



Container Security

User Guide

July 25, 2023

Copyright 2018-2023 by Qualys, Inc. All Rights Reserved.

Qualys and the Qualys logo are registered trademarks of Qualys, Inc. All other trademarks are the property of their respective owners.

Qualys, Inc.
919 E Hillsdale Blvd
4th Floor
Foster City, CA 94404
1 (650) 801 6100



Table of Contents

About this Guide	5
About Qualys	5
Qualys Support	5
About Container Security Documentation	5
Container Security Overview	6
Concepts and Terminologies	7
What data does Container Security collect?	9
Container Security free version	9
Container Runtime Security	11
Data Retention Policy	11
Get Started	13
Qualys Subscription and Modules required	13
System support	13
Deploying Container Sensor	13
Proxy Support	15
Qualys Platform (POD URL) your hosts need to access	15
POD URL value	15
Sensor network configuration	15
Static scanning of Docker images	16
Users and Permissions	16
Securing Container Assets.....	19
Asset Inventory	19
Unified Dashboard	19
Asset Details	20
Hosts	20
Images	21
Containers	22
Registries	24
Vulnerability scanning of Docker Images	24
On the local host or laptops	25
In the CI/CD pipeline	25
In the Registry	26
In AWS Fargate (ECS)	26
Vulnerability scanning of Docker Containers	28
Vulnerability Scanning of Docker Hosts	28
Registry Scanning	29

Docker host requirements	29
Connectivity	30
How does registry scanning work?	30
Listing Phase	30
Scanning Phase	30
What are the steps?	31
Installing Registry Sensor	31
Adding a new registry to scan	31
Creating a registry scan schedule	34
How to cancel a scan	36
How to restart a scan	36
Viewing vulnerable registry images	36
Defining Vulnerability Exceptions (Beta)	37
Defining Security Policies	38
Sensor Profiles	39
Vulnerability Reporting	41
Create Reports	41
View & Download Reports	42
Delete Reports	43
Compliance Scanning	44
Prerequisites	44
How it works	44
View compliance information	45
SCA Scanning	47
Prerequisites	47
How it works	47
View SCA Scanned Images	48
View Image Details	48
Note about Vulnerability Counts	50
Secret Detection	52
Administration	53
Sensor updates	53
How to uninstall sensor	54

About this Guide

Welcome to Qualys Container Security! We'll help you get acquainted with the Qualys solutions for securing your Container environments like Images, Containers and Docker Hosts using the Qualys Cloud Security Platform.

About Qualys

Qualys, Inc. (NASDAQ: QLYS) is a pioneer and leading provider of cloud-based security and compliance solutions. The Qualys Cloud Platform and its integrated apps help businesses simplify security operations and lower the cost of compliance by delivering critical security intelligence on demand and automating the full spectrum of auditing, compliance and protection for IT systems and web applications.

Founded in 1999, Qualys has established strategic partnerships with leading managed service providers and consulting organizations including Accenture, BT, Cognizant Technology Solutions, Deutsche Telekom, Fujitsu, HCL, HP Enterprise, IBM, Infosys, NTT, Optiv, SecureWorks, Tata Communications, Verizon and Wipro. The company is also founding member of the [Cloud Security Alliance \(CSA\)](#). For more information, please visit www.qualys.com

Qualys Support

Qualys is committed to providing you with the most thorough support. Through online documentation, telephone help, and direct email support, Qualys ensures that your questions will be answered in the fastest time possible. We support you 7 days a week, 24 hours a day. Access online support information at www.qualys.com/support/.

About Container Security Documentation

This document provides information about using the Qualys Container Security UI to monitor vulnerabilities in Images, Containers, and Registries.

For information on deploying the sensor on MAC, CoreOS, and various orchestrators and cloud environments, refer to:

[Qualys Container Sensor Deployment Guide](#)

For information on using the Container Security API, refer to:

[Qualys Container Security API Guide](#)

For information on deploying the sensor in CI/CD environments refer to:

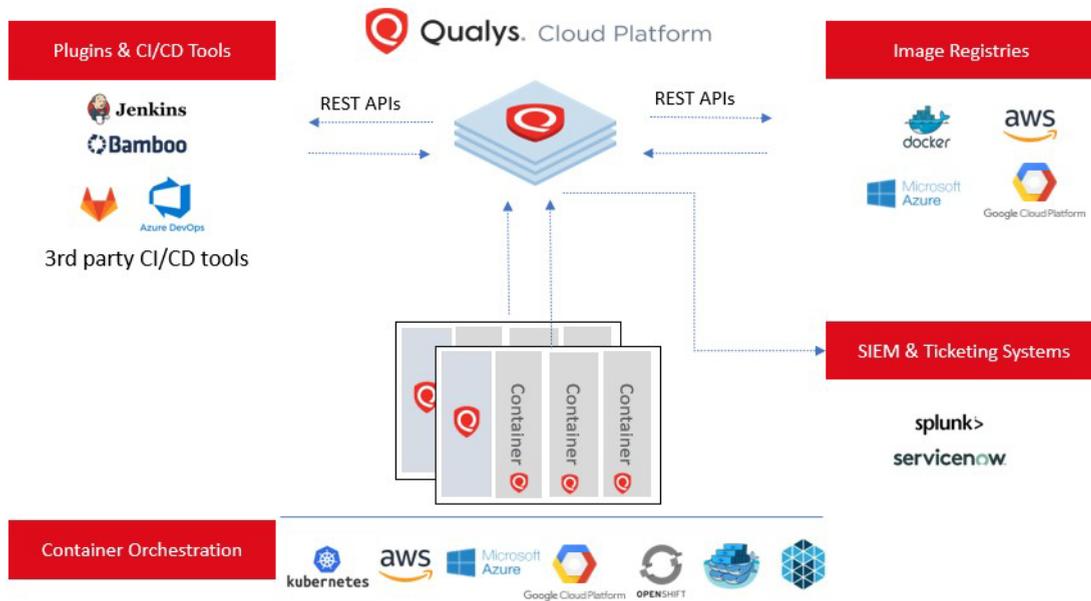
[Qualys Container Scanning Connector for Jenkins](#)

[Qualys Container Scanning Connector for Bamboo](#)

[Qualys Container Scanning Connector for Azure DevOps](#)

Container Security Overview

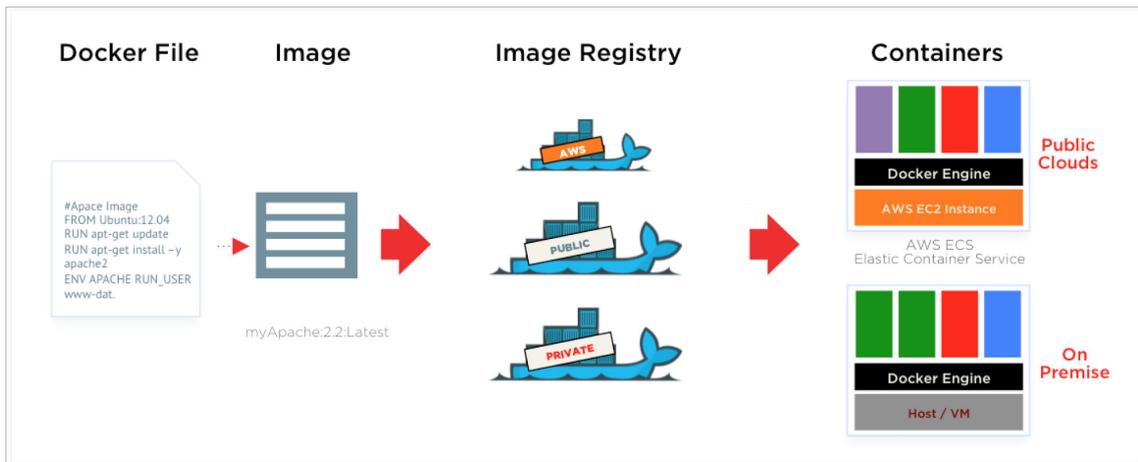
Qualys Container Security provides discovery, tracking, and continuously protecting container environments. This addresses vulnerability management for images and containers in their DevOps pipeline and deployments across cloud and on-premise environments.



With this version, Qualys Container Security supports

- Discovery, inventory, and near-real time tracking of container environments
- Vulnerability analysis for images and containers
- Vulnerability analysis for registries
- Compliance assessment for images and containers
- Integration with CI/CD pipeline using APIs (DevOps flow)
- Uses 'Container Sensor' – providing native container support, distributed as docker image

Concepts and Terminologies



Docker Image

A Docker image is a read-only template. For example, an image could contain an Ubuntu operating system with Apache and your web application installed. Images are used to create Docker containers. Docker provides a simple way to build new images or update existing images, or you can download Docker images that other people have already created. Docker images are the build component of Docker.

An image is a static specification what the container should be in runtime, including the application code inside the container and runtime configuration settings. Docker images contain read-only layers, which means once an image is created it is never modified.

Image is tracked within Qualys Container Security module using Image Id and also a unique identifier generated by Qualys called Image UUID.

Docker Registry

Docker registries hold images. These are public or private stores from which you upload or download images. It serves a huge collection of existing images for your use. These can be images you create yourself or you can use images that others have previously created. Docker registries are the distribution component of Docker. See [Registry Scanning](#) to learn about the public and private registries we support for scanning. For instrumentation support, see [Container Runtime Security](#).

Docker Containers

Docker containers are similar to a directory. A Docker container holds everything that is needed for an application to run. Each container is created from a Docker image. Docker containers can be run, started, stopped, moved, and deleted. Each container is an isolated and secure application platform. Docker containers are the run component of Docker.

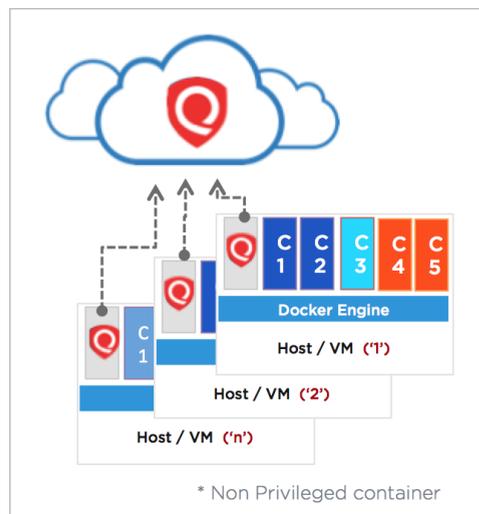
A running Docker container is an instantiation of an image. Containers derived from the same image are identical to each other in terms of their application code and runtime dependencies. But unlike images that are read-only, each running container includes a writable layer (a.k.a. the container layer) on top of the read-only content. Runtime

changes, including any writes and updates to data and files, are saved in the container layer only. Thus multiple concurrent running containers that share the same underlying image may have different container layers.

Containers are tracked within Qualys Container Security module using Container Id and also a unique identifier generated by Qualys called Container UUID.

Docker Host

Hosts or servers running on top of ContainerD, CRI-O and Docker Daemon, and hosting containers and images. Qualys tracks them as Host Assets, collects the metadata including IP address, DNS and other attributes of the Host. A host in Qualys is identified by a unique identifier Host UUID. The UUID is also stored in a marker file under /usr/local/qualys directory by the Agent or a scan with authentication via a Scanner Appliance.



Qualys Container Sensor

Qualys Container Sensor is designed for native support of Docker environments. Sensor is packaged and delivered as a Docker Image. Download the image and deploy it as a Container alongside with other application containers on the host.

The sensor is docker based, can be deployed on hosts in your data center or cloud environments like AWS ECS, Azure Container Service or Google Container Service. Sensor currently is only supported on Linux Operating systems like CentOS, Ubuntu, RHEL, Debian and requires docker daemon of version 1.12 and higher to be available.

Since they are docker based, the sensor can be deployed into orchestration tool environments like Kubernetes, Mesos or Docker Swarm just like any other application container.

Upon installation, the sensor does automatic discovery of Images and Containers on the deployed host, provides a vulnerability analysis of them, and additionally it monitors and reports on the docker related events on the host. The sensor also performs compliance assessments. The sensor container runs in non-privileged mode. It requires a persistent storage for storing and caching files.

Currently, the sensor only scans Images and Containers. To scan Hosts, you would require Qualys Cloud Agents or a scan using Qualys Virtual Scanner Appliance.

Refer to the [Qualys Container Security Sensor Deployment Guide](#) to learn about sensor modes (General, Registry, CI/CD).

What data does Container Security collect?

The Qualys Container Security sensor fetches the following information about Images and Containers in your environment:

Inventory of Images and Containers in your environment from commands such as `docker ps` that lists all containers.

Metadata information about Images and Containers from commands such as `docker inspect` and `docker info` that fetches low level information on docker objects.

