AWS re:Invent

NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

CON209

Scanning containers for vulnerabilities

Senthil Kumaran (He/Him)

Senior Systems Development Engineer, EKS
AWS

My journey to this presentation

I joined EKS in Container Networking Team.

We learned about the needs of our customers.

Developed a system that will scan our container images for vulnerabilities and will help us stay on top.

Customers

In order to meet their growing demands for velocity, scale, and availability, customers have migrated their workloads to containers.

As mission-critical workloads are containerized and are deployed to production, customers are now worried about containers with common vulnerabilities and exposures.

Customers want to protect themselves from attacks. They have business mandates and security requirements to run container workloads without vulnerabilities.

AWS

Security is of paramount importance to AWS.

AWS has built multiple systems and processes to ensure that we stay on top of security issues and constantly works to detect and rectify issues before they impact customers.

In this talk, we will look at tools and services available to customers to scan their containerized workloads.

We will look at Amazon EKS team’s experience in handling container vulnerability issues.

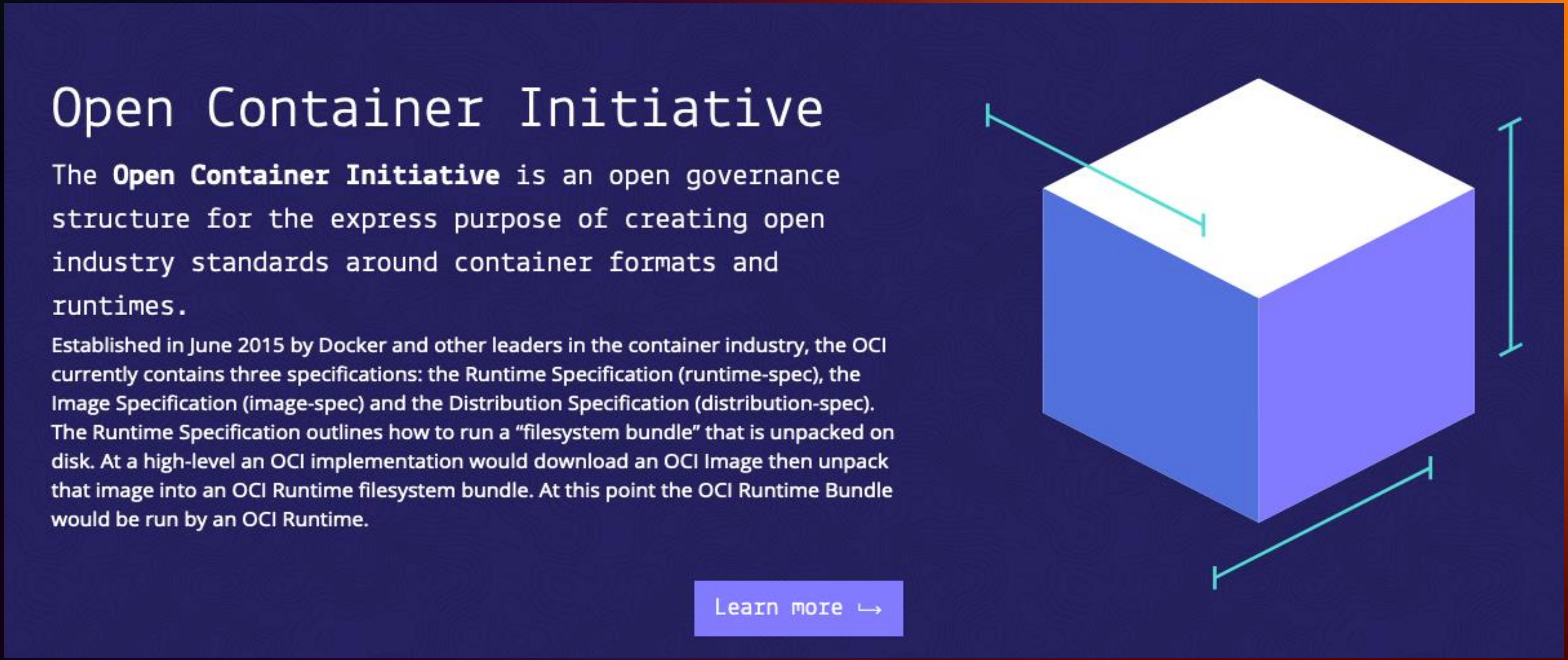
Containers

A container is like a lightweight virtual machine sharing kernel with the host.

Containers are created using a combination of kernel features such as Bind Mounts, Overlayfs, control groups, and namespaces.

An application container is started using a **container image**, which bundles the application together with its dependencies and just enough of a Linux root filesystem to run it.

Open Container Initiative



Open Container Initiative

The **Open Container Initiative** is an open governance structure for the express purpose of creating open industry standards around container formats and runtimes.

Established in June 2015 by Docker and other leaders in the container industry, the OCI currently contains three specifications: the Runtime Specification (runtime-spec), the Image Specification (image-spec) and the Distribution Specification (distribution-spec). The Runtime Specification outlines how to run a "filesystem bundle" that is unpacked on disk. At a high-level an OCI implementation would download an OCI Image then unpack that image into an OCI Runtime filesystem bundle. At this point the OCI Runtime Bundle would be run by an OCI Runtime.

Container images

Container images are executable software bundles that run standalone and that make very well-defined assumptions about their runtime environment.

Developers typically create a container image of the application and push it to a registry.

The container runtime is the software that is responsible for running containers.

