Security

Open Source management

Links

snyk/actions

Open issues 1

Pull requests 0

Report abuse

github.com/snyk/actions/tree/master/docker

Search or jump to... / Pull requests Issues Marketplace Explore

snyk / actions

Watch 32 Star 18 Fork 3

Code Issues 1 Pull requests 0 Actions Projects 0 Security Insights

Branch: master actions / docker / Create new file Upload files Find file History

garethr Added Docker action Latest commit 69fc553 on Oct 24, 2019

..

README.md	Added Docker action	4 months ago
action.yml	Added Docker action	4 months ago

README.md

Snyk Docker Action

A GitHub Action for using Snyk to check for vulnerabilities in your Docker images.

You can use the Action as follows:

```
name: Example workflow for Docker using Snyk
on: push
jobs:
  security:
    runs-on: ubuntu-latest
    steps:
    - name: Run Snyk to check Docker image for vulnerabilities
      uses: snyk/actions/docker@master
      env:
        SNYK_TOKEN: ${{ secrets.SNYK_TOKEN }}
      with:
        image: your/image-to-test
```

The Snyk Docker Action has properties which are passed to the underlying image. These are passed to the action using

SAMSUNG SDS

Hayley…

CNCF’s investment in security

CNCF’s investment in security

1. Kubernetes Security Audit

CNCF’s investment in security

1. Kubernetes Security Audit
2. Helm Security Audit

CNCF’s investment in security

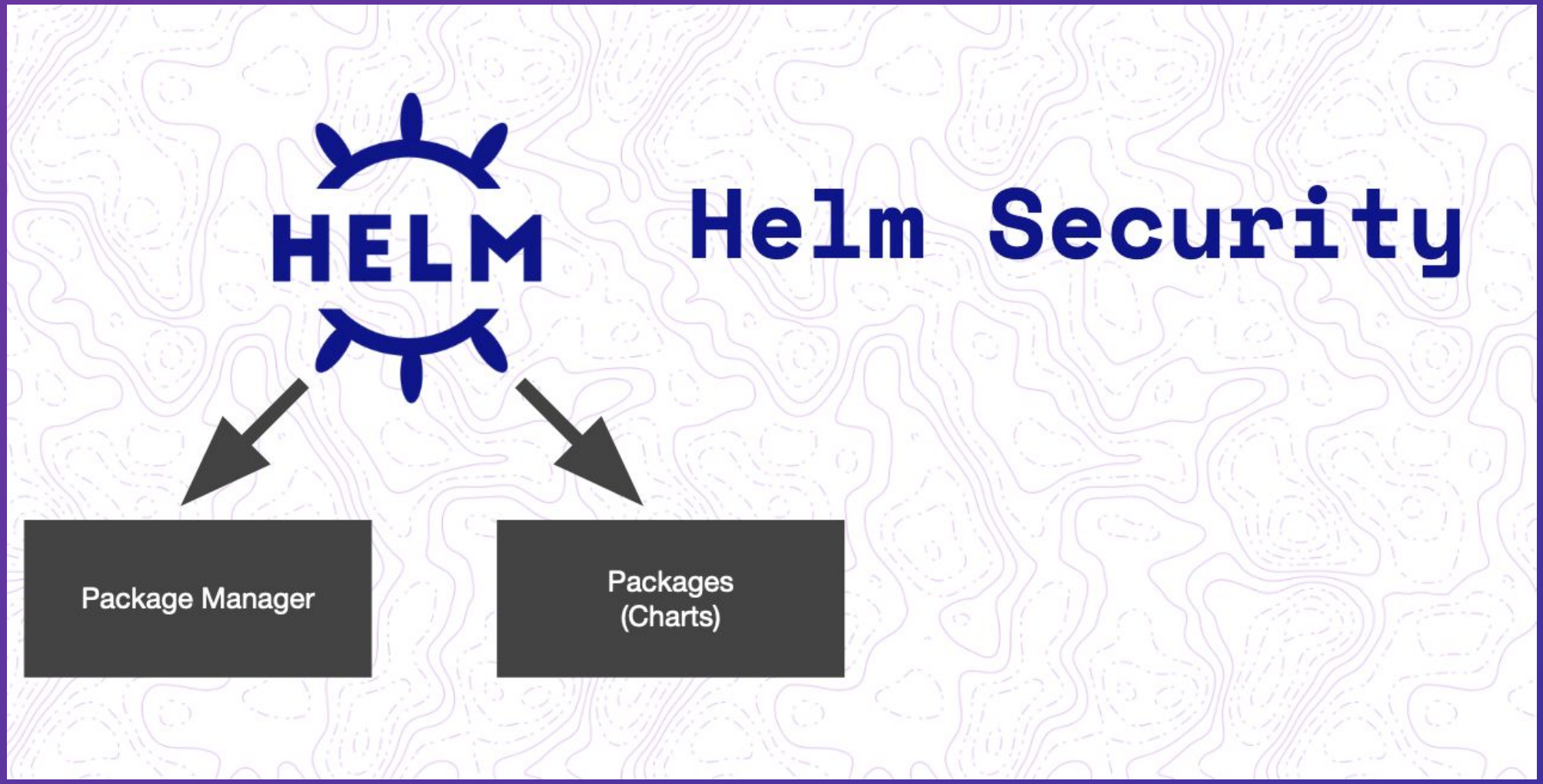
1. Kubernetes Security Audit
2. Helm Security Audit
3. Graduation Requirements