Event information about Images and Containers from the docker host for docker events like created, started, killed, push, pull, etc.

Vulnerabilities found on Images and Containers. This is the output of the vulnerability management manifests run for identifying vulnerability information in Images and Containers. This is primarily software package listing, services running, ports, etc. For example, package manager outputs like `rpm -qa`, `npm`. This is supported across various Linux distributions (CentOS, Ubuntu, CoreOS, etc) and across images like Python, NodeJS, Ruby, and so on.

Compliance configurations for OCI compliant images, running containers. We are supporting a subset of controls from CIS Docker benchmarks, which are applicable to running containers and images. Customers can assess configuration risks in their running containers and images and remediate them accordingly based on the Qualys finding. The compliance scans of containers, images will be transparent to customers and will function in a similar real-time cloud native manner like the vulnerability scanning feature.

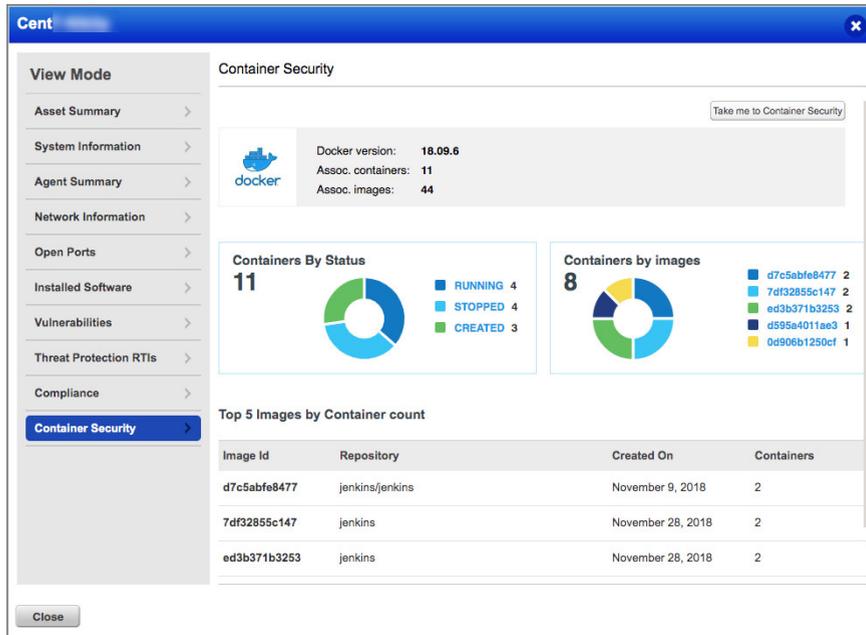
Container Security free version

Qualys has introduced a free version of the Container Security App to enable customers to get a glimpse of what Container Security offers. The free version provides you a view of containers and images in your environment. You must upgrade to a Container Security paid subscription if you want to scan those images and containers for vulnerabilities.

Container Security gets image and container information from either of the following sources if the host contains Docker:

Cloud Agents / Scanners

Cloud Agents installed on hosts or Scanners (via Authenticated Scans) will fetch a list of containers and images present on the host, and provide this information in the AssetView app for each asset under the **Asset Details > Container Security** pane.



Click the **Take me to Container Security** option to enable Container Security free version for your account.

The Container Security app will show metadata of the images and containers but not the vulnerability information. You must upgrade to a paid subscription in order to scan the images and containers for vulnerabilities. See [Hosts](#) to learn more.

Container Sensor

Installing the Container Sensor on hosts will fetch vulnerability information for all official images from Docker Hub, and the first 10 general sensors installed on assets in your account (does not include sensors for CI/CD and registry scanning). Upgrading to a Trial or Full (Paid) subscription will remove this limitation.

API Support

APIs to list Containers, Images and Sensors, and fetch Container, Image, Sensor Details are available for Container Security Free. Upgrade to a paid subscription to get access to all Container Security APIs. Please refer to the [Qualys Container Security API Guide](#).

Container Runtime Security

Container Runtime Security (CRS) provides runtime behavior visibility & enforcement capabilities for running containers. This allows customers to address various use cases for running containers around security best practice enforcement, file access monitoring, network access control.

CRS requires instrumentation of container images with the Qualys Container Runtime Instrumentation, which injects probes into the container image. Customers can configure instrumented images, containers with granular policies which govern container behavior, visibility. Based on these runtime enforcement policies - runtime events, telemetry can be viewed obtained from the backend via UI, API.

CRS is currently supported for Linux OS based containers only.

CRS Documentation

[CRS User Guide](#) | [CRS API Guide](#)

Data Retention Policy

We have implemented a data retention policy for sensors, containers, and events. The data retention policy specifically removes data from Qualys Container Sensor platform when the time period over which the data is retained exceeds the stated data retention policy period.

Data retention periods

Data retention periods are configured with the default values shown in the table below. Please use the API to export the data before it is deleted as per these data retention policies.

Data Type	Default Retention Value (in days)
Sensor	390 (approx. 13 months)
Container	390 (approx. 13 months)
Event (Behavioral)	3
Event (Standard)	7

Why have we implemented a data retention policy?

As we standardize data retention policies across our cloud platform, we are updating the policies for sensors and containers. Having uniform data retention policies ensures the latest data is always available, increases data relevancy while optimizing overall system performance.

Will I be able to access the data once the retention period has elapsed?

No, once the data is purged as per the retention policies, the data cannot be restored. If you have any questions regarding the data retention policy for Container Security, please reach out to Qualys Support.

Can I customize the data retention policy?

Yes, you can customize the retention policy for images, containers, and sensors. To customize the retention policy options, go to **Configurations > General**.

For more information, see [Link](#).

Get Started

This chapter provides an overview of Container Security Sensor installation.

For information on deploying the sensor on MAC, CoreOS, and various orchestrators and cloud environments, refer to the Qualys Container Sensor Deployment Guide.

See [About Container Security Documentation](#)

Qualys Subscription and Modules required

You would require “Container Security” (CS) module enabled for your account. Additionally, in order to get vulnerabilities for the hosts that run the containers, you would need to enable Vulnerability Management (VM), either via Scanner Appliance or Cloud Agent.

System support

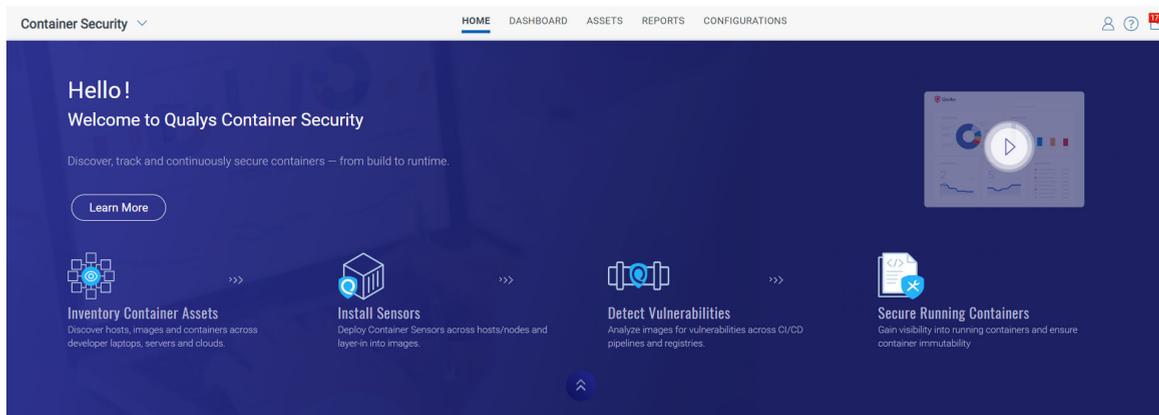
Please refer to the [Qualys Container Security Sensor Deployment Guide](#) for a list of supported systems.

Deploying Container Sensor

IMPORTANT: Sensor deployment is one sensor in one mode on one host/node. Deploying more than one sensor or more than one sensor in another mode is not supported.

Let's get started! Log into your Qualys portal with your user credentials. Select Container Security from the module picker.

As a first time user, you'll land directly into the Getting Started page.



Go to **Configurations > Sensors**, and then click **Download Sensor** to download the sensor tar file. You can see various sensor types:

← Download and Deploy Qualys Container Sensor

 **Download and Deploy Qualys Container Sensor**
Select the environment where you want to deploy the Qualys Container Sensor and follow the installation instructions.

Sensor now supports ARM architecture
Sensor is supported for ARM architecture when downloaded from Docker Hub. Binary installation is not supported for ARM architecture.

General (Host)
Container Runtimes:
Containerd, CRI-O and Docker

Registry
Container Runtimes:
Containerd, CRI-O and Docker

Build (CI/CD)
Container Runtimes:
Docker only

General (Host) Sensor: Scan any host other than registry / build (CI/CD).

Registry Sensor: Scan images in a registry (public / private).

Build (CI/CD) Sensor: Scan images on CI/CD pipeline (Jenkins / Bamboo).

For Registry you need to append the install command with **--registry-sensor** or **-r**

For CI/CD you need to append the install command with **--cicd-deployed-sensor** or **-c**

Installation Instructions



DOCKERHUB BINARY(TAR.XZ)

Installation Steps

- ✓ Run the following commands to install the sensor. The sensor is pre-configured to connect to the Qualys Cloud Platform.

```
sudo docker run -d --restart on-failure -v /var/run/docker.sock:/var/run/docker.sock:ro -v /etc/qualys:/usr/local/qualys/qpq/data/conf/agent-data -v /usr/local/qualys/sensor/data:/usr/local/qualys/qpq/data -e ACTIVATIONID=819dbd9c-afa8-48c0-9af8-c2c2e4a8b8
```



System Requirements & Troubleshooting

System requirements for efficient installation and running of the sensor

Additional Instructions

Download the QualysContainerSensor.tar.xz file and run the commands generated directly from the screen on the docker host. Note the requirements for installing the sensor, the sensor needs a minimum of 1 GB persistent storage on the host.

For information on the “installsensor.sh” script command line parameters, refer to the “Deploying Container Sensor” section in the [Qualys Container Security Sensor Deployment Guide](#).

Proxy Support

The install script asks for proxy configuration. You need to provide the IP Address/FQDN and port number along with the proxy certificate file path. For example,

```
Do you want connection via Proxy [y/N]: y
Enter Https Proxy settings [<IP Address>:<Port #>]: 10.xxx.xx.xx:3xxx
Enter Https Proxy certificate file path: /etc/qualys/cloud-
agent/cert/ca-bundle.crt
```

Your proxy server must provide access to the Qualys Cloud Platform (or the Qualys Private Cloud Platform) over HTTPS port 443. See [Qualys Platform \(POD URL\) your hosts need to access](#) below.

Qualys Platform (POD URL) your hosts need to access

The Qualys URL you use depends on the Qualys platform where your account is located. [Click here to identify your Qualys platform and get the Container Security Server URL](#)

POD URL value

The “Container Security Server URL” for your platform (found at the link above) is the URL you’ll need to provide for the POD_URL variable in Container Security Sensor commands and in configuration yaml files when deploying the sensor.

Sensor network configuration

The sensor is pre-configured with the Qualys URL and the subscription details it needs to communicate to Qualys. In order for the sensor to communicate to Qualys, the network configuration and firewall needs to provide accessibility to Qualys domain over port 443.

After successful installation of the Sensor, the sensor is listed in the Container Security UI under **Configurations > Sensors** where you can see its version, status, etc, and access details. Additionally, you can Download the sensor from the UI.

Static scanning of Docker images

The sensor will perform static scanning for docker images as a fallback mechanism to current dynamic scanning in case docker image does not have a shell. Static scanning will also be performed for Google distroless images without shell. Static scanning will not be performed on Docker container or Docker images having a shell.

Static scanning collects the list of installed software from the Docker image file system to find vulnerabilities in the Docker images. The installed software list is retrieved from the Package manager metadata files. Package managers supported are RPM, DPKG and Alpine.

If you have large images without shell on the host where sensor is running, the requirement for disk space may exceed the minimum requirement of 1GB.

Users and Permissions

The Qualys Container Security application uses a Role Based Access Control (RBAC) model to control access to Container Security features. With RBAC, each user is assigned a pre-defined user role which determines which actions the user can take in the UI and API.

About User Roles

A **Manager** user (superuser with full permissions and scope) can access the Administration utility, has all roles assigned, can add and manage users, can create custom roles and assign roles to users. The first user in a new customer subscription is a Manager user.

We have the following pre-defined roles for Container Security. These roles are exclusive to the Container Security module. The roles defined in other modules have NO correlation with those defined in Container Security.

CS Manager: A CS Manager has all Container Security permissions and can perform all actions in the Container Security UI and API. All Container Security users existing prior to the Container Security 1.17 release will be assigned the CS Manager role automatically, which means they can perform all actions.