Vulnerable Zones

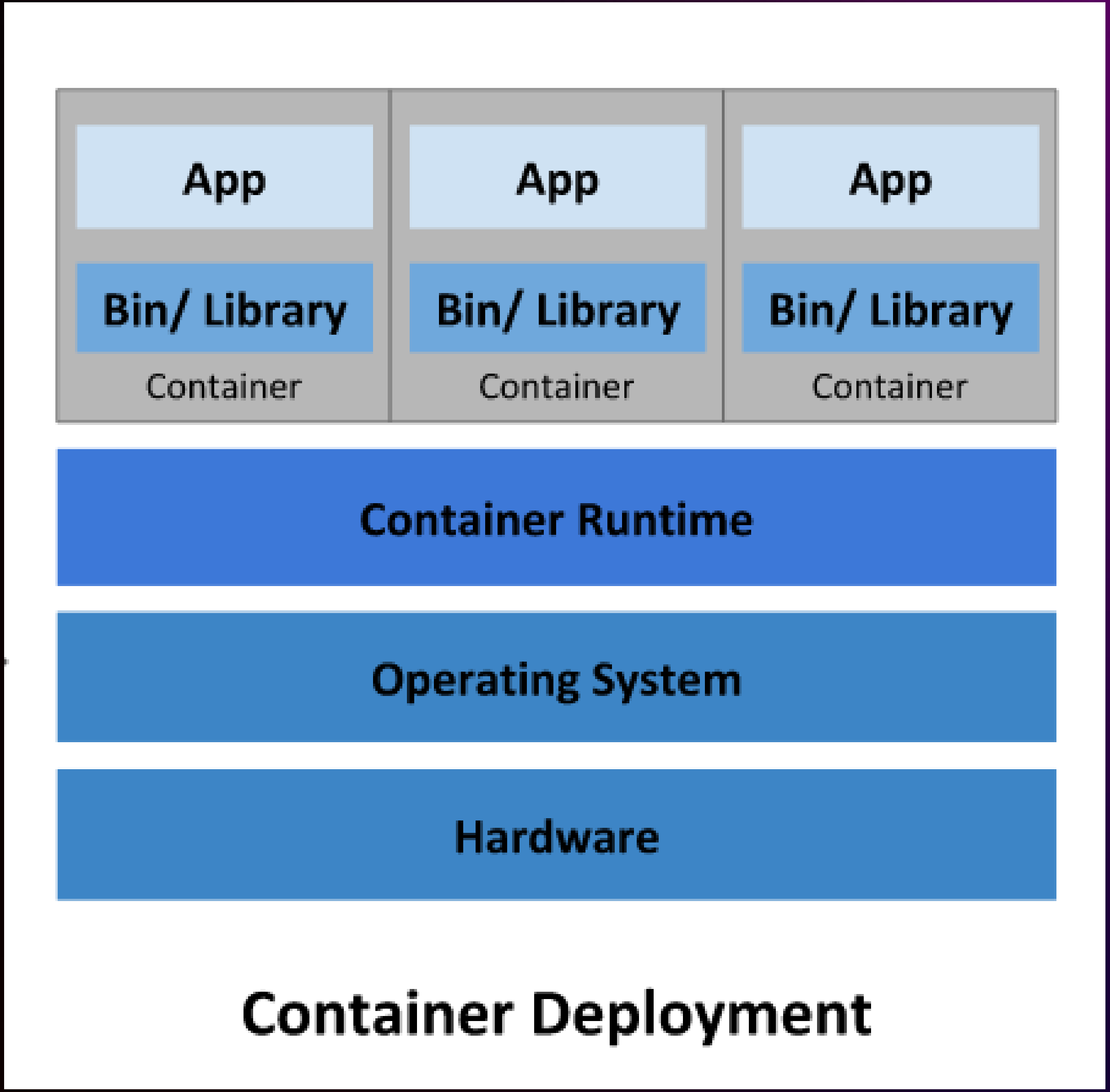
Container image layer

Application within the container

Container runtime

Linux Kernel in the host

Host machine software



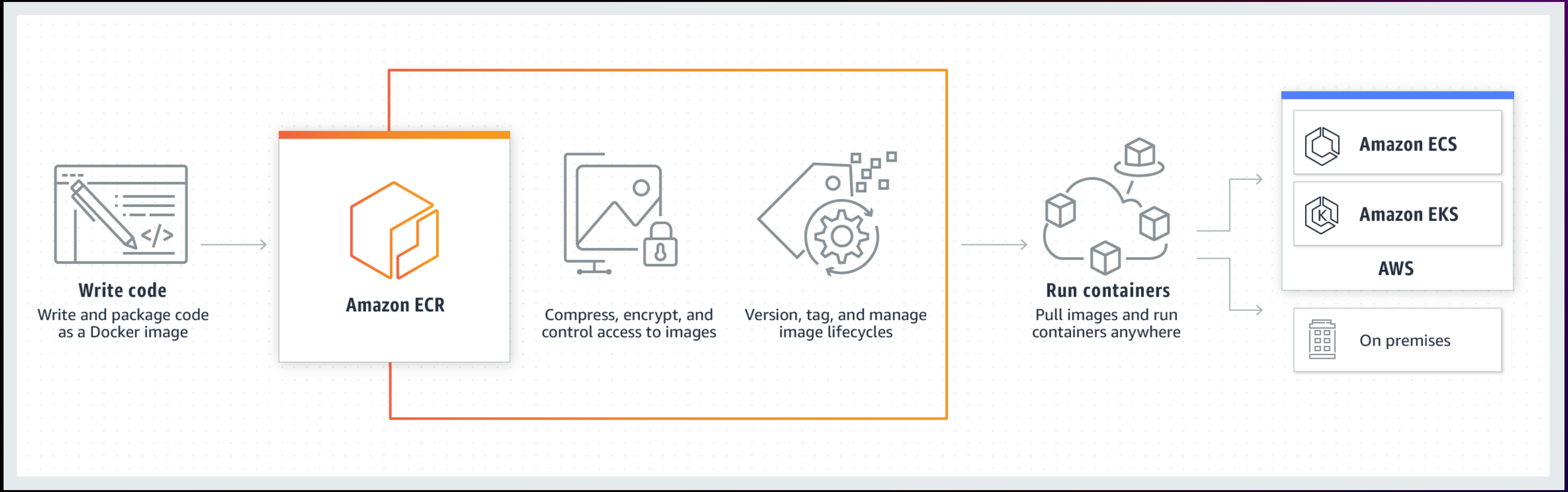
Detecting vulnerabilities

Domain of container scanning tools

Open specification leads to multiple solutions that customers can use

- Aqua Security
- Prisma Scan from Twistlock
- Clair
- Snyk
- Many more …

Amazon Elastic Container Registry


Write code
Write and package code as a Docker image
Amazon ECR
Compress, encrypt, and control access to images
Version, tag, and manage image lifecycles
Run containers
Pull images and run containers anywhere
Amazon ECS
Amazon EKS
AWS
On premises

Amazon ECR Scan

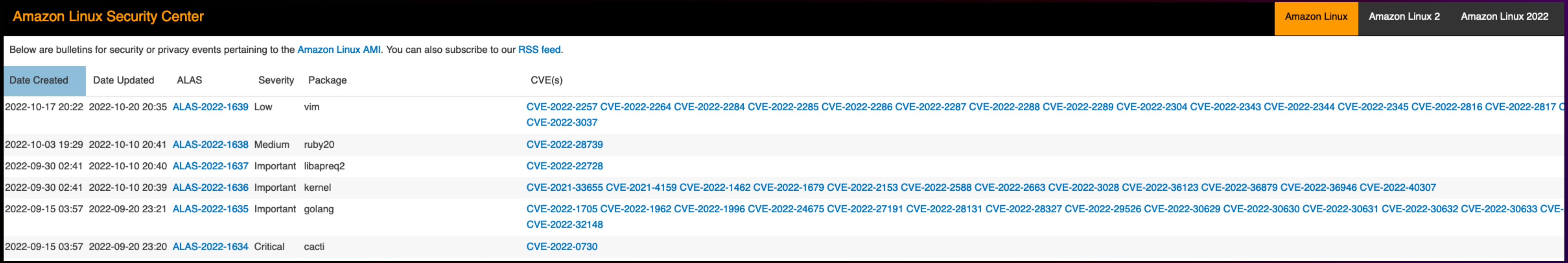
Amazon ECR supports scanning for containers in its private registries

Enhanced scanning—Amazon ECR integrates with Amazon Inspector to provide automated, continuous scanning

Basic scanning—Amazon ECR uses the Common Vulnerabilities and Exposures (CVEs) database from the open-source Clair project

Enhanced with periodically updated database from Amazon Linux Security Center (ALAS) for Amazon Linux containers

ALAS



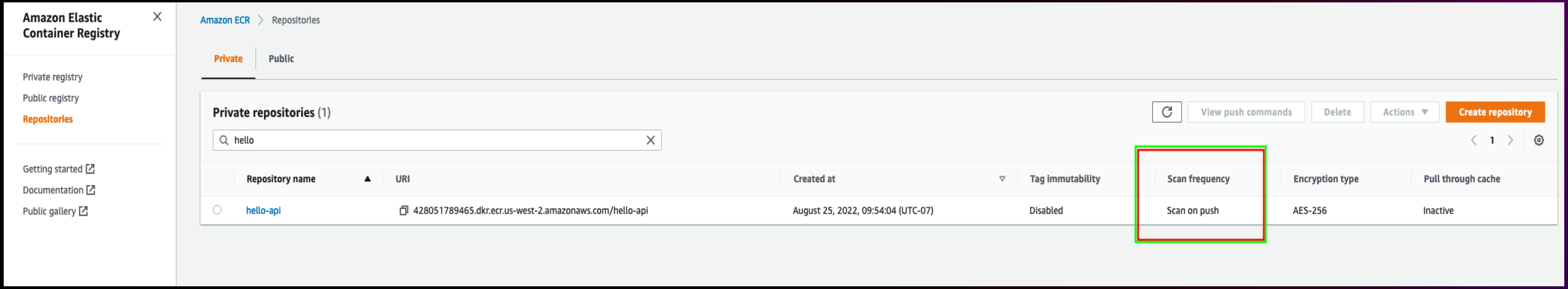
Amazon Linux Security Center

Below are bulletins for security or privacy events pertaining to the Amazon Linux AMI. You can also subscribe to our RSS feed.

Date Created	Date Updated	ALAS	Severity	Package	CVE(s)
2022-10-17 20:22	2022-10-20 20:35	ALAS-2022-1639	Low	vim	CVE-2022-2257 CVE-2022-2264 CVE-2022-2284 CVE-2022-2285 CVE-2022-2286 CVE-2022-2287 CVE-2022-2288 CVE-2022-2289 CVE-2022-2304 CVE-2022-2343 CVE-2022-2344 CVE-2022-2345 CVE-2022-2816 CVE-2022-2817 C CVE-2022-3037
2022-10-03 19:29	2022-10-10 20:41	ALAS-2022-1638	Medium	ruby20	CVE-2022-28739
2022-09-30 02:41	2022-10-10 20:40	ALAS-2022-1637	Important	libapreq2	CVE-2022-22728
2022-09-30 02:41	2022-10-10 20:39	ALAS-2022-1636	Important	kernel	CVE-2021-33655 CVE-2021-4159 CVE-2022-1462 CVE-2022-1679 CVE-2022-2153 CVE-2022-2588 CVE-2022-2663 CVE-2022-3028 CVE-2022-36123 CVE-2022-36879 CVE-2022-36946 CVE-2022-40307
2022-09-15 03:57	2022-09-20 23:21	ALAS-2022-1635	Important	golang	CVE-2022-1705 CVE-2022-1962 CVE-2022-1996 CVE-2022-24675 CVE-2022-27191 CVE-2022-28131 CVE-2022-28327 CVE-2022-29526 CVE-2022-30629 CVE-2022-30630 CVE-2022-30631 CVE-2022-30632 CVE-2022-30633 CVE- CVE-2022-32148
2022-09-15 03:57	2022-09-20 23:20	ALAS-2022-1634	Critical	cacti	CVE-2022-0730

Basic Scan

Container images are scanned for operating system vulnerabilities



Amazon Elastic Container Registry

- Private registry
- Public registry
- Repositories

- Getting started
- Documentation
- Public gallery

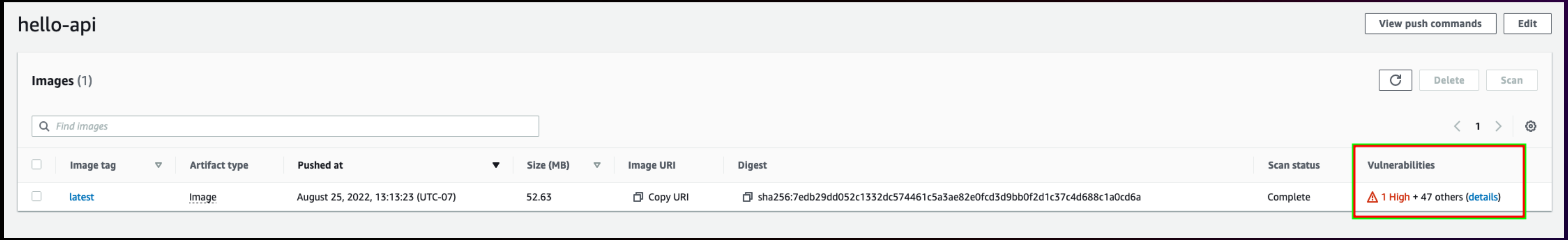
Amazon ECR > Repositories

Private | Public

Private repositories (1)

hello

Repository name	URI	Created at	Tag immutability	Scan frequency	Encryption type	Pull through cache
hello-api	428051789465.dkr.ecr.us-west-2.amazonaws.com/hello-api	August 25, 2022, 09:54:04 (UTC-07)	Disabled	Scan on push	AES-256	Inactive