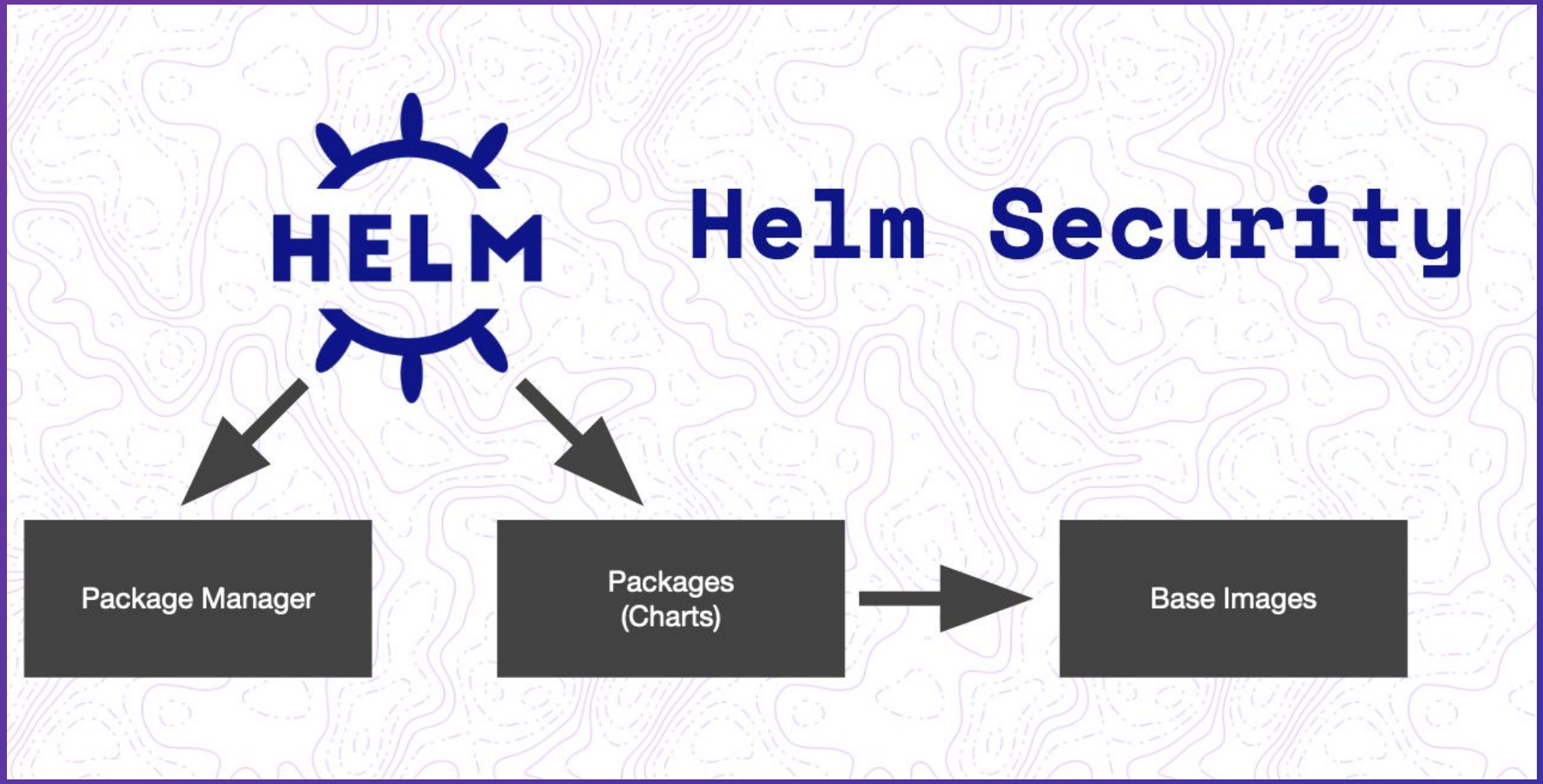
To Review:
Helm Is The Package Manager For Kubernetes