CS User: The CS User role only has permission to access the Container Security UI, and has no other permissions assigned.

Note: This role will not be available in new customer subscriptions created after Container Security 1.17.

How to View Roles and Permissions

Managers can view user roles and permissions from the Administration utility. If you need help at any time, please refer to the [Qualys Administration Utility Help](#).

How to Remove Permissions from an Existing User

All users existing prior to the Container Security 1.17 release will automatically get the “CS Manager” role which gives them all Container Security permissions. If you want to limit the permissions for a particular user, then you’ll need to create a custom role and select only the permissions the user should be granted.

Edit the user account from the Users > User Management tab in the Administration utility. Remove the “CS Manager” role from the user since this gives the user all permissions, and assign the new custom role to the user.

How to Add New Users

Any Manager can add new users and assign them roles and permissions. You can add users from the Administration utility. Before you begin, think about which roles and permissions you want to grant to the new user. See [Users and Permissions](#) for more details.

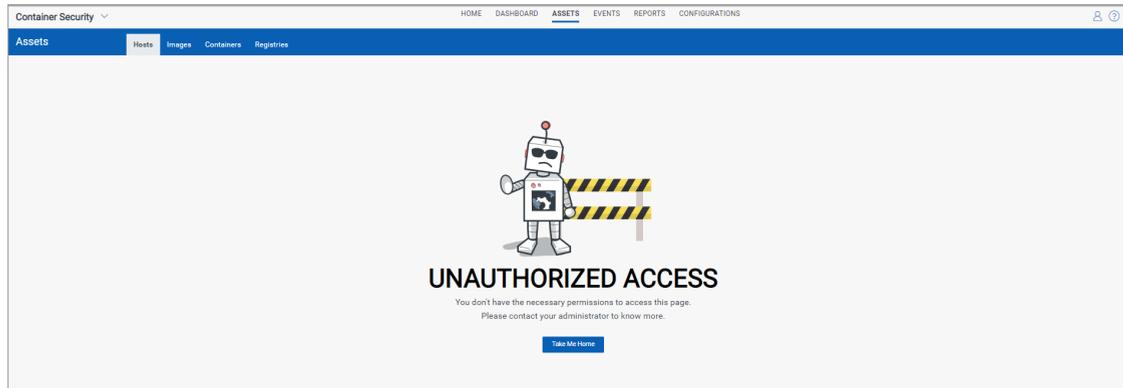
Follow these steps to add a user:

- 1) Choose **Administration** under **Utilities** in the application picker.
- 2) Go to the **Users > User Management** tab.
- 3) From the **Create User** menu, choose one of the following options:
 - **Create Reader User** – The user will be assigned the following roles automatically: VM User, Reader, Reporting Reader. The user will not be assigned any Container Security roles/permissions automatically. You’ll need to edit the user account to add CS roles.
 - **Create Manager User** – The user will have all roles assigned, full permissions and scope. Manager users have access to the Administration utility.
- 4) Define the user settings. For help with settings, click the **Launch Help** link in the upper right corner. Once you've added the user, we'll send them a welcome email with login instructions.
- 5) For a non-Manager user, you’ll need to edit the user's settings to assign the user Container Security roles and permissions. From the **User Management** tab, choose **Edit** from the **Quick Actions** menu. Go to the **Roles and Scopes** tab to assign roles that you’ve already defined.

When a User Does Not Have Permission to Perform an Action

If a user is not granted a particular permission then the user will not be able to perform the related action from the UI or API.

When a user does not have the List permission for an object, then the user will not be able to view the related data list in the UI or fetch the list from the API. In the UI, you'll see an **UNAUTHORIZED ACCESS** message when you do not have permission to view the list. In the example below, the user does not have the List Hosts permission.



If the user has the List permission but does not have other permissions like Create, Update, and Delete, then the list will be visible to the user, but the button or menu option for the action will not be visible. For example, if the user does not have the Create Registry permission then the user will not see the New Registry button and will not be able to create registries from the API.

Securing Container Assets

Asset Inventory

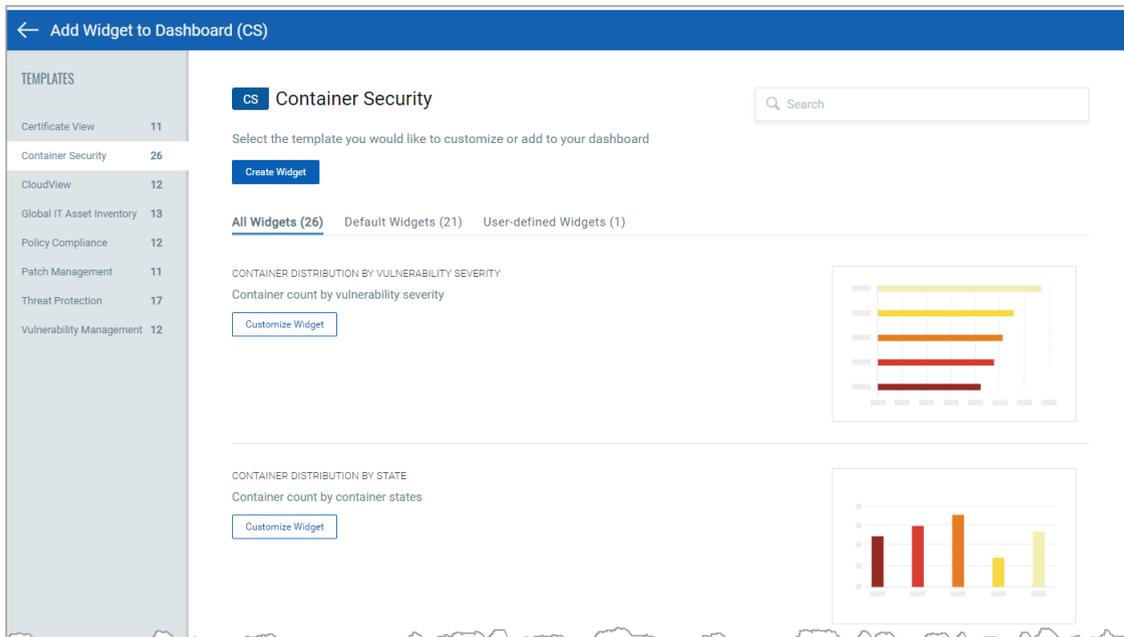
Upon installation of the sensor, it automatically scans the host for the images and containers that are present on the host. The inventory and the metadata of the inventory is pushed to Qualys portal.

Unified Dashboard

Dashboards help you visualize your container environment assets, see your threat exposure, leverage saved searches, and fix priority of vulnerabilities quickly.

We have integrated Unified Dashboard (UD) with Container Security. UD brings information from all Qualys applications into a single place for visualization. UD provides a powerful new dashboarding framework along with platform service that will be consumed and used by all other products to enhance the existing dashboard capabilities.

You can use the default Container Security dashboard provided by Qualys or easily configure widgets to pull information from other modules/applications and add them to your dashboard. You can also add as many dashboards as you like to customize your view. For help creating widgets, dashboards, templates and more, please refer to the [Unified Dashboard online help](#).



Asset Details

The Assets section lists the Images and Containers discovered along with their metadata information like ports, networks, services, users, installed software, etc. The assets are listed along with their associations like associated containers and hosts for an image, other containers from the same parent image. Users can search for images and containers based on their attributes.

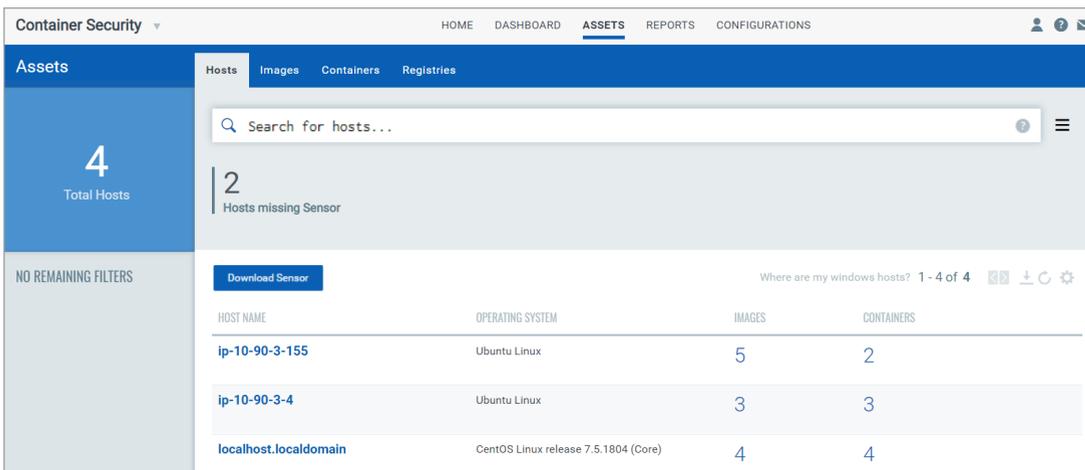
Jump to a section: [Hosts](#) | [Images](#) | [Containers](#) | [Registries](#)

Hosts

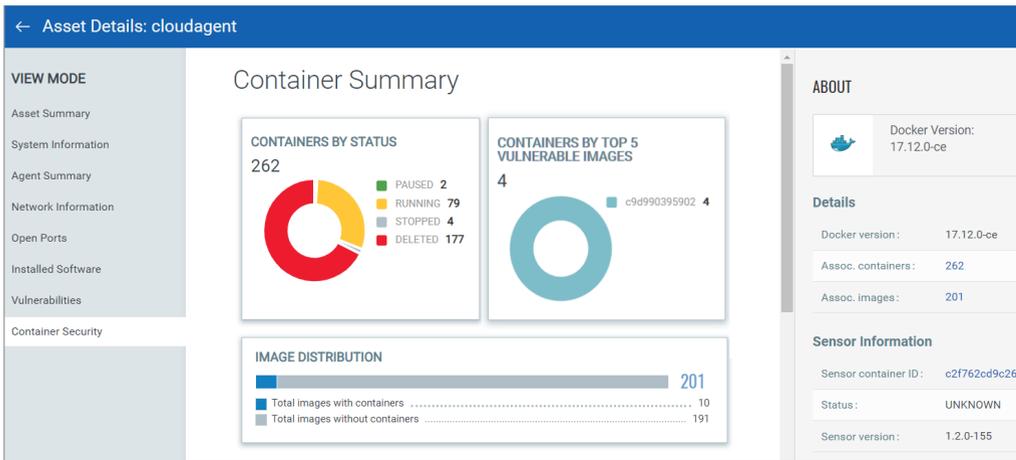
The **Assets > Hosts** tab shows container hosts discovered, scanned by the Qualys Cloud Agent and/or Qualys Network Scanner. Currently, container hosts discovered, scanned only by the Qualys Container Sensor are not shown in this list. It is recommended you use the Images or Containers tabs for these. Additionally, Qualys Container sensors currently only support hosts and clusters with Linux-based host OSes and Mac OS.

For each host in the list, you'll see the image and container count. Image and container details can be viewed in their respective tabs.

Use QQL search tokens to search for hosts. See the [online help](#) for a list of search tokens.



Access the details page for a host from the Sensor details page. Asset Details view displays information about the host on which the sensor is deployed. Besides system, network, and port information, the Asset Details view also displays a list of software installed on the host, vulnerabilities present, certificates, and Threat Protection RTIs (when Qualys TP app is enabled). Container Security panel shows all containers installed on the host, their status, and the images from which the containers are spawned.



Images

The **Assets > Images** tab shows the discovered images along with their metadata information. Use QQL search tokens to search for images. See the [online help](#) for a list of search tokens.

Container Security - HOME DASHBOARD **ASSETS** REPORTS CONFIGURATIONS

Assets Hosts Images Containers Registries

96 Total Images

- 2 Images detected without CS Sensor
- 61 Images with Sev 5, 4 Vulnerabilities
- 0 Docker Hub Official Images
- 18 Images not Compliant

REGISTRY	REPOSITORY	CREATED ON	TAGS	CONTAINERS	VULNERABILITIES	COMPLIANCE
registry-1.docker.io	image_1 Image ID: 4b72a9a397b0	Mar 15, 2021	registrycheck-lat...	0 On Hosts: 0	182	-
registry-1.docker.io	image_2 Image ID: 7fb6194d019	Mar 15, 2021	distroless-java-8...	0 On Hosts: 0	0	-
docker.io	image_abc Image ID: be249b1ccc35	Mar 15, 2021	latest	1 On Hosts: 1	213	2
docker.io	image_xyz Image ID: 468de79f8e88	Mar 15, 2021	latest	1 On Hosts: 1	4	2
registry-1.docker.io	my_image Image ID: 3e8e8af135a0	Mar 14, 2021	distroless-java11...	0 On Hosts: 0	-	-

Select **Add Tags** from the Quick Actions menu to assign static asset tags to images. You have the option to create new tags while adding them. You can also choose to pass on the assigned tags to containers that are associated with the selected images.