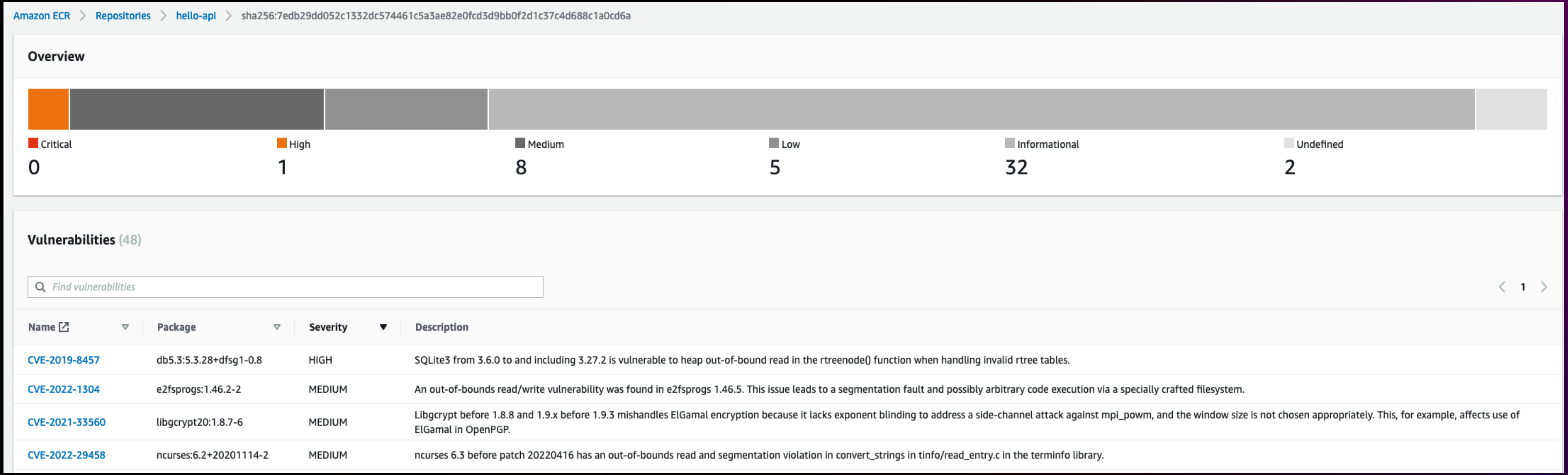


hello-api

Images (1)

Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest	Scan status	Vulnerabilities
latest	Image	August 25, 2022, 13:13:23 (UTC-07)	52.63	Copy URI	sha256:7edb29dd052c1332dc574461c5a3ae82e0fcd3d9bb0f2d1c37c4d688c1a0cd6a	Complete	⚠ 1 High + 47 others (details)

Basic Scan - Operating System Vulnerabilities



Amazon ECR > Repositories > hello-api > sha256:7edb29dd052c1332dc574461c5a3ae82e0fcd3d9bb0f2d1c37c4d688c1a0cd6a

Overview

🟥 Critical	🟧 High	▨ Medium	▨ Low	▨ Informational	▨ Undefined
0	1	8	5	32	2

Vulnerabilities (48)

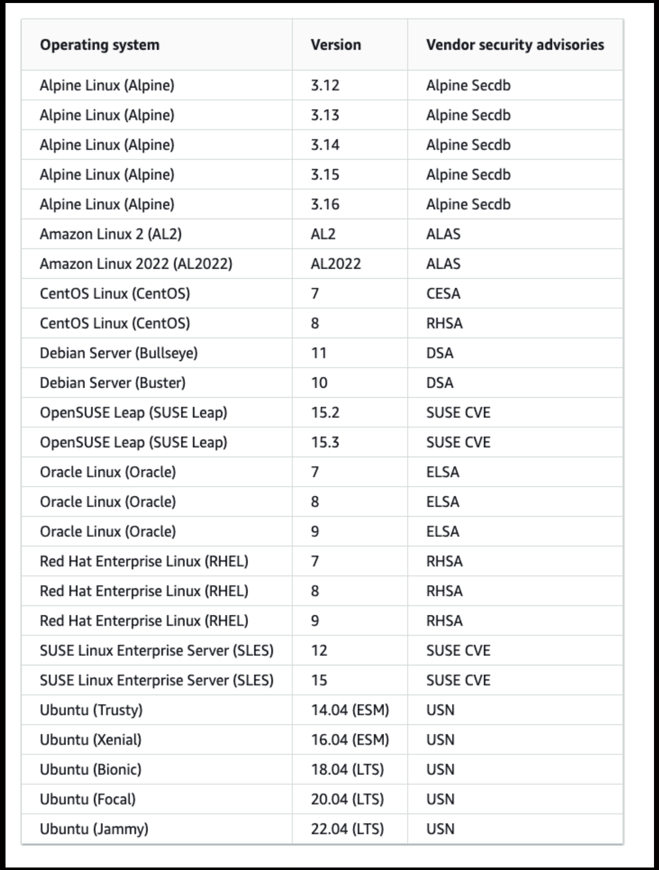
Find vulnerabilities

< 1 >

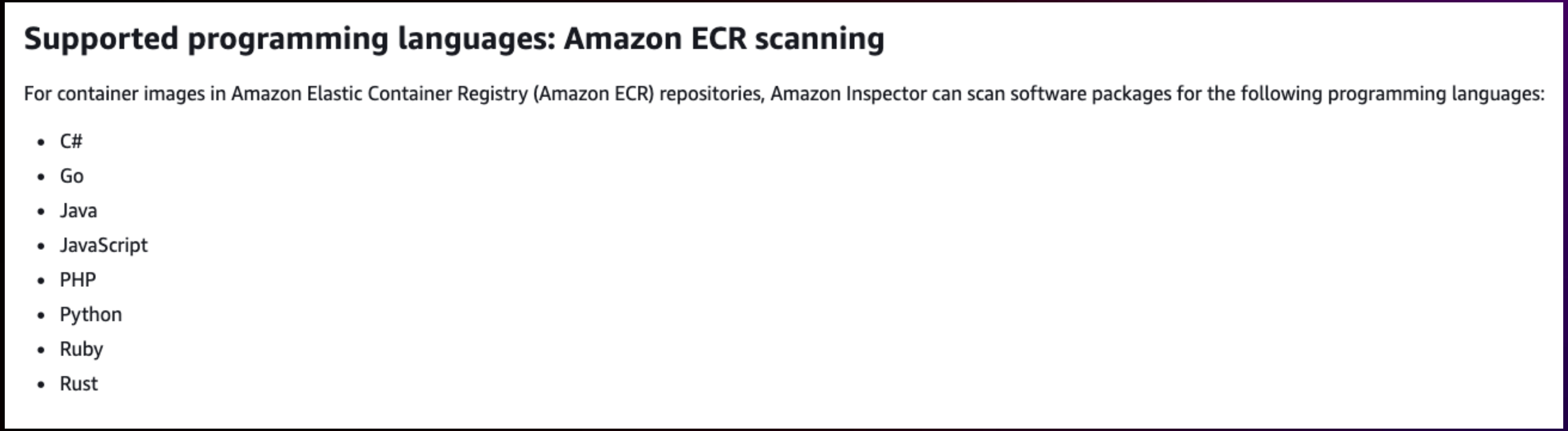
Name	Package	Severity	Description
CVE-2019-8457	db5.3:5.3.28+dfsg1-0.8	HIGH	SQLite3 from 3.6.0 to and including 3.27.2 is vulnerable to heap out-of-bound read in the rtreenode() function when handling invalid rtree tables.
CVE-2022-1304	e2fsprogs:1.46.2-2	MEDIUM	An out-of-bounds read/write vulnerability was found in e2fsprogs 1.46.5. This issue leads to a segmentation fault and possibly arbitrary code execution via a specially crafted filesystem.
CVE-2021-33560	libgcrypt20:1.8.7-6	MEDIUM	Libgcrypt before 1.8.8 and 1.9.x before 1.9.3 mishandles ElGamal encryption because it lacks exponent blinding to address a side-channel attack against mpi_powm, and the window size is not chosen appropriately. This, for example, affects use of ElGamal in OpenPGP.
CVE-2022-29458	ncurses:6.2+20201114-2	MEDIUM	ncurses 6.3 before patch 20220416 has an out-of-bounds read and segmentation violation in convert_strings in tinfo/read_entry.c in the terminfo library.