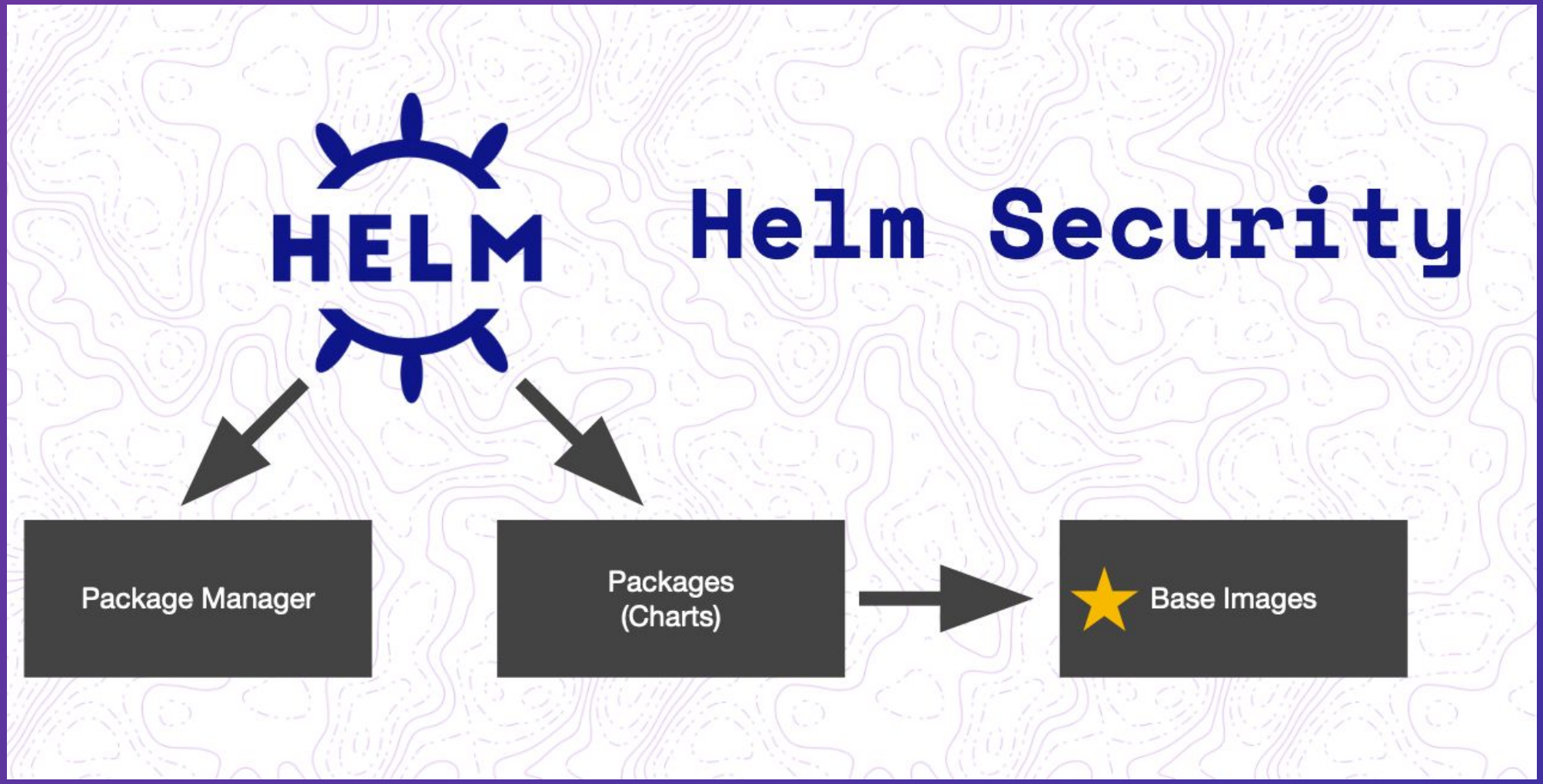
≈

Helm Security


HELM
Helm Security
Package Manager


HELM
Helm Security
Package Manager
Packages
(Charts)


HELM
Helm Security
Package Manager
Packages
(Charts)
Base Images


HELM
Helm Security
Package Manager
Packages
(Charts)
Base Images

snyk.io/helm-report

How was the report put together?

helm / charts

Watch 372 | Star 12.1k | Fork 13k

Code | Issues 274 | Pull requests 238 | Actions | Security | Insights

Curated applications for Kubernetes

kubernetes charts helm

12,031 commits | 6 branches | 0 packages | 0 releases | 2,914 contributors | Apache-2.0

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

czuares Use tpl for labels/annotations (#20644) ··· Latest commit 6ad9947 1 hour ago

.circleci	[ci] Upgrade to chart-testing v2.4.0 (#18538)	4 months ago
.github	Chore: Update github template to be more specific about naming conven...	4 months ago
incubator	[incubator/solr] Bug/fix solr requirements (#20220)	5 hours ago
stable	Use tpl for labels/annotations (#20644)	1 hour ago
test	[ci] Upgrade to chart-testing v2.4.0 (#18538)	4 months ago

How was the report put together?

helm / charts

Watch 372 | Star 12.1k | Fork 13k

Code | Issues 274 | Pull requests 238 | Actions | Security | Insights

Curated applications for Kubernetes

kubernetes | charts | helm

12,031 commits | 6 branches | 0 packages | 0 releases | 2,914 contributors | Apache-2.0

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