Select **View Details** from the Quick Actions menu for any image in the list to get comprehensive information about the image. You can view detailed information about the image, its associations with containers, drift containers, and hosts.

- The Installed Software section displays software having vulnerabilities, and for which fixes (patches) are available.

- The Vulnerabilities section provides vulnerability information, such as confirmed and potential vulnerabilities with their severity. For each vulnerability you'll see the vulnerability age (in days). Age is calculated from the point Qualys published the vulnerability.
- The Compliance section provides a list of controls that were scanned with control details (CID, criticality, statement, category, technologies).
- The Layers section displays a list of layers the image is made of.

The screenshot shows the 'Image Details: image_abc' page. The 'Summary' section includes metadata: Tag: latest, Size: 801.31 MB, Registry Name: docker.io, Repository Name: image_abc, DockerHub: -, and Scan Type: Dynamic. The 'Vulnerabilities' section shows 213 total vulnerabilities, with 100% confirmed and 0% potential. The 'Compliance' section shows 2 total controls, with 0% passed and 100% failed. The 'Associated Containers' section shows 1 total container, with 100% running, 0% stopped, and 0% paused.

Containers

The **Assets > Containers** tab shows the discovered containers along with their metadata information. Use [QQL](#) search tokens to search for containers. See the [online help](#) for a list of search tokens.

The screenshot shows the 'Container Security' dashboard. The 'Assets' section displays 24 total containers, with 19 root containers, 0 privileged containers, 1 container detected without CS Sensor, 4 containers in drift, and 16 containers not compliant. The main table lists containers with the following columns: Container, Created On, Host, State, Last Scanned, Vulnerabilities, and Compliance. The table shows 10 containers, including 'container_1', 'container_2', 'container_3', 'container_abc', 'container_xyz', 'my_container', 'sample_container', and 'sample2_container'.

CONTAINER	CREATED ON	HOST	STATE	LAST SCANNED	VULNERABILITIES	COMPLIANCE
Container id: 6b0add73afe7	Mar 15, 2021	--	RUNNING 2 days ago	--	--	--
container_1 Container id: e5a288061bbf	Mar 14, 2021	dockercent 10.115.98.192	RUNNING 2 days ago	8 hours ago	212	24
container_2 Container id: 1f42cd3725f4	Mar 14, 2021	dockercent 10.115.98.192	RUNNING 2 days ago	8 hours ago	212	24
container_3 Container id: 9f8f84e250f6b	Mar 14, 2021	ip-10-82-9-192 10.82.9.192	RUNNING 3 days ago	3 days ago	4	24
container_abc Container id: 9f0979eca794	Mar 12, 2021	ip-10-82-9-192 10.82.9.192	RUNNING 5 days ago	3 days ago	4	24
container_xyz Container id: 0202e3d22215	Mar 12, 2021	ip-10-82-9-192 10.82.9.192	RUNNING 3 days ago	3 days ago	4	24
my_container Container id: 3e762442295f	Mar 12, 2021	ip-10-82-9-192 10.82.9.192	RUNNING 5 days ago	3 days ago	213	24
sample_container Container id: 6f6c4c5bd265	Feb 24, 2021	localhost localdomain 10.115.119.175	RUNNING 21 days ago	20 days ago	0	24
sample2_container Container id: e7f8bdeed7ac	Feb 24, 2021	localhost localdomain 10.115.119.175	RUNNING 21 days ago	20 days ago	0	--

Select **Add Tags** from the Quick Actions menu to assign static asset tags to containers. You can also create new tags on the fly while assigning them.

Select **View Details** from the Quick Actions menu for any container in the list to get comprehensive information about the container. You'll get detailed information about the container, its associations with an image, drift containers, and hosts.

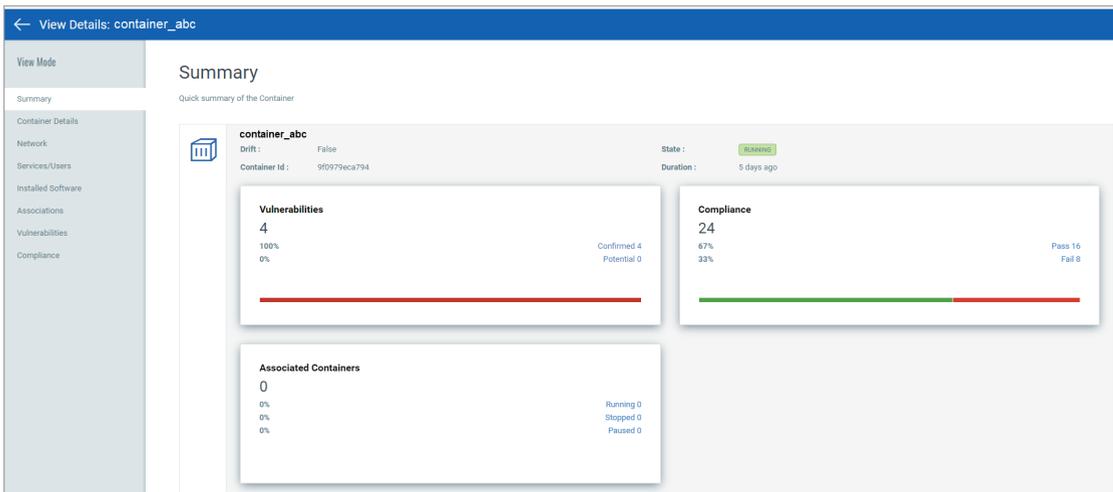
- Container "State" is updated based on the docker events (exec_start, kill, destroy, stop) that Qualys Sensor reports to Qualys Cloud Platform.

- The Services/Users section displays the list of services available in the container and users associated with the container.

- The Installed Software section displays software having vulnerabilities, and for which fixes (patches) are available.

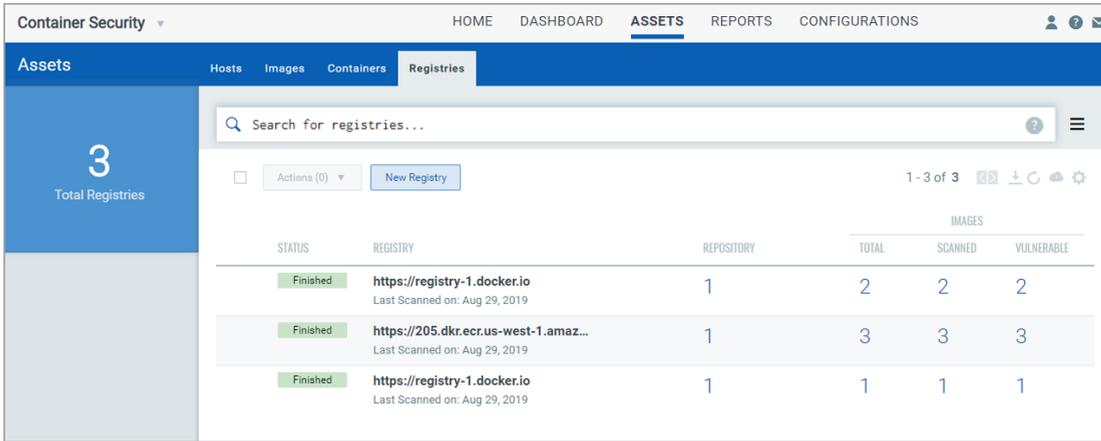
- The Vulnerabilities section provides vulnerability information, such as confirmed and potential vulnerabilities with their severity. For each vulnerability you'll see the vulnerability age (in days). Age is calculated from the point Qualys published the vulnerability.

- The Compliance provides a list of controls that were scanned with control details (CID, criticality, statement, category, technologies).

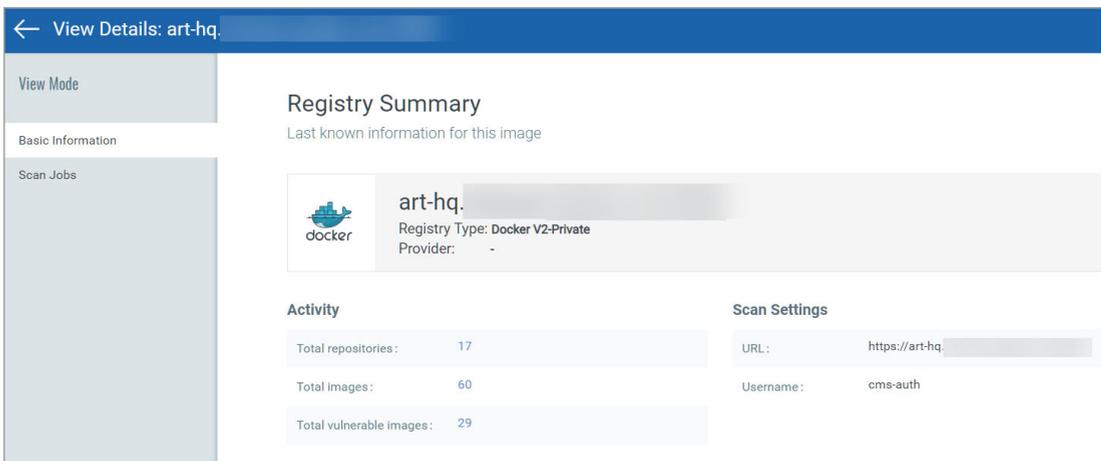


Registries

The **Assets > Registries** tab shows the registries in your account. Use QQL search tokens to search for registries. See the [online help](#) for a list of search tokens.



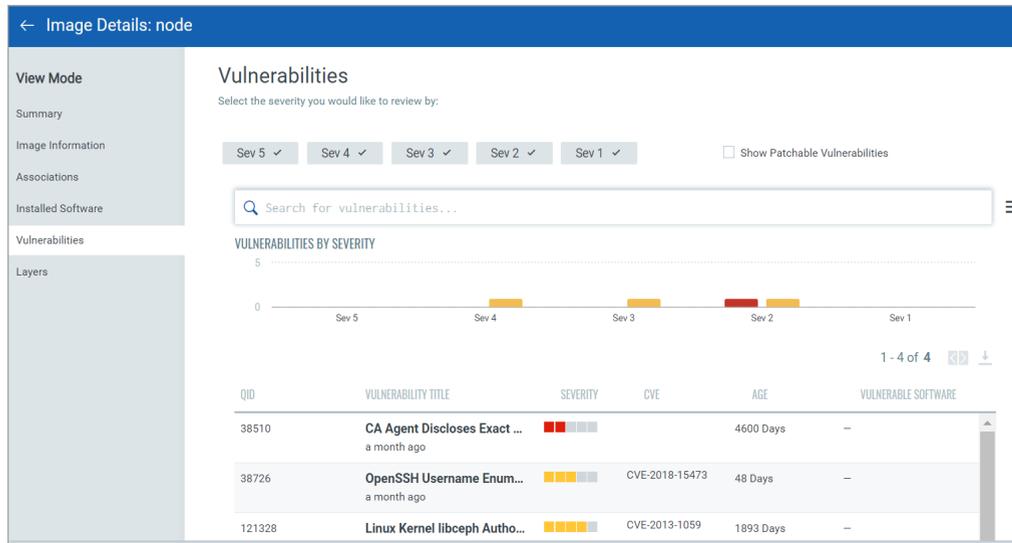
Select **View Details** from the Quick Actions menu for any registry in the list to get comprehensive information about the registry. You can view detailed information about the registry: number of repositories, total number of images and number of vulnerable images within that registry. The Scan Jobs panel lists the On Demand and Automatic Jobs created for that registry. For more information, see [Registry Scanning](#).



Vulnerability scanning of Docker Images

The docker images are scanned to check the presence of any vulnerabilities by the Qualys container sensor. The vulnerabilities panel in **Image Details** provides a list of vulnerabilities with Severity along with their QIDs. Select **Show Patchable Vulnerabilities** to view vulnerabilities with available patches.

Qualys scans the docker images for vulnerabilities not through static analysis but via a non-static method, where it looks at the Image as a complete entity. This process is more effective and has lesser false positives (FP) than the more commonly used Static Analysis.



Docker Images are found distributed across the environment from developer laptops, build systems, Image Registry to being cached on the docker hosts running Containers. To scan for vulnerabilities you would need the Container Sensor deployed on the host asset.

To get an inventory of the images and scan them for vulnerabilities, deploy the container sensor on the host. Refer to [Deploying Container Sensor](#) for the install instructions and system requirements.