Enhanced Scan

Container images are scanned for both operating systems and programing language package vulnerabilities



Operating system	Version	Vendor security advisories
Alpine Linux (Alpine)	3.12	Alpine Secdb
Alpine Linux (Alpine)	3.13	Alpine Secdb
Alpine Linux (Alpine)	3.14	Alpine Secdb
Alpine Linux (Alpine)	3.15	Alpine Secdb
Alpine Linux (Alpine)	3.16	Alpine Secdb
Amazon Linux 2 (AL2)	AL2	ALAS
Amazon Linux 2022 (AL2022)	AL2022	ALAS
CentOS Linux (CentOS)	7	CESA
CentOS Linux (CentOS)	8	RHSA
Debian Server (Bullseye)	11	DSA
Debian Server (Buster)	10	DSA
OpenSUSE Leap (SUSE Leap)	15.2	SUSE CVE
OpenSUSE Leap (SUSE Leap)	15.3	SUSE CVE
Oracle Linux (Oracle)	7	ELSA
Oracle Linux (Oracle)	8	ELSA
Oracle Linux (Oracle)	9	ELSA
Red Hat Enterprise Linux (RHEL)	7	RHSA
Red Hat Enterprise Linux (RHEL)	8	RHSA
Red Hat Enterprise Linux (RHEL)	9	RHSA
SUSE Linux Enterprise Server (SLES)	12	SUSE CVE
SUSE Linux Enterprise Server (SLES)	15	SUSE CVE
Ubuntu (Trusty)	14.04 (ESM)	USN
Ubuntu (Xenial)	16.04 (ESM)	USN
Ubuntu (Bionic)	18.04 (LTS)	USN
Ubuntu (Focal)	20.04 (LTS)	USN
Ubuntu (Jammy)	22.04 (LTS)	USN

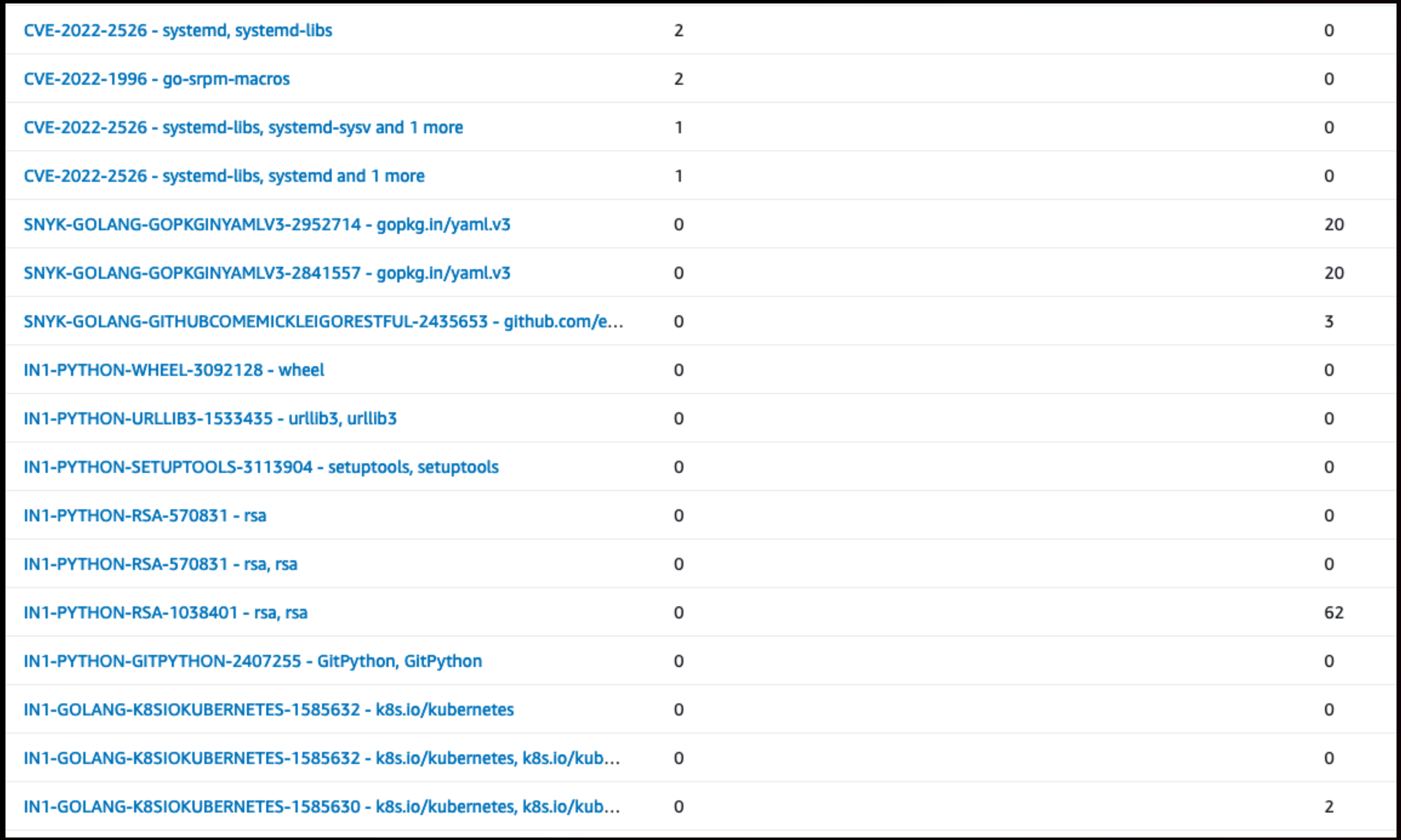


Supported programming languages: Amazon ECR scanning

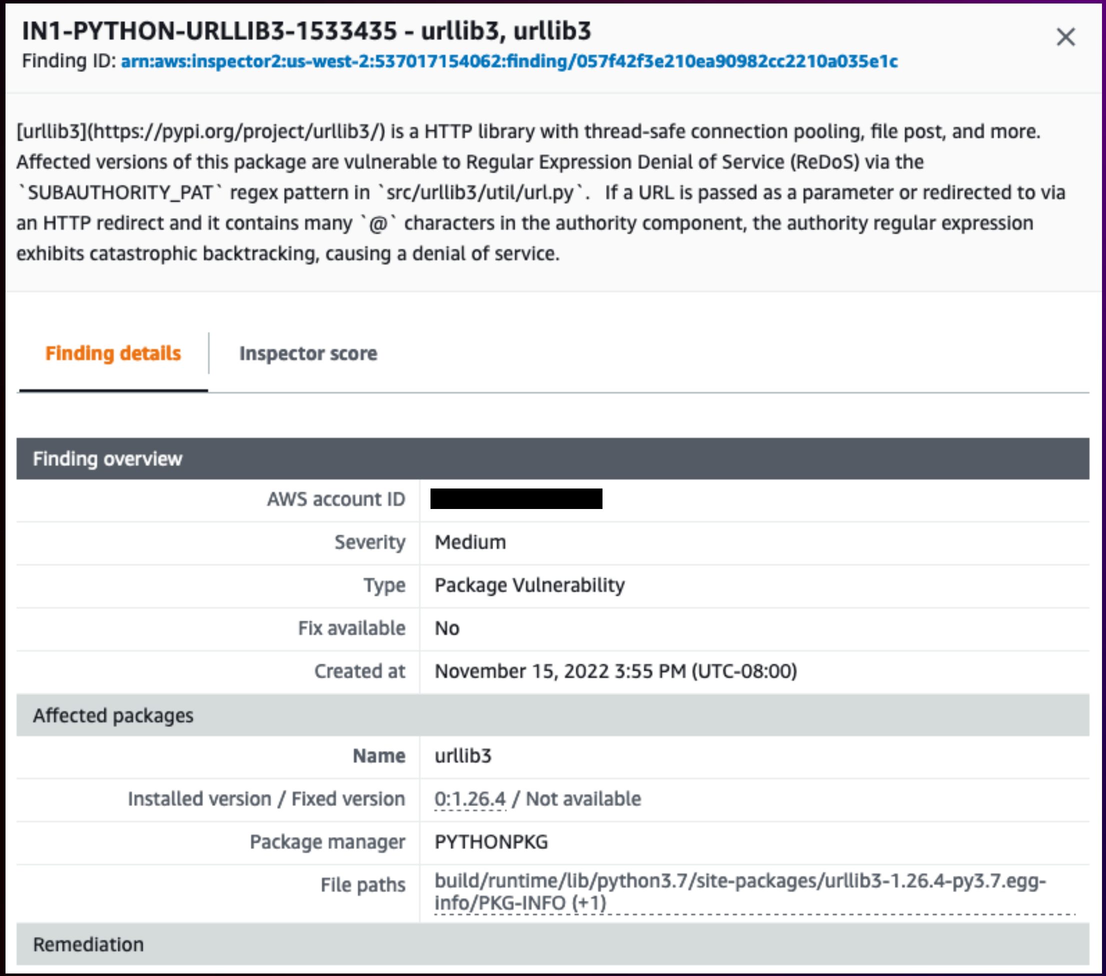
For container images in Amazon Elastic Container Registry (Amazon ECR) repositories, Amazon Inspector can scan software packages for the following programming languages:

- C#
- Go
- Java
- JavaScript
- PHP
- Python
- Ruby
- Rust

Languages and Runtime Vulnerability



CVE-2022-2526 - systemd, systemd-libs	2	0
CVE-2022-1996 - go-srpm-macros	2	0
CVE-2022-2526 - systemd-libs, systemd-sysv and 1 more	1	0
CVE-2022-2526 - systemd-libs, systemd and 1 more	1	0
SNYK-GOLANG-GOPKGINYAMLV3-2952714 - gopkg.in/yaml.v3	0	20
SNYK-GOLANG-GOPKGINYAMLV3-2841557 - gopkg.in/yaml.v3	0	20
SNYK-GOLANG-GITHUBCOMEMICKLEIGORESTFUL-2435653 - github.com/e...	0	3
IN1-PYTHON-WHEEL-3092128 - wheel	0	0
IN1-PYTHON-URLLIB3-1533435 - urllib3, urllib3	0	0
IN1-PYTHON-SETUPTOOLS-3113904 - setuptools, setuptools	0	0
IN1-PYTHON-RSA-570831 - rsa	0	0
IN1-PYTHON-RSA-570831 - rsa, rsa	0	0
IN1-PYTHON-RSA-1038401 - rsa, rsa	0	62
IN1-PYTHON-GITPYTHON-2407255 - GitPython, GitPython	0	0
IN1-GOLANG-K8SIOKUBERNETES-1585632 - k8s.io/kubernetes	0	0
IN1-GOLANG-K8SIOKUBERNETES-1585632 - k8s.io/kubernetes, k8s.io/kub...	0	0
IN1-GOLANG-K8SIOKUBERNETES-1585630 - k8s.io/kubernetes, k8s.io/kub...	0	2



IN1-PYTHON-URLLIB3-1533435 - urllib3, urllib3

Finding ID: arn:aws:inspector2:us-west-2:537017154062:finding/057f42f3e210ea90982cc2210a035e1c

[urllib3](https://pypi.org/project/urllib3/) is a HTTP library with thread-safe connection pooling, file post, and more. Affected versions of this package are vulnerable to Regular Expression Denial of Service (ReDoS) via the `SUBAUTHORITY_PAT` regex pattern in `src/urllib3/util/url.py`. If a URL is passed as a parameter or redirected to via an HTTP redirect and it contains many `@` characters in the authority component, the authority regular expression exhibits catastrophic backtracking, causing a denial of service.