czuares Use tpl for labels/annotations (#20644) ··· Latest commit 6ad9947 1 hour ago

.circleci	[ci] Upgrade to chart-testing v2.4.0 (#18538)	4 months ago
.github	Chore: Update github template to be more specific about naming conven...	4 months ago
incubator	(#20220)	5 hours ago
stable	Use tpl for labels/annotations (#20644)	1 hour ago
test	[ci] Upgrade to chart-testing v2.4.0 (#18538)	4 months ago

How was the report put together?

helm / charts

Watch 372 | Star 12.1k | Fork 13k

Code | Issues 274 | Pull requests 238 | Actions | Security | Insights

Curated applications for Kubernetes

kubernetes | charts | helm

12,031 commits | 6 branches | 0 packages | 0 releases | 2,914 contributors | Apache-2.0

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

czuares Use tpl for labels/annotations (#20644) ··· Latest commit 6ad9947 1 hour ago

.circleci	[ci] Upgrade to chart-testing v2.4.0 (#18538)	4 months ago
.github	Chore: Update github template to be more specific about naming conven...	4 months ago
incubator	[incubator/solr] Bug/fix solr requirements (#20220)	5 hours ago
stable		1 hour ago
test	[ci] Upgrade to chart-testing v2.4.0 (#18538)	4 months ago

How was the report put together?

helm / charts

Watch 372 | Star 12.1k | Fork 13k

Code | Issues 274 | Pull requests 238 | Actions | Security | Insights

Branch: master | charts / stable / | Create new file | Upload files | Find file | History

czuares Use tpl for labels/annotations (#20644) ... Latest commit 6ad9947 1 hour ago