On the local host or laptops

To get an inventory of the images and scan them for vulnerabilities, deploy the container sensor on the local host. Refer to [Deploying Container Sensor](#) for the install instructions and system requirements

To deploy the Sensor on the Mac laptops, there are additional install steps - follow the instructions in the Qualys Container Security Sensor Deployment Guide. See [About Container Security Documentation](#).

Upon Installation the sensor automatically detects the images, and provides -inventory and vulnerability scans of the image.

In the CI/CD pipeline

Doing a complete check of vulnerabilities in an image during the build time ensures a lot cleaner operating environment. Qualys Container Security provides a plugin for Jenkins and Bamboo to get the vulnerability analysis of images in the build environment. If you are using other tools you can use the REST APIs available to perform vulnerability analysis on the images.

To start, deploy the Container Sensor on the Build host where the images are being created. The sensor upon install would automatically trigger a vulnerability analysis of the new images found. Use the API or the plug-in to look for vulnerabilities in the Images. If you are in Jenkins or Bamboo environment, the plug-in would provide detail list of the vulnerabilities and its details directly within the plug-in, you could optionally access your Qualys subscription to view the full report.

In the Registry

Currently, the Qualys Container Sensor doesn't automatically poll or pull images to do an analysis. Rather you would need to deploy the sensor on the host that is configured to pull images from the registry. Either manually or via a cron pull the new images to the host. The sensor does an automatic analysis as soon as it finds a new image. Use the APIs or the Qualys portal to query for the vulnerabilities identified.

In AWS Fargate (ECS)

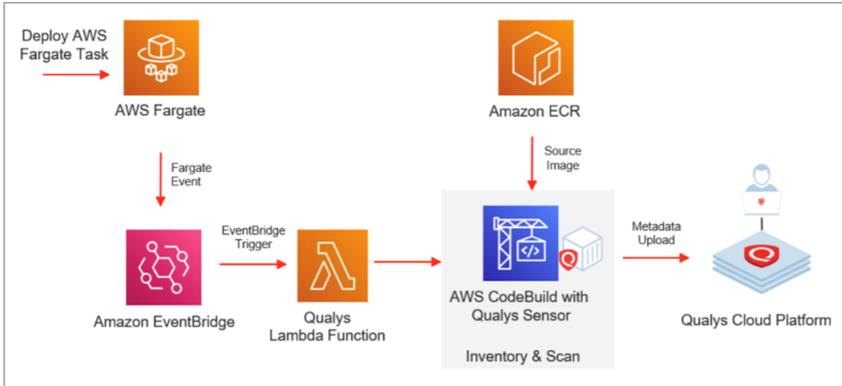
Qualys Container Security can be used to secure AWS Fargate. AWS Fargate is a serverless compute engine for containers that works with Amazon Elastic Container Service (ECS). This feature allows you to know the containers running on AWS Fargate, perform vulnerability and compliance scanning on container images launched by Amazon Fargate tasks (ECS), and view the findings to take remediation actions.

Since AWS Fargate is serverless, the solution launches a sensor whenever a new Fargate task is being deployed. We will use AWS CloudFormation and a Qualys Lambda function to trigger scanning automatically. You'll configure a CloudFormation template with your subscription details and a Qualys Lambda function with the Qualys S3 bucket name & S3 bucket key to trigger image scanning of images pulled from Amazon Elastic Container Registry (ECR).

How it Works

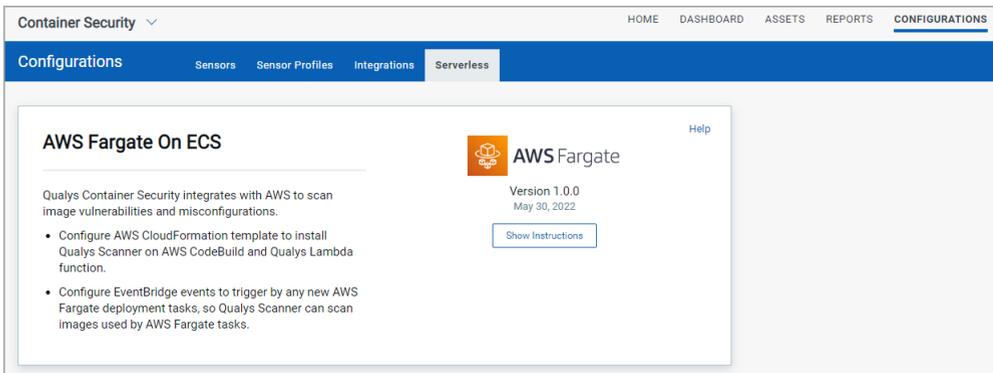
We support scanning Docker images pulled from Amazon Elastic Container Registry (Amazon ECR) with x86_64 architecture. When an AWS ECS Fargate task is launched, the AWS EventBridge rule created during Qualys deployment consumes the event. The EventBridge rule is set in such a way that it triggers the Qualys scanning Lambda function. The Qualys Lambda function then processes the event received from EventBridge to decide on image scanning. The Qualys Lambda function launches the AWS CodeBuild to run the Qualys sensor, which pulls the image from Amazon ECR and then performs the

vulnerability and compliance scan on the image. After a successful image scan, image metadata gets uploaded to the Qualys Cloud Platform for evaluation, and users can view details from the Container Security UI & API.



Serverless Configuration

Go to the **Serverless** tab under **Configurations**. Click the **Show Instructions** button to open the [Qualys Container Sensor Deployment Guide](#) for configuration steps. After you complete the one-time configuration, all images deployed from Amazon ECS tasks in AWS Fargate will be scanned automatically and results will be uploaded to your account.



Vulnerability scanning of Docker Containers

The containers are scanned to check the presence of any vulnerabilities within the containers. The Vulnerabilities panel in Container Details provides a list of vulnerabilities with Severity along with their QIDs. Select **Show Patchable Vulnerabilities** to view vulnerabilities with available patches.

The screenshot shows the 'Vulnerabilities' section of a container's details. It includes a search bar, a severity filter (All, Rogue, Sev 5, Sev 4, Sev 3, Sev 2, Sev 1), and a 'Show Patchable Vulnerabilities' checkbox. Below the filter is a bar chart titled 'VULNERABILITIES BY SEVERITY' showing counts for Sev 5, Sev 4, Sev 3, Sev 2, and Sev 1. A table below the chart lists vulnerabilities with columns for QID, Vulnerability Title, Severity, CVE, Age, and Vulnerable Software.

QID	VULNERABILITY TITLE	SEVERITY	CVE	AGE	VULNERABLE SOFTWARE
370845	Linux Kernel 'drivers/scsi/libsas/sas...	Sev 3	CVE-2018-7757	182 Days	-
38510	CA Agent Discloses Exact Operating...	Sev 2		4605 Days	-

Good to know

Drift Containers are those which contain vulnerabilities or software, not found in the image from which the container is spawned.

Rogue Vulnerabilities are classified as either New, Fixed or Varied. New are those which are newly found on the containers, but were not present in the image from which the container is spawned. Fixed, are the vulnerabilities that are not found in the container but in the image. Varied, are the vulnerabilities that are found in both Containers and Images but the detection varies between them.

Rogue Software are classified as new or removed. New, software which are found in the Container but not in the image from which the container is spawned. Fixed, Software not seen in the Container but is present in the parent Image.

Vulnerability Scanning of Docker Hosts

Container Security Sensor scans Images and Containers for vulnerabilities, and not the actual host machine. You can scan the host via Scanner Appliance or Cloud Agent. Configurations required on the host for using the Cloud Agent are independent of the Sensor. For example, proxy configuration.

Registry Scanning

Using Qualys Container Security you can scan public and private registries. Public registries are cloud accessible registries hosted on Amazon, Azure and Google. While, private registries are on premise registries deployed on a private network such as those hosted using Artifactory or Nexus. Qualys supports scanning only authenticated registries. Note: Currently you can only scan V2 type of registries with Qualys Container Security. We support scanning the following registries:

Public registries: Docker Hub, AWS ECR, Google Cloud Registry (GCR), Google Artifact Registry, Azure Container Registry (ACR)

Private registries: v2-private registry

- Docker Private Registry: insecure (http), secure (auth + https)
- Docker Trusted Registry
- Harbor
- JFrog Artifactory Private
- Mirantis Secure Registry (MSR) 2.9.4+
- OpenShift Container Registry (OCR)
- RedHat Quay
- Sonatype Nexus

Note: Using http requires customers to manually configure their docker-engine for the registry. Qualys does not recommend using http and it's intended more for testing in dev environments.

For details on the sensor versions supported for each registry type and interoperability with 3rd party solutions, refer to the [Qualys Container Security Interoperability Matrix](#).

For instrumentation support, see [Container Runtime Security](#).

Docker host requirements

As a prerequisite, you must install the registry sensor on a docker host (with Docker, Containerd or CRI-O Runtime) which has access to the registry to pull images to scan.

Docker version: 1.12 or later

Disk space on docker host: Minimum 20 GB of free space on the partition where docker is installed. This is required to scan registry images. Additionally, 1 GB of free space is required for persistent storage.

Connectivity

The registry sensor host should have connectivity to the registry to be scanned. If runtime is Docker, you can validate connectivity by performing a successful docker login from the host to the registry. If runtime is Containerd or CRI-O, you can validate connectivity by trying to pull any image from the registry.

Docker Runtime:

```
docker login <registryurl> (No protocol)
```

For Example:

```
docker login myregistry.com:5001
```

Containerd/CRI-O Runtime:

```
crictl pull anyimage from registry
```

How does registry scanning work?

Registry scanning is divided into two phases: Listing phase and Scanning phase.

Listing Phase

In the Listing phase, the Container Security sensor calls Docker Registry v2 APIs to collect all the image metadata information for the repository provided in the registry scan schedule.

Qualys sensor makes catalog, tag, manifest and config API calls to collect information and this information is displayed on the UI. Based on the filters defined in the schedule by the user (e.g., scan images created in last 14 days), the images are queued for scanning.

Note - For public registries (cloud accessible), Qualys makes the Docker Registry API calls and fetches information to feed the sensors for performing an image scan. In case of private registries, as Qualys cannot connect to them, the sensor performs both listing and scanning actions and sends information to Qualys.

Scanning Phase

Sensors which are provisioned as registry sensors, poll Qualys periodically to see if any images are queued for scanning. Qualys assigns only a subset of discovered images to the sensor for scanning. The response payload includes image details along with authentication credentials required to pull image from the registry.

Qualys Registry Sensor pulls these images from the registry and gathers and pushes the information (snapshot) to Qualys Cloud. Qualys then runs signatures on the collected information and generates a vulnerability report which can be viewed on the Container Security UI. If the repository has a lot of images to scan, the overall scanning time might be longer than usual. You can install multiple registry sensors to distribute the scanning payload to reduce the scan time and view the results faster.

What are the steps?

From the Container Security UI, you'll download the sensor image and deploy the sensor as a registry sensor in the network where the sensor can communicate with the registry and Qualys. Then, create a new registry and set up a scanning schedule on the repository that you need the security posture of. You can perform an on-demand or a scheduled scan. As Scheduled scans are incremental, only the new images that are added to the configured repository since the last scan will be considered.

We'll describe these steps in more detail:

[Installing Registry Sensor](#)

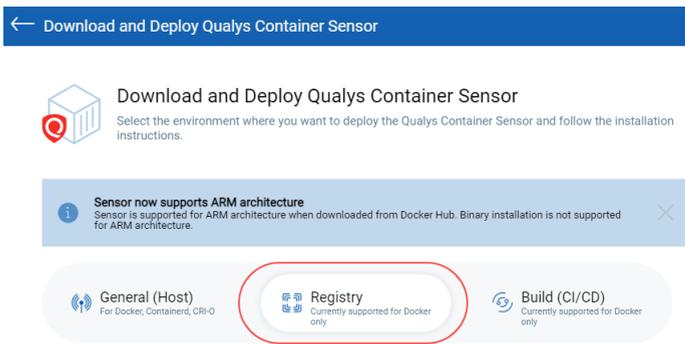
[Adding a new registry to scan](#)

[Creating a registry scan schedule](#)

[Viewing vulnerable registry images](#)

Installing Registry Sensor

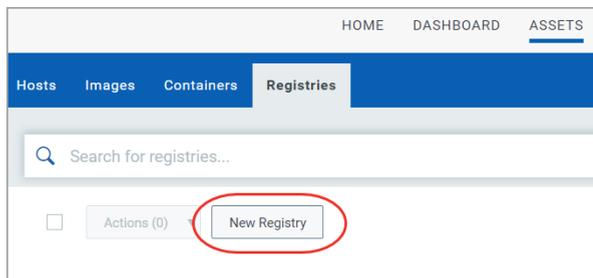
Download the Registry sensor. Go to **Configurations > Sensors**, click **Download Sensor** and then click **Registry**.