Finding overview

AWS account ID	[redacted]
Severity	Medium
Type	Package Vulnerability
Fix available	No
Created at	November 15, 2022 3:55 PM (UTC-08:00)

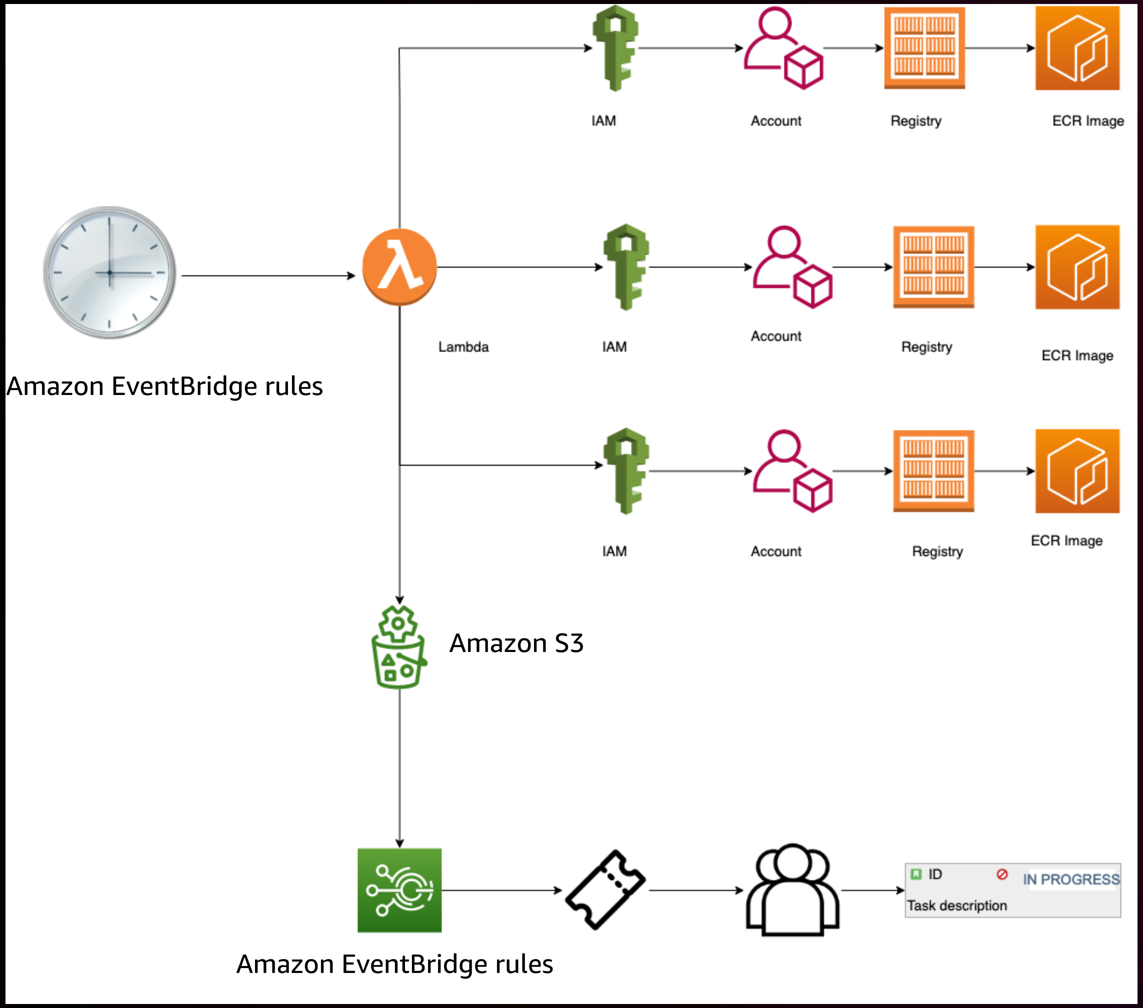
Affected packages

Name	urllib3
Installed version / Fixed version	0:1.26.4 / Not available
Package manager	PYTHONPKG
File paths	build/runtime/lib/python3.7/site-packages/urllib3-1.26.4-py3.7.egg-info/PKG-INFO (+1)

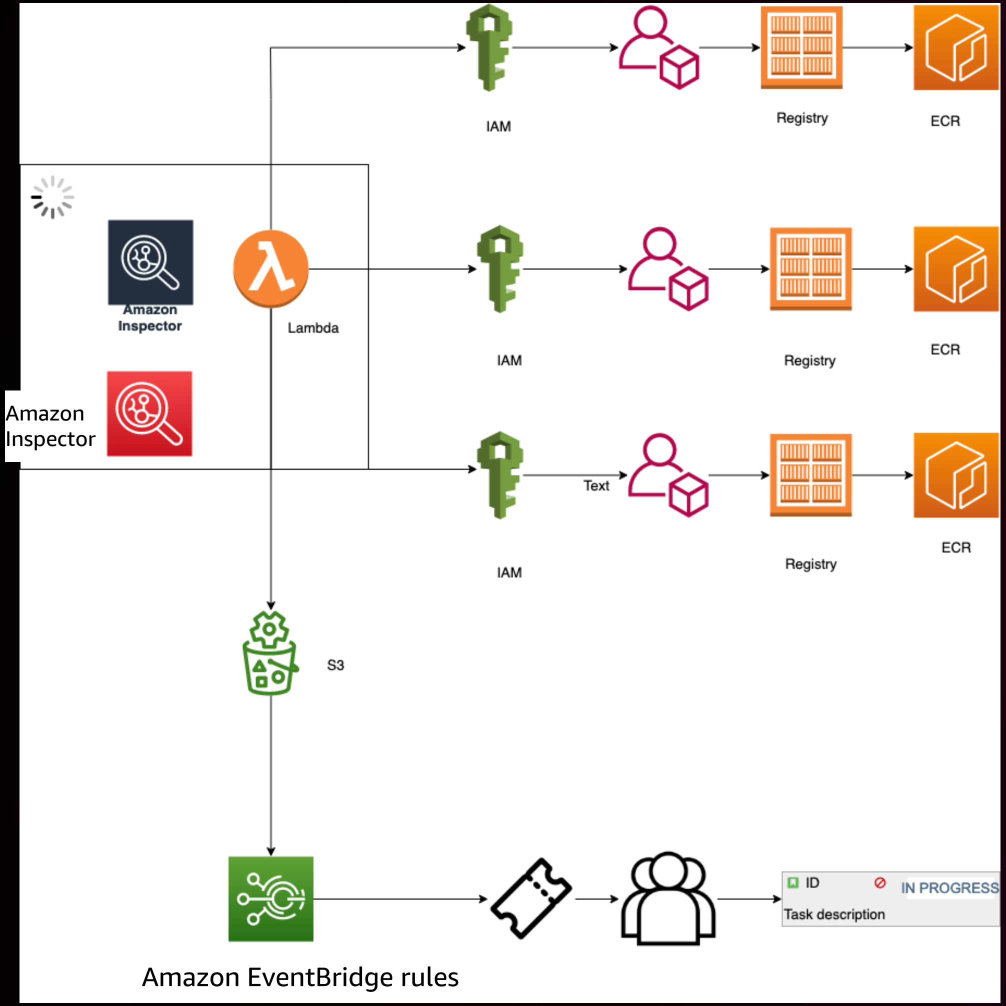
Remediation

How do we build the system using blocks now?