..		
acs-engine-autoscaler	deprecate acs-engine autoscaler as it is no longer maintained (#9867)	15 months ago
aerospike	Fixed imagePullSecrets (#19705)	2 months ago
airflow	[stable/airflow] fix postgres database config name (#20828)	15 hours ago
ambassador	Add deprecation warning to helm chart (#20588)	11 days ago
anchore-engine	[stable/anchore-engine] added default admin password helper (#20524)	7 days ago
apm-server	[stable/apm-server] Use a non-root user (#18366)	4 months ago
ark	[stable/velero] Upgrade Ark 0.10.x to Velero 0.11 (#12031)	10 months ago
artifactory-ha	Deprecate JFrog charts (moved to https://github.com/jfrog/charts) (#7627	2 years ago
artifactory	Deprecate JFrog charts (moved to https://github.com/jfrog/charts) (#7627	2 years ago

How was the report put together?

snyk-labs / helm-snyk

Watch 3 | Unstar 17 | Fork 2

Code | Issues 1 | Pull requests 1 | Actions | Projects 0 | Wiki | Security | Insights

Check images in your charts for vulnerabilities https://snyk.io

snyk helm-plugin

97 commits | 7 branches | 0 packages | 11 releases | 4 contributors | View license

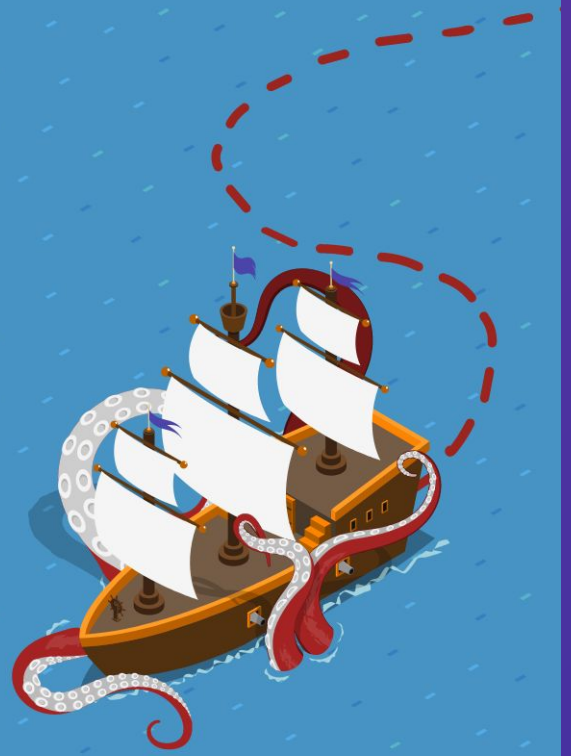
Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

maxjeffos Merge pull request #34 from snyk-labs/chore/add-ci-badge ... Latest commit 36b363f on Dec 30, 2019

.circleci	chore: add nodejs-lib-release to the version_check job	3 months ago
.github	chore: Include bizdev-engineering team to codeowners file	4 months ago
scripts	fix: removed debug message from run script	3 months ago
src	fix: Handle error for `helm template` when it's missing dependencies	3 months ago
.eslintrc.js	feat: initial version	4 months ago

Uncharted territories:

The untold tale of Helm Chart security



powered by snyk

TL;DR

Helm Chart

- 277 stable Helm Charts
- 68% of stable Helm Charts contain an image with a high severity vulnerability

Images

- 416 images used across stable Helm Charts
- 6 images account for nearly half of all vulnerable paths, the other 410 images account for the other half
- 15% of stable charts utilize the Bats image (dduportal/bats:0.4.0) which is the image with the most vulnerable paths. This makes the image a potential vector for attacking the ecosystem. Bats is a popular testing tool, so coming up with an exploit to compromise valuable data might be difficult.

Vulnerabilities

- The most common types of vulnerabilities were out-of-bounds reads or writes, access restriction bypass, and NULL pointer dereference.
- 40,047 vulnerabilities found when each vulnerability is counted only once per image in which it appears

Remediation

- 176 stable Helm Charts (64%) can benefit from an image upgrade
- There are 261 image upgrades that can be made across the stable Helm Charts to improve security.