You'll need to append **--registry-sensor** or **-r** to the sensor install command to install the sensor for registry scan.

Adding a new registry to scan

You must add a registry in order to scan it. Go to **Assets > Registries**, and click **New Registry**. Make sure the registry sensor deployed on the docker host is in Running state.



In order to perform vulnerability and compliance analysis, you'll need to connect to the registries using registry authentication. Different types of authentication are needed to connect to different types of registries.

Registry authentication types are Token, BasicAuth, DockerHub, AWS.

Note: Token authentication is used by the sensor host while connecting to the registry if the registry supports token-based authentication.

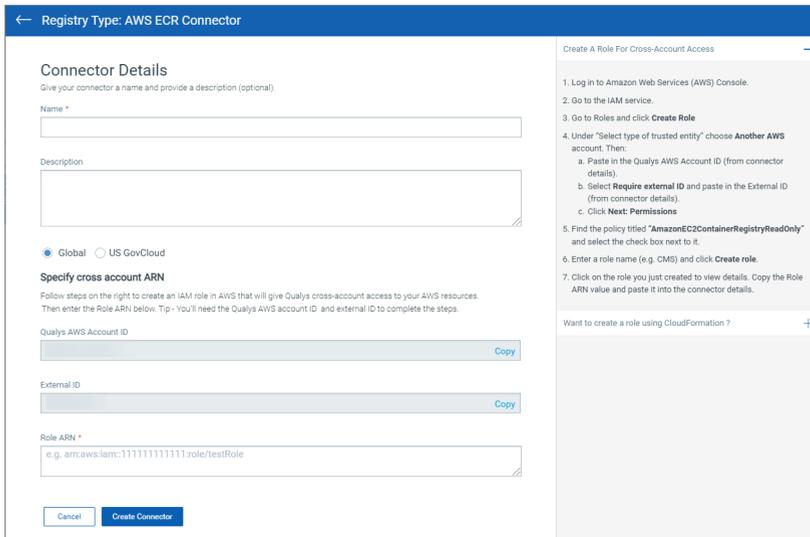
The following table lists the privileges required for authentication for different private registries:

Registry	Authentication Privileges Required	Description
JFrog Artifactory Private	Any user	Authenticate using either of the following: - Credentials of any user account - An access token
Mirantis Secure Registry (MSR) 2.9.4+	Administrator	Enter the credentials of an administrator account. Mirantis Secure Registry supports token-based authentication.
OpenShift Container Registry (OCR)	Registry-Viewer	Enter the service account credentials. The registry-viewer role must be associated with the service account.
RedHat Quay	Administrator or Super user	Enter the credentials for any of the following accounts: - An account with administrator or super user privileges - A robot account For the robot account, the username is formatted as <code>UserName+RobotAccountName</code> and the password is the password token value for the robot account.

Registry	Authentication Privileges Required	Description
Harbor Registry	Administrator	Enter the credentials of an administrator account. If your Harbor registry version supports token-based authentication, the sensor will perform the V2 catalog call with the authentication token. If authentication fails, the sensor will automatically fall back to the basic authentication method for the V2 catalog call.
Sonatype Nexus	Administrator	Enter the credentials of an administrator account.
Docker Trusted Registry	Any user	Enter the credentials of any user account.
Docker Private Registry: Secure and Insecure	Any user	Enter the credentials of any user account.

For public registries, a role with reader privileges is sufficient to connect to the registries and access the resources.

For AWS ECR, you can create a connector to connect to your AWS Global or US GovCloud account. If you selected a standard AWS region, then pick the Global account type in connector details. If you selected a US GovCloud region, then you must pick the US GovCloud account type in connector details.



Note: Currently, the registry sensor can only scan AWS ECR Private repositories.

For GCR (Google Cloud Registry), you can create a connector to connect to your GCP account.

← Add GCR Connector

Connector Details
Give your connector a name and provide a description (optional).

Name Required

Description

Authentication Details
Configuration File Required

Drop file here to attach or [browse](#)

Cancel Create

Enable access to some API's in API library +

Create service account and download configuration file -

1. Login to the GCP console and select a project.
2. From the left sidebar, navigate to **IAM & admin > Service accounts** and click **CREATE SERVICE ACCOUNT**. Provide a name and description (optional) for the service account and click **CREATE**.
3. Choose **Viewer** and **Security Reviewer** role to assign at least reader permissions to the service account and click **CONTINUE**.
4. Click **CREATE KEY**. Select **JSON** as **Key type** and click **CREATE**. A message saying "Private key saved to your computer" is displayed and the JSON file is downloaded to your computer. Click **CLOSE** and then click **DONE**.

Upload the configuration (JSON) file to complete GCP connector creation in Qualys Cloud Platform.

Assinging Service Principal +

For ACR (Azure Container Registry), create a connector to connect to your Azure account.

← Registry Type: Azure Container Registry Connector

Connector Details
Give your connector a name and provide a description (optional).

Name Required

Description

Application ID Required

Client Secrets Required

Cancel Create Connector

Create Application and get Application Id & Client Secret -

Create Application in Azure Active Directory and you can then note the **Application ID** and generate the **client secret**.

1. Log on to **Microsoft Azure portal**, navigate to **Azure Active Directory** then to **App Registrations**.
2. Click on **New Registration** and provide the following details:
 - a. **Name**: A name for the application.
 - b. **Supported account types**: Single Tenant and Accounts in this organizational directory only.
3. Click on **Register**.
4. Copy the **Application (client) ID**.
5. Navigate to the **Certificates & secrets** on the left panel then generate client secret by clicking on **New Client Secret**, provide the following details:
 - a. **Description**: A description of the cleint secret.
 - b. **Expires**: Never.
 - c. Click on **Add**.
 - d. Copy the Client secret that is generated.

Assinging Service Principal +

Creating a registry scan schedule

After providing registry information, move on to Step 2 to provide scan settings.

Scan Type

You can choose to scan immediately (On Demand) or on an on-going basis (Automatic). On Demand scan allows you to scan repositories as well as specific images within those repositories (use date and tag filters). With Automatic scan, you can scan entire repositories on a recurring basis following a user-specified scan schedule.

Repository

Add one or more repositories to scan. In the **Repository** field, enter the full repository path up to the last sub-directory containing the images you want to scan. Tip: The following command helps you to get a list of full repository names that are part of a registry.

```
curl -u <username>:<password> https://<registry-url>/v2/_catalog
```

Notes:

- For Google Cloud Registry, the repository name should not include location information since you already provided the location under registry information. For example, the repository name should be: project-Id/repository-name
- For Google Artifact Registry, only the repository name is needed. We'll auto populate the full path.

Using Filters (for On Demand Scans)

When the scan type is On Demand, you'll see filters that allow you to select specific images within the repository to scan.

By Date - Filter the list of images based on when the image was created. Select one of the options on the **Created Date** menu for the number of days, weeks or months ago the image was created.

By Tags - Filter the list of images to scan within the repository by selecting tags assigned to those images. Enter a single tag name and click **Add**. Then enter another tag name and click **Add**, and so on.

Using JFrog Artifactory Private registry? In this case you'll need to select images by tag name. You can further filter images by the image pushed date.

Pushed Date - This option allows you to filter the images to be scanned based on when each image was pushed into the repository being scanned. Choose "All" to scan all images pushed into the repository regardless of the pushed date or "Custom Days" to only scan images pushed into the repository a set number of days ago that you specify.

Scan Schedule (for Automatic Scans)

Configure how often an Automatic registry scan job will run – daily or weekly. Choose an option from the Recurrence menu under Scan Schedule.

For daily scans, select the time of day you want the scan to start from the Start Time menu. The scan starts every day at the selected time.

For weekly scans, select a day of the week and the start time. The scan happens every week on the specified day and time.

Scan all images: You can select the **Scan all images** option to scan all images in a registry every time the registry scan is launched. You need to get this feature enabled for your subscription. Contact your Technical Account Manager or Qualys support to enable it.

How to cancel a scan

You can cancel an ongoing scan by editing the registry and then using the **Cancel** option from the Quick Actions menu of a scan job. You cannot cancel jobs which are in “Error” or “Finished” state.

How to restart a scan

Use the **Rescan** option to restart an On Demand scan. You cannot restart scan jobs that are in “Queued” or “Running” state.

Viewing vulnerable registry images

Once you connect to the registry, Container Security pulls the inventory data and performs scans on repositories and images within the registries. Images are listed on the **Assets > Images** tab.

REGISTRY	REPOSITORY	CREATED ON	TAGS	CONTAINERS	VULNERABILITIES	COMPLIANCE
registry-1.docker.io	image_1 Image ID: 4b72a9a97b0	Mar 15, 2021	registrycheck_lat...	0 On Hosts: 0	182	-
registry-1.docker.io	image_2 Image ID: f7b16194d019	Mar 15, 2021	distroless-java8...	0 On Hosts: 0	0	-
docker.io	image_abc Image ID: be249b1ccc35	Mar 15, 2021	latest	1 On Hosts: 1	213	2
docker.io	image_xyz Image ID: 468dc76f8e88	Mar 15, 2021	latest	1 On Hosts: 1	4	2
registry-1.docker.io	my_image Image ID: 34eb6af135a0	Mar 14, 2021	distroless-java11...	0 On Hosts: 0	-	-

To get the total count of vulnerable images in a registry, go to **Assets > Registries** tab, and choose **View Details** from the Quick Actions menu for any registry. You’ll see basic information like total repositories, total images and total vulnerable images. You’ll also see a list of scan schedules created for scanning the registry.

Defining Vulnerability Exceptions (Beta)

You can flag the required vulnerabilities as exceptions for specific images and containers.

Vulnerability exceptions refer to specific vulnerabilities that have been identified within a containerized environment but are intentionally exempted from remediation measures.

Here are a few possible reasons for granting exceptions:

- **False Positives:** Some vulnerabilities reported may be false positives.
- **Third-Party Dependencies:** Certain vulnerabilities may exist in third-party libraries or components that are beyond your immediate control.
- **Compatibility Issues:** Applying a fix for a vulnerability might have other impacts.

To define vulnerability exceptions, go to **Exceptions > Vulnerability Exceptions**.

For more information, refer to Online Help: [Defining Vulnerability Exceptions](#).

Defining Security Policies

You can create policies in Container Security for managing configurations, vulnerability management, compliance, access, and auditing in containerized environments, thus automating the process of securing images and containers. Policies provide a combination of rules that assess specific artifacts such as images, and containers, and provide actions associated with the rules.

Currently, only image assessment policies for CICD are available. You can define rules for scanning container images for known vulnerabilities before deployment and specify the actions to be taken if the count of vulnerabilities of specific severity is exceeded, such as blocking CICD build or triggering alerts.

Go to the **Policies** tab to create a new policy. For more information, see Online Help: [Creating Security Policies](#).

Sensor Profiles

You can create sensor profiles, edit the configuration values, and assign the profiles to the sensors.

For registry sensors, you can configure sensor profiles to control which sensors are used for scanning different registries. Each profile associates a list of registries with a list of sensors that can scan them. This is especially useful when you have sensors that don't have Internet access and are not able to scan cloud-based registries. Now you can create a profile with your cloud-based registries and include only the sensors that can reach them for scanning.

Good to Know

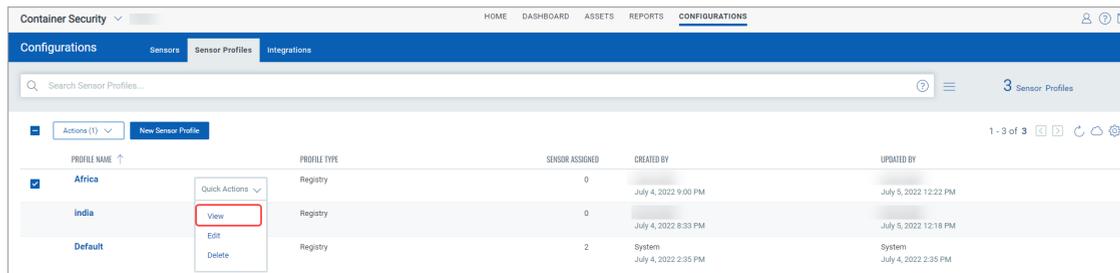
- If you do not associate a sensor profile with a sensor, the default sensor profile is used.
- You can associate one sensor with only one sensor profile.
- In case of registry sensors,
 - You can add multiple registries in a sensor profile.
 - At the scan time, only sensors associated with a registry are used for the scan job. If a registry is not included in a sensor profile, then any sensor can be used to scan it.
 - By default, all the sensors and registries that are not associated with any profile will come under **Default** sensor profile. Any of the registries in the default profile can be scanned from any of the sensors available in the **Default** sensor profile.