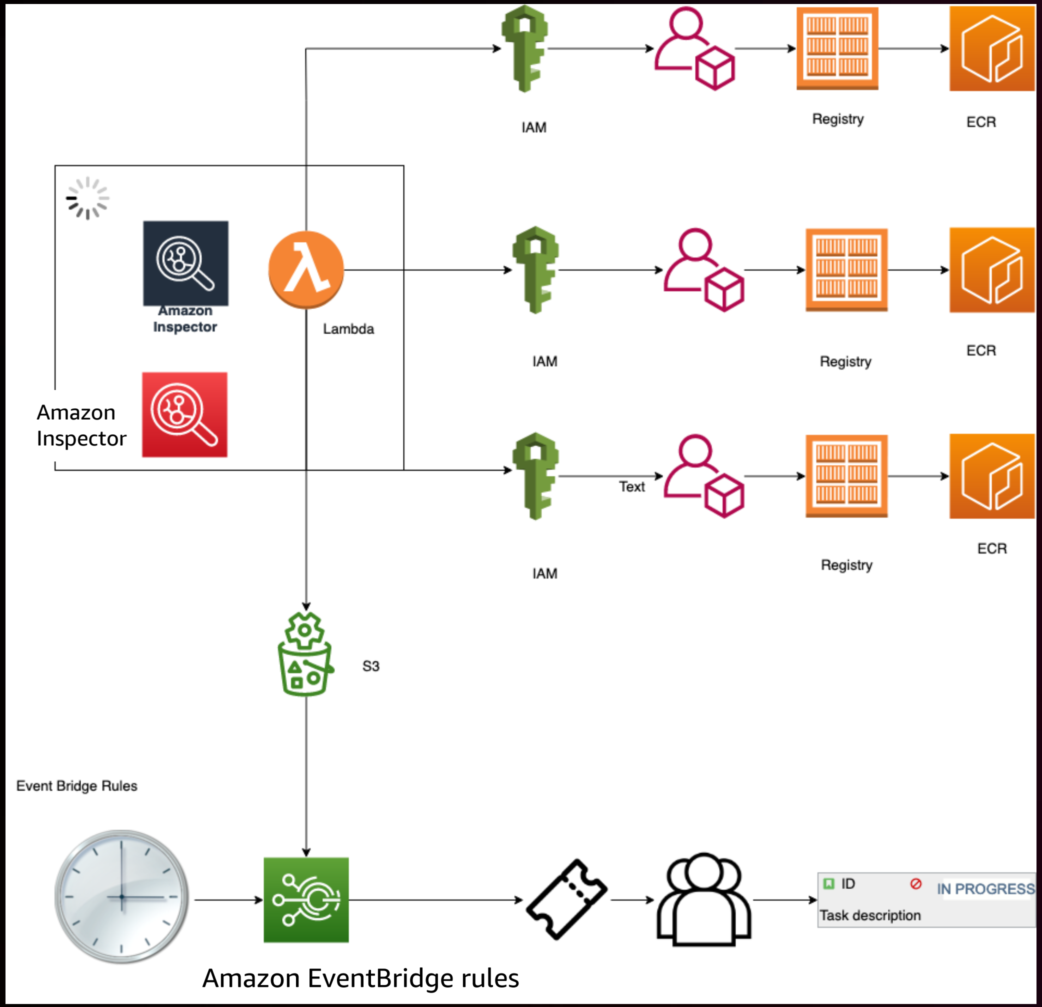
Scan Images Periodically



Continuously



Use Both – Continuous and Event Driven



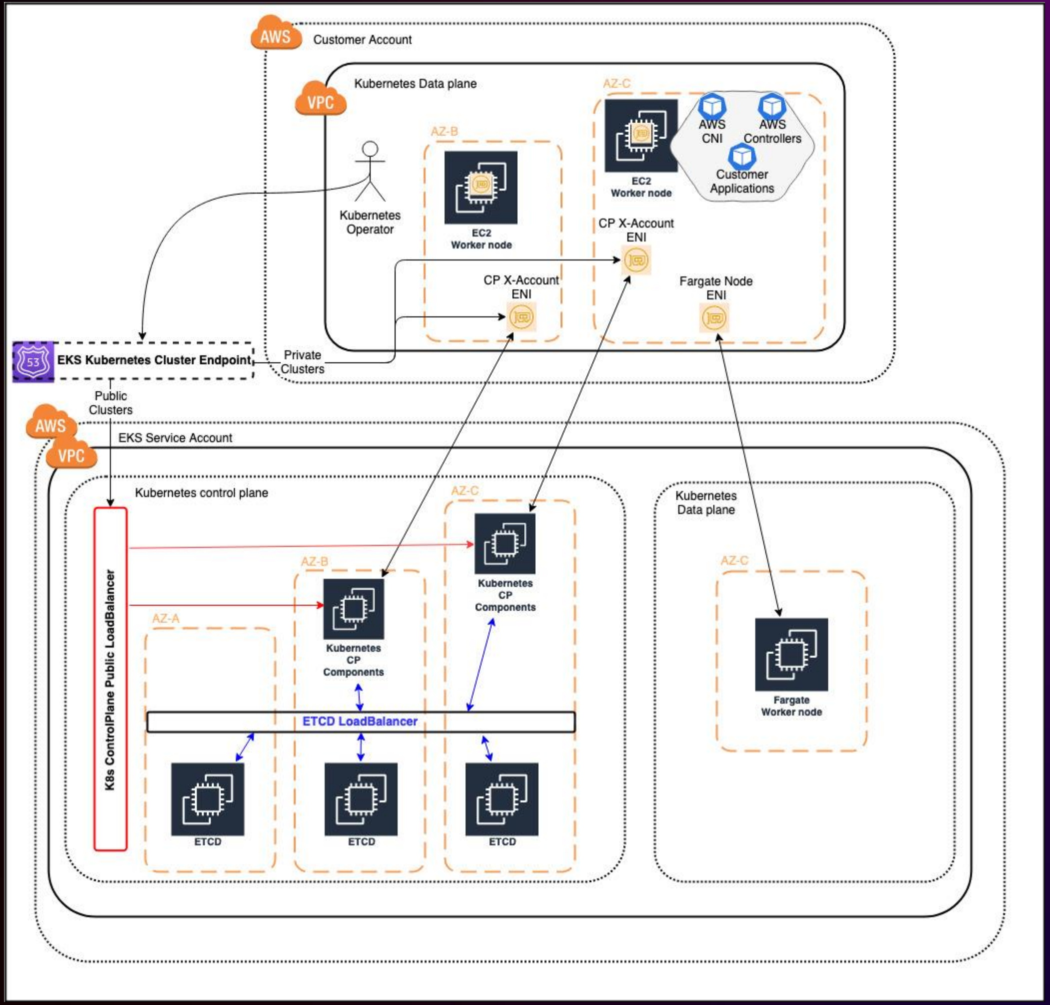
Amazon EKS

Amazon EKS has containers in Managed End of the Service called Control Plane.

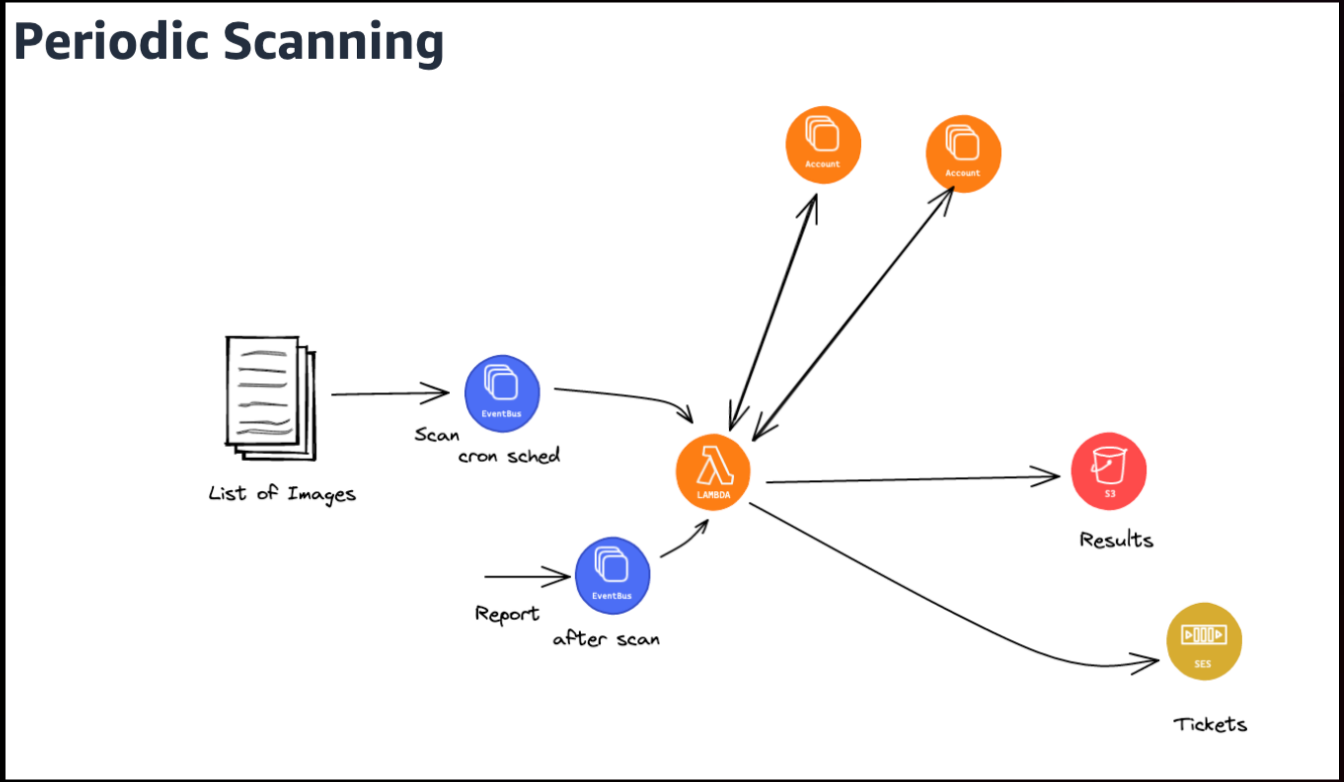
In the Customer Accounts running workloads.

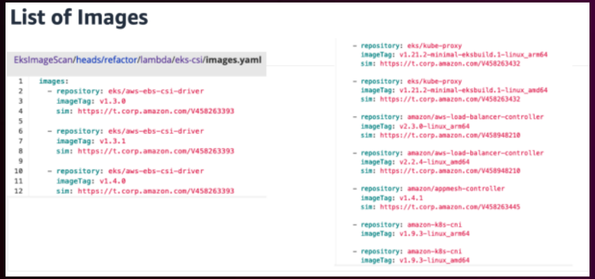
In AWS Fargate Compute Nodes.

Amazon EKS scans for vulnerabilities in the images and we take remediate actions to resolve the vulnerabilities.