Helm Chart

- 277 stable Helm Charts
- 68% of stable Helm Charts contain an image with a high severity vulnerability

Images

- 416 images used across stable Helm Charts
- 6 images account for nearly half of all vulnerable paths, the other 410 images account for the other half
- 15% of stable charts utilize the Bats image (dduportal/bats:0.4.0) which is the image with the most vulnerable paths. This makes the image a potential vector for attacking the ecosystem. Bats is a popular testing tool, so coming up with an exploit to compromise valuable data might be difficult.

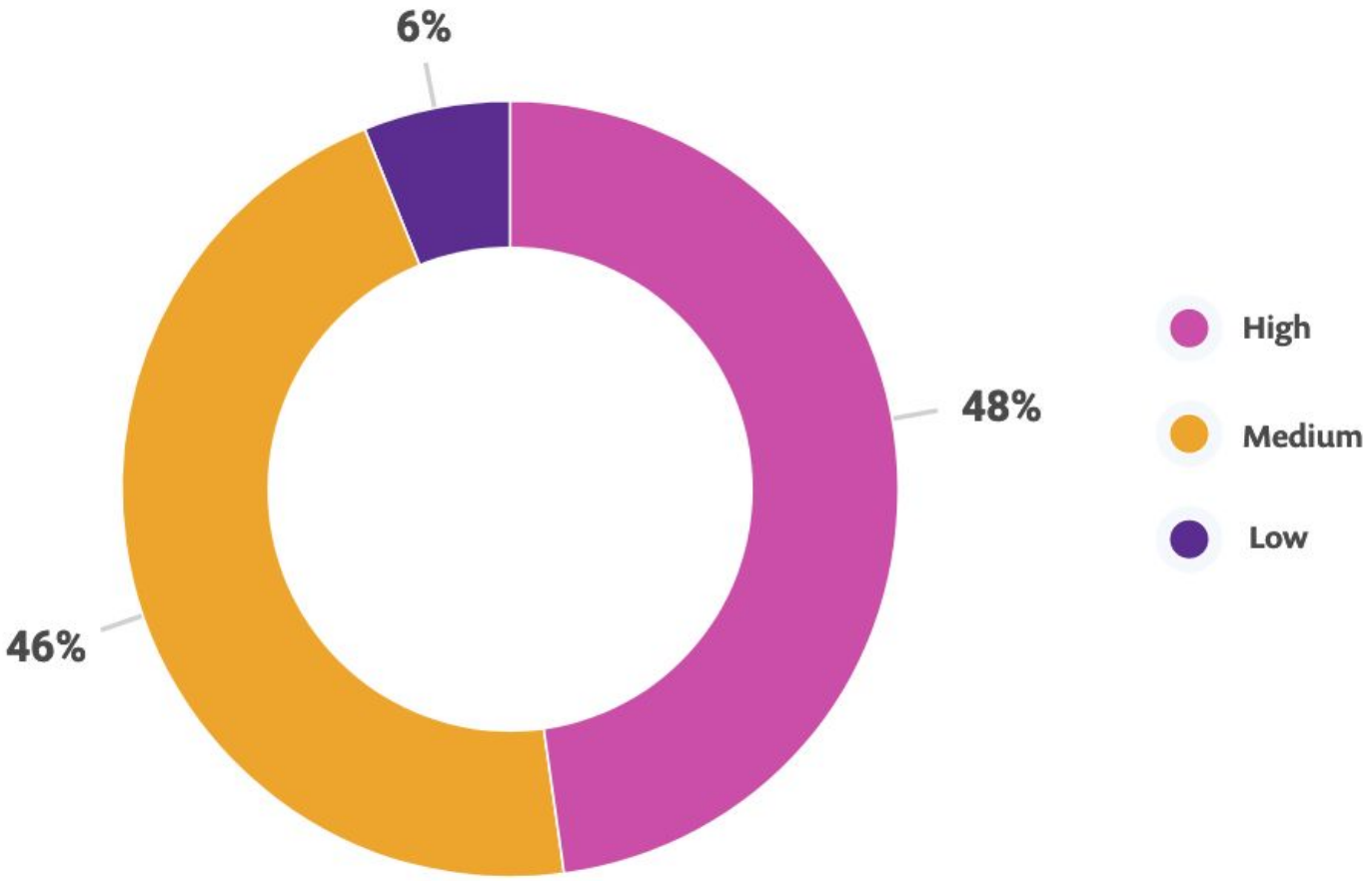
Vulnerabilities

- The most common types of vulnerabilities were out-of-bounds reads or writes, access restriction bypass, and NULL pointer dereference.
- 40,047 vulnerabilities found when each vulnerability is counted only once per image in which it appears