PROFILE NAME ↑	PROFILE TYPE	SENSOR ASSIGNED	CREATED BY	UPDATED BY
Default	Registry	0	System July 4, 2022 2:35 PM	System July 4, 2022 2:35 PM

View Sensor Profiles

Sensor profiles are listed on the **Sensor Profiles** tab under **Configurations**. The search field allows you to find sensor profiles by different criteria like profile name, profile UUID or the name of the user who created or updated the profile.

To see more details for a profile in the list, select **View** from the **Quick Actions** menu.



The screenshot shows the 'Sensor Profiles' configuration page in the Container Security interface. The page has a search bar and a table of profiles. The 'India' profile is selected, and its 'Quick Actions' menu is open, with the 'View' option highlighted by a red box. The table contains the following data:

PROFILE NAME	PROFILE TYPE	SENSOR ASSIGNED	CREATED BY	UPDATED BY
Africa	Registry	0	July 4, 2022 9:00 PM	July 5, 2022 12:22 PM
India	Registry	0	July 4, 2022 8:33 PM	July 5, 2022 12:18 PM
Default	Registry	2	System July 4, 2022 2:35 PM	System July 4, 2022 2:35 PM

To perform actions such as add a sensor profile, update a sensor profile, delete a sensor profile, refer **Manage Sensor Profiles** section in the [Online Help](#).

Vulnerability Reporting

Create customizable QQL query driven on-demand report jobs. Reports are driven by reporting templates. Currently we support vulnerability report templates for Images and Containers. Reporting workflows can be performed from the “Reports” tab in the Container Security UI.

These vulnerability report templates are available:

- Image Vulnerability Report
- Container Vulnerability Report

Image Vulnerability Report

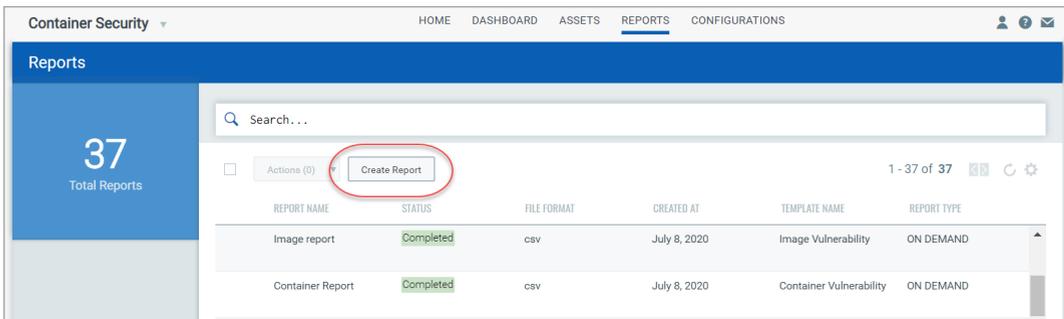
For each row in the report, you’ll see image details (e.g. Repository, Image ID, SHA, etc) followed by vulnerability details (e.g. QID, Title, Severity, etc) for a single detected vulnerability. If the image has multiple vulnerabilities it will be listed multiple times (e.g. 10 rows for 10 vulnerabilities on the same image).

Container Vulnerability Report

For each row in the report, you’ll see container details (e.g. Container Name, Container ID, Host Name, etc) followed by vulnerability details (e.g. QID, Title, Severity, etc) for a single detected vulnerability. If the container has multiple vulnerabilities it will be listed multiple times (e.g. 10 rows for 10 vulnerabilities on the same container).

Create Reports

Go to the **Reports** section (on the top menu) and click the **Create Report** button.



Walk through the **Create New Report** wizard. In the **Report Details** section, give your report a name and description. In the **Report Source** section, choose the report template for the type of report you want to create: Image Vulnerability or Container Vulnerability.

You may choose to add a search query to limit the report to certain images/containers. For an Image Vulnerability report, only the images that match your query will be included. For a Container Vulnerability report, only the containers that match your query will be included.

In the **Report Schedule** section, specify whether you want to create an on-demand report or a scheduled report. For a scheduled report, you need to define a schedule to run the report at regular intervals. You can create a daily, weekly, or monthly recurring schedule.

The **Report Display** section shows you the types of details that can be included in the report. Simply select the check box next to each detail you want to include in the report. Your selections determine which columns appear in the CSV output. Note that certain details are selected by default and cannot be unchecked. Want to include all details? Pick the “Select All” option and all details will be included.

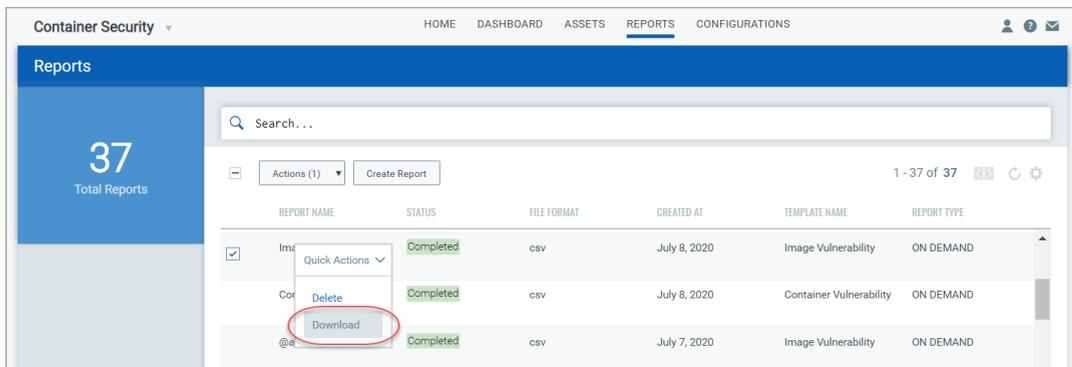
For an on demand report, specify the time zone in which you want to view the dates and time in your report.

Click **Next** again to review the Report Summary and click **Submit** to generate your report job. Once saved, the report job cannot be edited.

Your report job will appear on the reports list with a status of **Accepted**. The status will change to **Completed** once the report is done and ready to download.

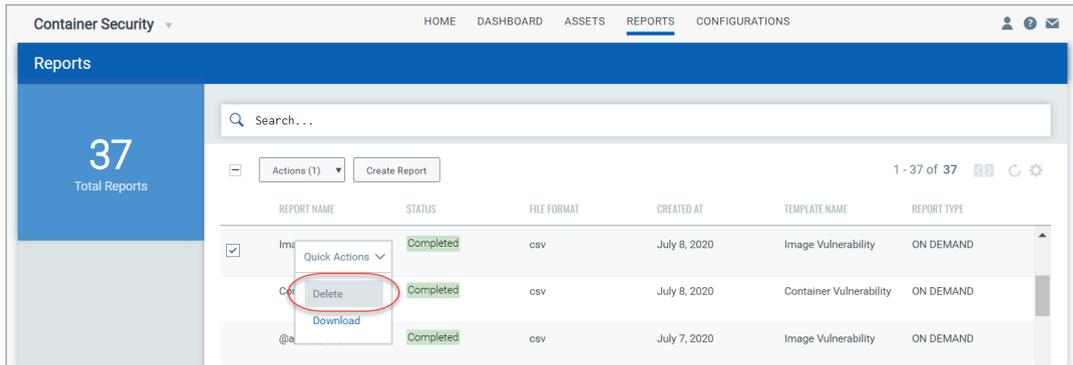
View & Download Reports

Choose **Download** from the Quick Actions menu for a completed report. The CSV report will be saved to your local downloads area. (Tip - Use the Search field above the reports list to quickly find a report using the search token reportName.)



Delete Reports

To delete a single report, choose **Delete** from the Quick Actions menu, as shown below. To delete multiple reports in bulk, select each row for the reports you want to delete and choose **Actions** > **Delete** above the reports list.



Compliance Scanning

Qualys supports compliance scanning/assessments of running containers and images. Perform Policy Compliance (PC) checks and configuration assessments on your running containers and images. We support a subset of controls from CIS Docker benchmarks, which are applicable to running containers and container images. Customers can assess configuration risks in their running containers and images and remediate them accordingly based on the Qualys findings.

Prerequisites

Upgrade your sensors to the latest version (sensor version 1.9.0 or later).

How it works

The updated Qualys Container Sensor runs an additional scan of configurations in containers, images and uploads additional scan metadata to the Qualys backend. Based on the scan metadata, the backend performs an assessment against various industry standard benchmarks and controls for compliance assessment. The compliance scans of containers, images will be transparent to customers and will function in a similar real-time cloud native manner like the vulnerability scanning feature. The configuration scan results will be available in the UI and the API. In the UI, view Image and Container details to get compliance posture (PASS or FAIL) and control information.

View compliance information

You'll see compliance information in the UI for your images and containers. On the Images list and Containers list, you'll see a column called Compliance with the number of controls that have a posture of PASS and FAIL. Here's a sample list of containers:

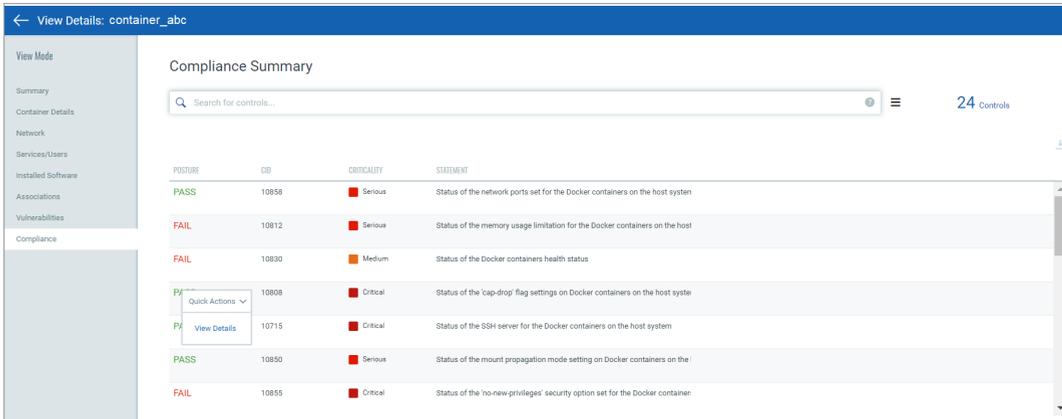
The screenshot shows the 'Containers' tab in the Container Security UI. A summary bar at the top indicates 24 total containers, with 19 root containers, 0 privileged containers, 1 container detected without CS Sensor, 4 containers in drift, and 16 containers not compliant (circled in red). Below this is a table of containers with columns for CONTAINER, CREATED ON, HOST, STATE, LAST SCANNED, VULNERABILITIES, and COMPLIANCE. The COMPLIANCE column shows a score of 24 for all containers, with a green bar indicating full compliance. A 'COMPLIANCE POSTURE' sidebar on the left shows 16 PASS and 16 FAIL results (circled in red).

CONTAINER	CREATED ON	HOST	STATE	LAST SCANNED	VULNERABILITIES	COMPLIANCE
container_1 Container id: 456289061b8f	Mar 14, 2021	dockercnt 10.115.98.192	RUNNING 2 days ago	6 hours ago	212	24
container_2 Container id: 7140cd72554	Mar 14, 2021	dockercnt 10.115.98.192	RUNNING 2 days ago	6 hours ago	212	24
container_3 Container id: 98B4e250f6b	Mar 14, 2021	ip-10-82-9-192 10.82.9.192	RUNNING 3 days ago	3 days ago	4	24
container_abc Container id: 9f0979eca794	Mar 12, 2021	ip-10-82-9-192 10.82.9.192	RUNNING 3 days ago	3 days ago	4	24
container_xyz Container id: 0202a3d22215	Mar 12, 2021	ip-10-82-9-192 10.82.9.192	RUNNING 3 days ago	3 days ago	4	24
my_container Container id: 3e762442295f	Mar 12, 2021	ip-10-82-9-192 10.82.9.192	RUNNING 3 days ago	3 days ago	213	24
sample_container Container id: 6f0c4c5bd265	Feb 24, 2021	localhost.localdomain 10.115.119.175	RUNNING 21 days ago	20 days ago	0	24
sample2_container Container id: 47deb4ed7ac	Feb 24, 2021	localhost.localdomain 10.115.119.175	RUNNING 21 days ago	20 days ago	0	24

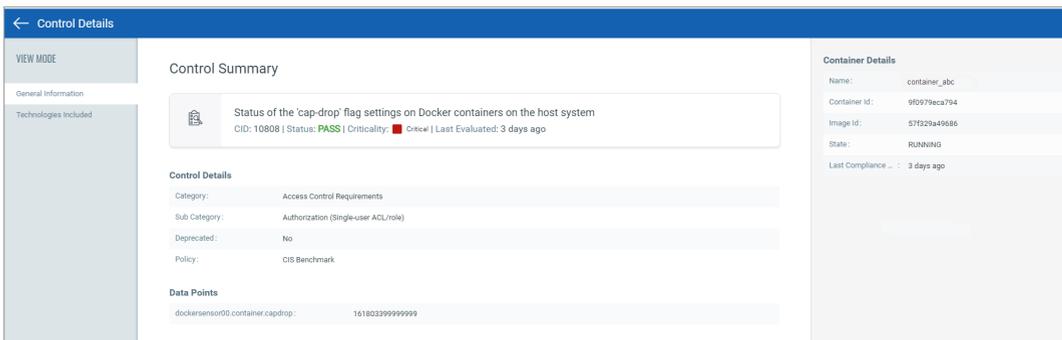
Easily search images and containers by control ID, control criticality (MINIMAL, MEDIUM, SERIOUS, CRITICAL, URGENT) and control posture (PASS, FAIL).