System Design



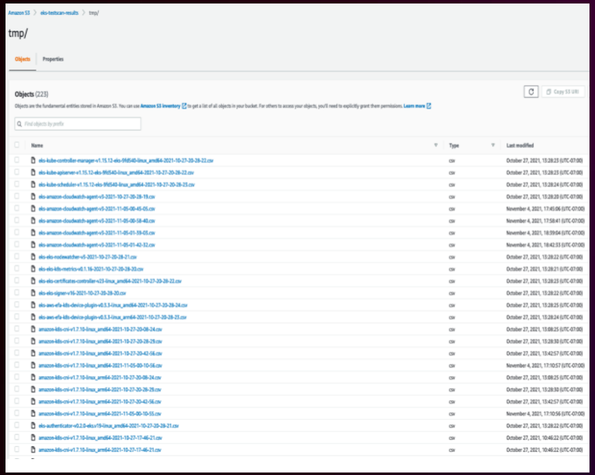


List of Images

EksImageScan/heads/refactor/lambda/eks-csi/images.yaml

```yaml
images:
  - repository: eks/aws-ebs-csi-driver
    imageTag: v1.3.0
    sim: https://t.corp.amazon.com/V458263393

  - repository: eks/aws-ebs-csi-driver
    imageTag: v1.3.1
    sim: https://t.corp.amazon.com/V458263393

  - repository: eks/aws-ebs-csi-driver
    imageTag: v1.4.0
    sim: https://t.corp.amazon.com/V458263393
```

```yaml
  - repository: eks/kube-proxy
    imageTag: v1.21.2-minimal-eksbuild.1-linux_arm64
    sim: https://t.corp.amazon.com/V458263432

  - repository: eks/kube-proxy
    imageTag: v1.21.2-minimal-eksbuild.1-linux_amd64
    sim: https://t.corp.amazon.com/V458263432

  - repository: amazon/aws-load-balancer-controller
    imageTag: v2.3.0-linux_arm64
    sim: https://t.corp.amazon.com/V458948210

  - repository: amazon/aws-load-balancer-controller
    imageTag: v2.2.4-linux_amd64
    sim: https://t.corp.amazon.com/V458948210

  - repository: amazon/appmesh-controller
    imageTag: v1.4.1
    sim: https://t.corp.amazon.com/V458263445

  - repository: amazon-k8s-cni
    imageTag: v1.9.3-linux_arm64

  - repository: amazon-k8s-cni
    imageTag: v1.9.3-linux_amd64
```



Amazon S3 > eks-testscan-results > tmp/

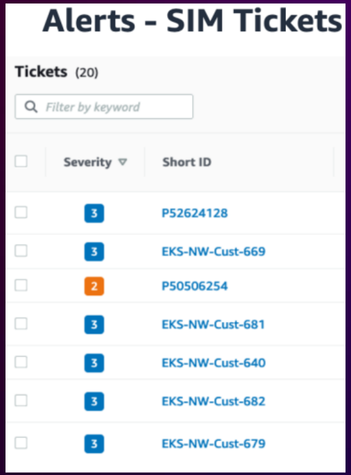
tmp/

Objects | Properties

Objects (223)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified
eks-kube-controller-manager-v1.15.12-eks-9fd540-linux_amd64-2021-10-27-20-28-22.csv	csv	October 27, 2021, 13:28:23 (UTC-07:00)
eks-kube-apiserver-v1.15.12-eks-9fd540-linux_amd64-2021-10-27-20-28-22.csv	csv	October 27, 2021, 13:28:23 (UTC-07:00)
eks-kube-scheduler-v1.15.12-eks-9fd540-linux_amd64-2021-10-27-20-28-23.csv	csv	October 27, 2021, 13:28:24 (UTC-07:00)
eks-amazon-cloudwatch-agent-v5-2021-10-27-20-28-19.csv	csv	October 27, 2021, 13:28:20 (UTC-07:00)
eks-amazon-cloudwatch-agent-v5-2021-11-05-00-45-05.csv	csv	November 4, 2021, 17:45:06 (UTC-07:00)
eks-amazon-cloudwatch-agent-v5-2021-11-05-00-58-40.csv	csv	November 4, 2021, 17:58:41 (UTC-07:00)
eks-amazon-cloudwatch-agent-v5-2021-11-05-01-39-03.csv	csv	November 4, 2021, 18:39:04 (UTC-07:00)
eks-amazon-cloudwatch-agent-v5-2021-11-05-01-42-32.csv	csv	November 4, 2021, 18:42:33 (UTC-07:00)
eks-eks-nodewatcher-v5-2021-10-27-20-28-21.csv	csv	October 27, 2021, 13:28:22 (UTC-07:00)
eks-eks-k8s-metrics-v0.1.16-2021-10-27-20-28-20.csv	csv	October 27, 2021, 13:28:21 (UTC-07:00)
eks-eks-certificates-controller-v23-linux_amd64-2021-10-27-20-28-22.csv	csv	October 27, 2021, 13:28:23 (UTC-07:00)
eks-eks-signer-v16-2021-10-27-20-28-20.csv	csv	October 27, 2021, 13:28:22 (UTC-07:00)
eks-aws-efa-k8s-device-plugin-v0.3.3-linux_amd64-2021-10-27-20-28-24.csv	csv	October 27, 2021, 13:28:25 (UTC-07:00)
eks-aws-efa-k8s-device-plugin-v0.3.3-linux_arm64-2021-10-27-20-28-23.csv	csv	October 27, 2021, 13:28:24 (UTC-07:00)
amazon-k8s-cni-v1.7.10-linux_amd64-2021-10-27-20-08-24.csv	csv	October 27, 2021, 13:08:25 (UTC-07:00)
amazon-k8s-cni-v1.7.10-linux_amd64-2021-10-27-20-28-29.csv	csv	October 27, 2021, 13:28:30 (UTC-07:00)
amazon-k8s-cni-v1.7.10-linux_amd64-2021-10-27-20-42-56.csv	csv	October 27, 2021, 13:42:57 (UTC-07:00)
amazon-k8s-cni-v1.7.10-linux_amd64-2021-11-05-00-10-56.csv	csv	November 4, 2021, 17:10:57 (UTC-07:00)
amazon-k8s-cni-v1.7.10-linux_arm64-2021-10-27-20-08-24.csv	csv	October 27, 2021, 13:08:25 (UTC-07:00)
amazon-k8s-cni-v1.7.10-linux_arm64-2021-10-27-20-28-29.csv	csv	October 27, 2021, 13:28:30 (UTC-07:00)
amazon-k8s-cni-v1.7.10-linux_arm64-2021-10-27-20-42-56.csv	csv	October 27, 2021, 13:42:57 (UTC-07:00)
amazon-k8s-cni-v1.7.10-linux_arm64-2021-11-05-00-10-55.csv	csv	November 4, 2021, 17:10:56 (UTC-07:00)
eks-authenticator-v0.2.0-eks.v19-linux_amd64-2021-10-27-20-28-21.csv	csv	October 27, 2021, 13:28:22 (UTC-07:00)
amazon-k8s-cni-v1.7.10-linux_amd64-2021-10-27-17-46-21.csv	csv	October 27, 2021, 10:46:22 (UTC-07:00)
amazon-k8s-cni-v1.7.10-linux_arm64-2021-10-27-17-46-21.csv	csv	October 27, 2021, 10:46:22 (UTC-07:00)



Alerts - SIM Tickets

Tickets (20)

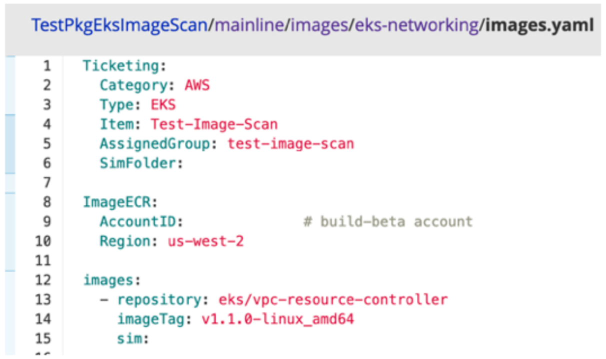
Filter by keyword

Severity	Short ID
3	P52624128
3	EKS-NW-Cust-669
2	P50506254
3	EKS-NW-Cust-681
3	EKS-NW-Cust-640
3	EKS-NW-Cust-682
3	EKS-NW-Cust-679

Enhanced Scanning

Support for Enhanced Scanning

- Basic Scan:
  - Triggered manually

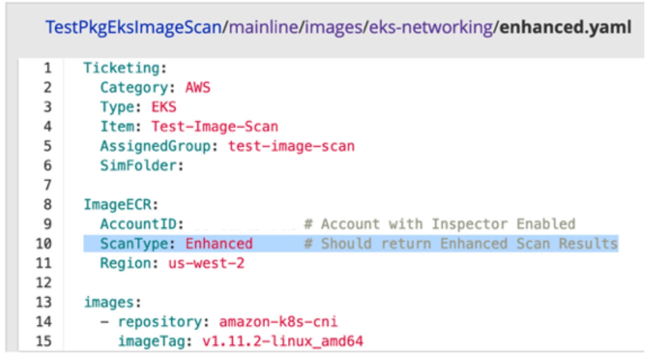


TestPkgEksImageScan/mainline/images/eks-networking/**images.yaml**

```yaml
Ticketing:
  Category: AWS
  Type: EKS
  Item: Test-Image-Scan
  AssignedGroup: test-image-scan
  SimFolder:

ImageECR:
  AccountID:              # build-beta account
  Region: us-west-2

images:
  - repository: eks/vpc-resource-controller
    imageTag: v1.1.0-linux_amd64
    sim:
```

- ScanType: Basic

- Enhanced Scan:
  - Continuous scan, triggered automatically



TestPkgEksImageScan/mainline/images/eks-networking/**enhanced.yaml**

```yaml
Ticketing:
  Category: AWS
  Type: EKS
  Item: Test-Image-Scan
  AssignedGroup: test-image-scan
  SimFolder:

ImageECR:
  AccountID:              # Account with Inspector Enabled
  ScanType: Enhanced      # Should return Enhanced Scan Results
  Region: us-west-2

images:
  - repository: amazon-k8s-cni
    imageTag: v1.11.2-linux_amd64
```

- ScanType: Enhanced
- Added reporting the enhanced scan results with Sim Ticket

Architectural patterns customers can adopt

There are multiple ways customers can use the Amazon ECR technology to their advantage.