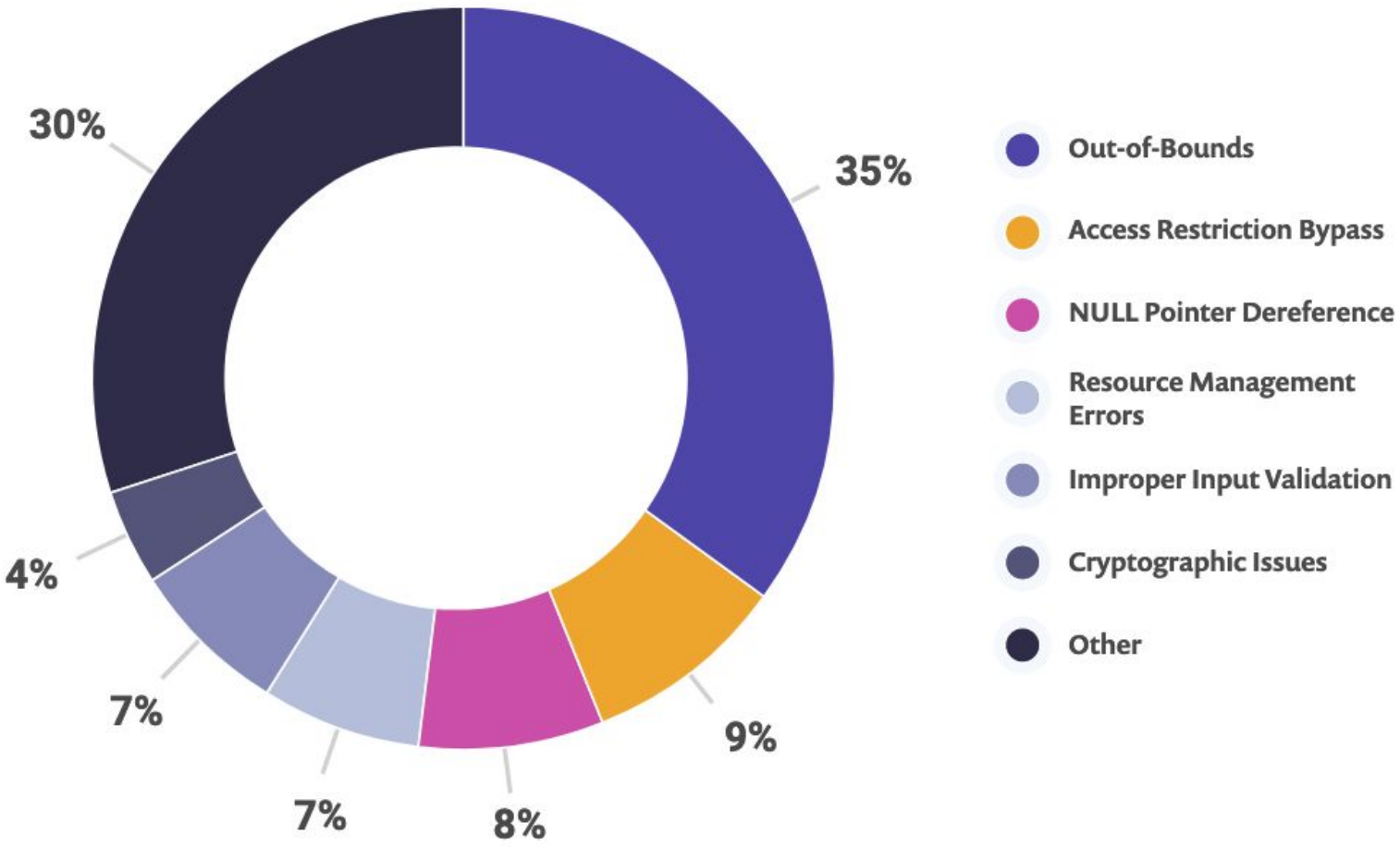
Remediation

- 176 stable Helm Charts (64%) can benefit from an image upgrade
- There are 261 image upgrades that can be made across the stable Helm Charts to improve security.

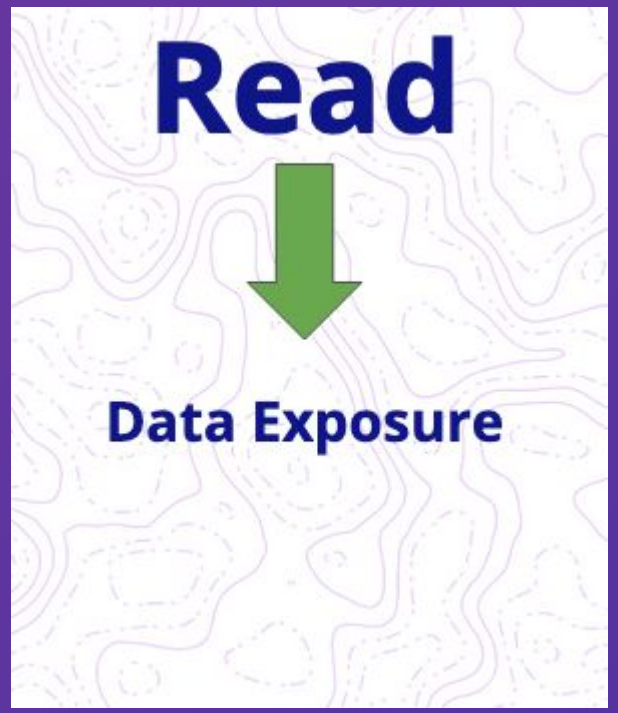
Vulnerability severity ratings

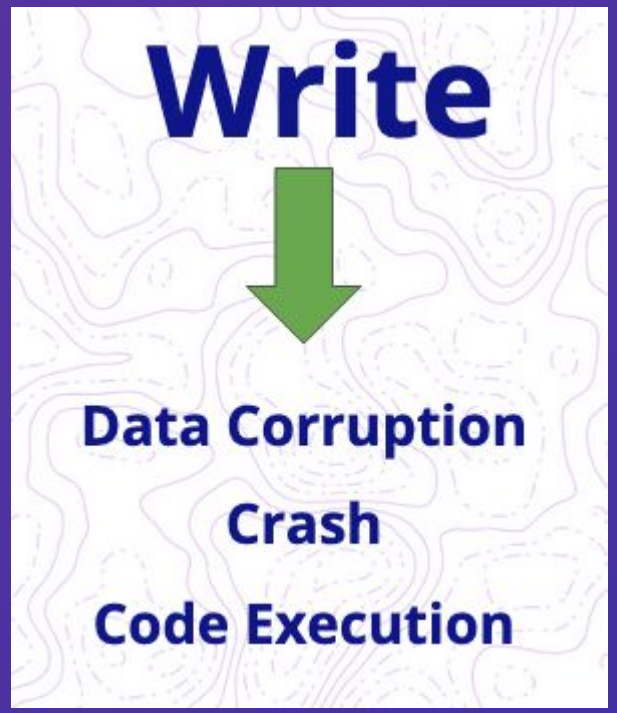


Vulnerability types



Out of Bounds





Access Restriction Bypass

1. May not check user identity correctly
2. User may be able to get around permissions
3. User's actions may not be logged correctly

NULL Pointer Dereference

1. Pointer with value NULL is treated as though it pointed to a valid memory area
2. Can be a security issue, but more likely to be a reliability concern

Final Helm Chart Security Thoughts

Resources

- Snyk security - https://snyk.io
- Helm Security Report - https://snyk.io/helm-report
- Container Vulnerability Management - https://snyk.io/product/container-vulnerability-management/

- Kubecon EU talks
  - Helm Security Report (Apr. 1st) - https://sched.co/Zel7
  - Kubernetes Security Panel (Apr. 1st) - https://sched.co/ZeqF

Helm Security - Q&A

Matt Farina - @mattfarina
Hayley Denbraver - @hayleydenb
Rags - @ragss

SAMSUNG SDS