Use a architecture similar to one described in previous slides for your application containers.

Monitor and evaluate the reports continuously to find real issues from non-applicable red herrings.

Prevent deployment of application containers from Amazon ECR if they have vulnerabilities. Has production impact.

Minimize Attack Vectors

Do not include shell in your containers.

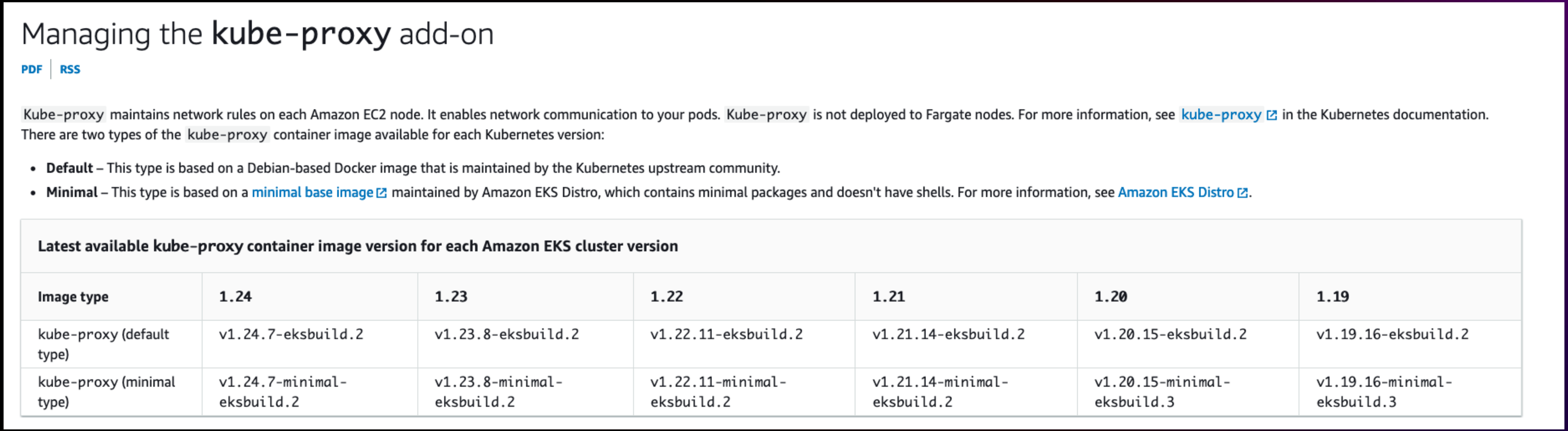
Just enough dependencies, nothing more.

Amazon EKS minimal base images

"Distroless" Container Images

Use static analysis tools on the binaries used in containers, especially the entry point binaries.

Example: Kube-proxy



Managing the **kube-proxy** add-on

PDF | RSS

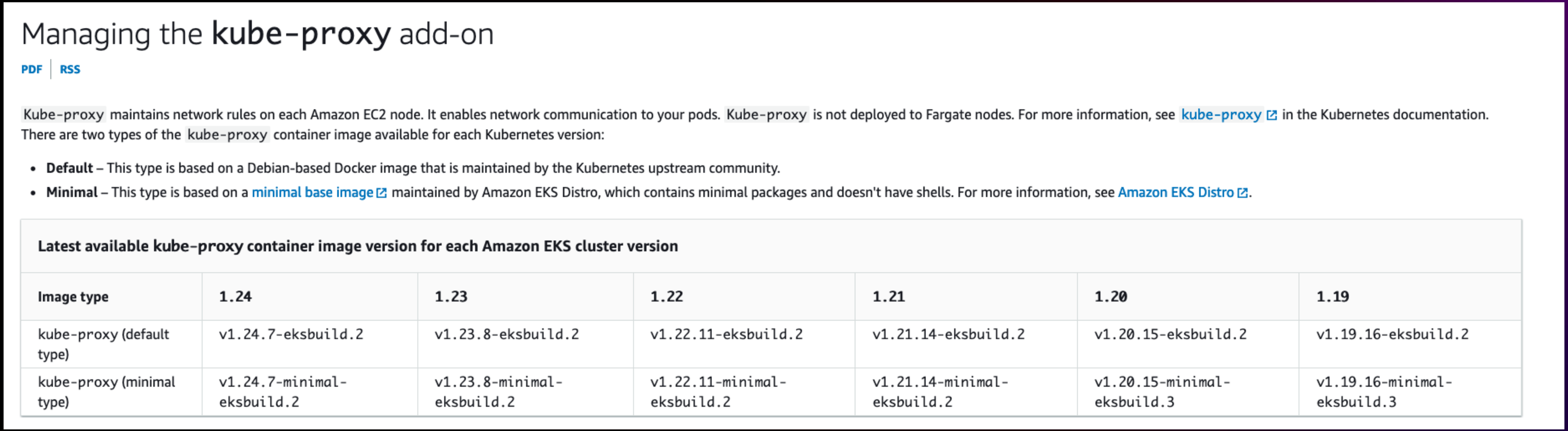
`Kube-proxy` maintains network rules on each Amazon EC2 node. It enables network communication to your pods. `Kube-proxy` is not deployed to Fargate nodes. For more information, see kube-proxy in the Kubernetes documentation. There are two types of the `kube-proxy` container image available for each Kubernetes version:

- **Default** – This type is based on a Debian-based Docker image that is maintained by the Kubernetes upstream community.
- **Minimal** – This type is based on a minimal base image maintained by Amazon EKS Distro, which contains minimal packages and doesn't have shells. For more information, see Amazon EKS Distro.

Latest available kube-proxy container image version for each Amazon EKS cluster version

Image type	1.24	1.23	1.22	1.21	1.20	1.19
kube-proxy (default type)	v1.24.7-eksbuild.2	v1.23.8-eksbuild.2	v1.22.11-eksbuild.2	v1.21.14-eksbuild.2	v1.20.15-eksbuild.2	v1.19.16-eksbuild.2
kube-proxy (minimal type)	v1.24.7-minimal-eksbuild.2	v1.23.8-minimal-eksbuild.2	v1.22.11-minimal-eksbuild.2	v1.21.14-minimal-eksbuild.2	v1.20.15-minimal-eksbuild.3	v1.19.16-minimal-eksbuild.3

Example: Kube-proxy



Managing the kube-proxy add-on

PDF | RSS

`Kube-proxy` maintains network rules on each Amazon EC2 node. It enables network communication to your pods. `Kube-proxy` is not deployed to Fargate nodes. For more information, see kube-proxy in the Kubernetes documentation. There are two types of the `kube-proxy` container image available for each Kubernetes version:

- **Default** – This type is based on a Debian-based Docker image that is maintained by the Kubernetes upstream community.
- **Minimal** – This type is based on a minimal base image maintained by Amazon EKS Distro, which contains minimal packages and doesn't have shells. For more information, see Amazon EKS Distro.

Latest available kube-proxy container image version for each Amazon EKS cluster version

Image type	1.24	1.23	1.22	1.21	1.20	1.19
kube-proxy (default type)	v1.24.7-eksbuild.2	v1.23.8-eksbuild.2	v1.22.11-eksbuild.2	v1.21.14-eksbuild.2	v1.20.15-eksbuild.2	v1.19.16-eksbuild.2
kube-proxy (minimal type)	v1.24.7-minimal-eksbuild.2	v1.23.8-minimal-eksbuild.2	v1.22.11-minimal-eksbuild.2	v1.21.14-minimal-eksbuild.2	v1.20.15-minimal-eksbuild.3	v1.19.16-minimal-eksbuild.3

Minimal build, upstream image

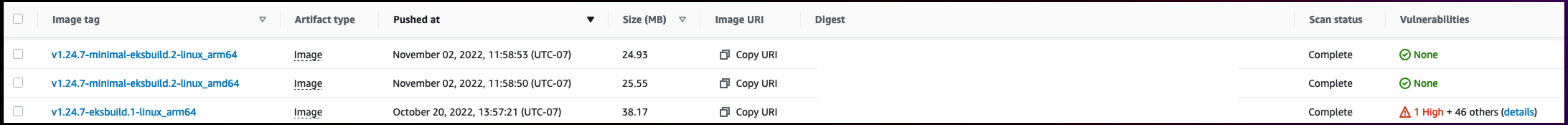


	Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest	Scan status	Vulnerabilities
☐	v1.24.7-minimal-eksbuild.2-linux_arm64	Image	November 02, 2022, 11:58:53 (UTC-07)	24.93	Copy URI		Complete	None
☐	v1.24.7-minimal-eksbuild.2-linux_amd64	Image	November 02, 2022, 11:58:50 (UTC-07)	25.55	Copy URI		Complete	None
☐	v1.24.7-eksbuild.1-linux_arm64	Image	October 20, 2022, 13:57:21 (UTC-07)	38.17	Copy URI		Complete	1 High + 46 others (details)

"Once a problem is described using an appropriate representation, the problem is almost solved."

Patrick Winston

Thank you!

Senthil Kumaran
linkedin.com/in/orsenthil

Please complete the session survey in the **mobile app**