The screenshot shows the search bar in the Container Security UI. The search term 'controls' is entered, and the results are: controls.controlid, controls.criticality, and controls.posture (circled in red). A 'Syntax Help' box is visible on the right, showing the format for searching by control ID: 'controls.controlId: 18826'.

Drill down into the details for any image or container to see compliance information, including the list of controls that were scanned with control details (CID, criticality, statement, category, technologies).



Drill down into the details for any control to get control details, including the control category, policy and technologies.



Compliance information can also be fetched using Compliance APIs. You can fetch compliance posture for an image or container, fetch control details, or fetch a list of controls. See the Compliance section of the [Qualys Container Security API Guide](#).

SCA Scanning

Qualys supports Software Composition Analysis (SCA) scanning of container images. An SCA scan discovers installed open source software and libraries, as well as associated vulnerabilities, present in your container images.

While evaluating security posture of container images it is important to identify all software packages present in the image. The SCA scan can be used to identify programming language-based software packages inside the image. In addition, metadata information for each image layer is also provided. The SCA scan detects packages for these programming languages: Java, Python, Go, Node.js, .NET, PHP, Ruby, and Rust.

SCA scanning is available for all sensor types (General, Registry, and CI/CD), and is supported for Docker, containerd, and CRI-O runtimes. Also, SCA scanning is only supported when scanning container images. SCA scanning is not supported for Mac OS.

Prerequisites

- The SCA Scanning feature must be enabled for your subscription. Contact Qualys Support to have this feature enabled.
- Update your sensors to sensor version 1.19 or later.
- Relaunch your sensors with the parameter **--perform-sca-scan** to perform SCA scanning.

How it works

SCA scanning is not performed by default. Users must enable SCA scanning using the new parameter **--perform-sca-scan** when deploying their sensors. When enabled, an SCA scan is performed after a standard vulnerability scan (Static or Dynamic) on your container images. When the SCA scan completes, the sensor uploads the metadata information collected by the scan to the Qualys backend where posture evaluation is performed. You can view SCA scan data findings in the Container Security UI and API as part of image details. Vulnerability detections found by the SCA scan are presented as QIDs. Filters are provided so you can identify the type of scan (SCA, Dynamic or Static) used to detect a particular vulnerability.

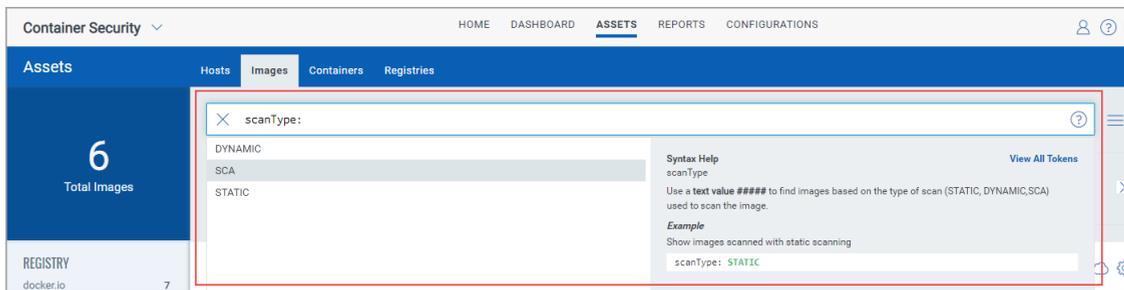
During an SCA scan, the following files are scanned for the language-specific software packages:

Language	Files
Python	egg package wheel package
Node.js	package.json

Language	Files
.NET	packages.lock.json packages.config *.deps.json
Java	JAR/WAR/PAR/EAR
Go	Binaries built by Go
PHP	Composer.lock
Ruby	gemspec
Rust	Cargo.lock and Binaries built with cargo-auditable

View SCA Scanned Images

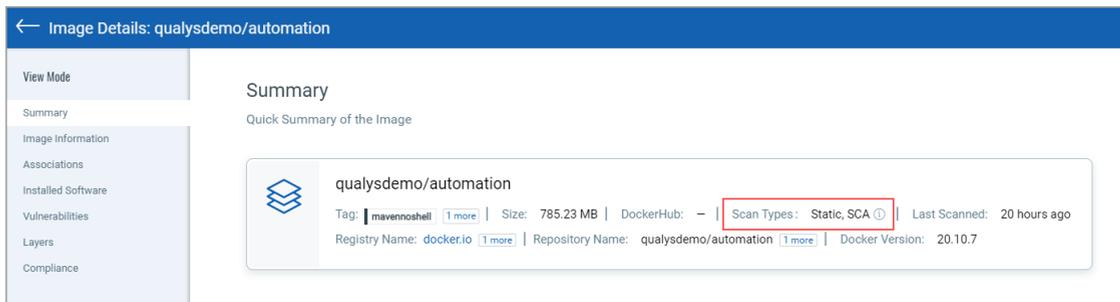
To search images, go to **Assets > Images**. Use **scanType** to find images based on the type of scan (Dynamic, Static or SCA) that was performed to scan the image.



View Image Details

Go to **Assets > Images** and choose **View Details** for any image listed.

The **Summary** tab shows general information about the image. The **Scan Types** field will show the types of scans run on the image, including SCA.



The **Installed Software** tab lists software detected by scans. Use the Packages filter to easily switch the list view. Choose **All** to see all software packages, choose **OS** to see only Operating System based packages, or choose **Non-OS** to see SCA related packages.

← Image Details: qualysdemo/automation

View Mode

- Summary
- Image Information
- Associations
- Installed Software
- Vulnerabilities
- Layers
- Compliance

Installed Software

Search for Installed Software...

TOTAL SOFTWARE: 220

- Patchable (has fix version): 26
- Unpatchable (no fix version): 194

VULNERABILITIES BY SEVERITY

Sev 5 Sev 4 Sev 3 Sev 2 Sev 1

Packages: All OS **Non-OS**

1 - 47 of 47

NAME	INSTALLED VERSION	FIX VERSION	TOTAL QIDS	PACKAGE PATH
org.apache.maven.resolver:maven	1.6.3	-	-	usr/share/maven/lib/m...
org.eclipse.sisu.org:eclipse.sisu.ir	0.3.5	-	-	usr/share/maven/lib/or...

You can also search installed software detected by SCA scans using **scanType: SCA**.

← Image Details: qualysdemo/automation

View Mode

- Summary
- Image Information
- Associations
- Installed Software
- Vulnerabilities
- Layers
- Compliance

Installed Software

scanType: SCA

TOTAL SOFTWARE: 47

- Patchable (has fix version): 3
- Unpatchable (no fix version): 44

VULNERABILITIES BY SEVERITY

Sev 5 Sev 4 Sev 3 Sev 2 Sev 1

Packages: **All** OS Non-OS

1 - 47 of 47

NAME	INSTALLED VERSION	FIX VERSION	TOTAL QIDS
org.apache.maven.resolver:maven-resolver-transport-wagon	1.6.3	-	-
org.eclipse.sisu.org:eclipse.sisu.inject	0.3.5	-	-

The **Vulnerabilities** tab shows vulnerabilities detected by all scans, including SCA scans. The **SCAN TYPE** column identifies the type of scan used for each detection.

Vulnerabilities

Select the severity you would like to review by:

Sev 5 ✓ Sev 4 ✓ Sev 3 ✓ Sev 2 ✓ Sev 1 ✓ Show Patchable Vulnerabilities

Search for vulnerabilities...

VULNERABILITIES BY SEVERITY

QID	VULNERABILITY TITLE	SEVERITY	CVE	AGE	VULNERABLE SOFTWARE	SCAN TYPE
159673	Oracle Enterprise Linux Security U... 20 hours ago	Sev 4	CVE-2022-24407	172 Days	1	Static
159764	Oracle Enterprise Linux Security U... 20 hours ago	Sev 4	CVE-2022-1271	121 Days	1	Static
980276	Java (maven) Security Update for ... 20 hours ago	Sev 3	CVE-2020-8908	164 Days	1	SCA
159624	Oracle Enterprise Linux Security U... 20 hours ago	Sev 4	CVE-2021-3521	184 Days	2	Static

You can also search vulnerabilities detected by SCA scans using **scanType: SCA**.

Vulnerabilities

Select the severity you would like to review by:

Sev 5 ✓ Sev 4 ✓ Sev 3 ✓ Sev 2 ✓ Sev 1 ✓ Show Patchable Vulnerabilities

scanType:SCA

VULNERABILITIES BY SEVERITY

QID	VULNERABILITY TITLE	SEVERITY	CVE	AGE	VULNERABLE SOFTWARE	SCAN TYPE
980276	Java (maven) Security Update for c... 20 hours ago	Sev 3	CVE-2020-8908	164 Days	1	SCA
980351	Java (maven) Security Update for c... 20 hours ago	Sev 3	CVE-2021-29425	164 Days	1	SCA
980408	Java (maven) Security Update for o... 20 hours ago	Sev 3	CVE-2021-37714	164 Days	1	SCA

Note about Vulnerability Counts

You'll notice a difference in the number of vulnerabilities reported for an image that has been scanned by SCA and the number of vulnerabilities for the containers launched from the image. This is because the SCA scan is only run on the image, not on containers, and the SCA scan detects package based vulnerabilities. In other words, the image scan reports all vulnerabilities, including OS based vulnerabilities and Non-OS or SCA package related vulnerabilities whereas the container scan reports only the OS based vulnerabilities.

For example, let's say we scan an image using a sensor launched with the Perform SCA flag enabled and get 25 vulnerabilities reported. We launch a container on this image and it reports 22 vulnerabilities. 3 vulnerabilities were excluded because they were package based.

Secret Detection

Container secrets are digital credentials providing identity authentication and authorizing access to privileged accounts, applications, and services. They can include passwords, API keys, and other credentials that are needed for applications to function properly.

If these secrets are not properly secured, they can be accessed by unauthorized users, leading to malicious attacks. Therefore, discovering secrets is one of the important aspects of container security that organizations must prioritize to protect their sensitive data, meet compliance requirements, and reduce the risk of security incidents.

Container Security can detect secrets for container images enabling you to mitigate potential security risks associated with the accidental or intentional exposure of secrets within containers.

In the **Configuration > Secret Detection** tab, you can see the secret detectors or the set of rules for identifying various types of secrets. Currently, only the default system-defined detectors are available.

Click **View Details** from the Quick Actions menu to view the details of a detector. Note that it is currently not possible to create new detectors or modify existing ones.

Note: Secret detection is supported only on:

- Sensors: CICD and registry
- OS: Linux
- Runtimes: Docker, Containerd, and CRI-O

For more information, refer to Online Help: [Detecting Container Secrets](#).

Administration

For information on sensor installation and troubleshooting, refer to the [Qualys Container Security Sensor Deployment Guide](#).

Sensor updates

Go to **Configurations** > **Sensors** to see a list of sensors. Use the search and filter options to search for sensors. See the [online help](#) for a list of QQL search tokens.

When a newer sensor version is available than the one deployed, you'll see "Update Available" next to the sensor name. You should update the sensor to the newer version to take advantage of new features, bug fixes and to remediate vulnerabilities.

VERSION	STATUS	SENSOR	VERSION	HOST
1.3.1-10	3			
1.3.0-29	1			
	Unknown 8 days ago	f5434a5e4577 qualys-container-sensor Created On: Aug 20, 2019	1.3.1-10	cent731611-76-11 10.115.76.116
	Unknown 9 days ago	095ac35988d0 qualys-container-sensor Created On: Aug 13, 2019	1.3.0-29 Update Available	pci51.r...qualys.com 10.115.77.151

For sensors downloaded from the Qualys UI

Sensors deployed on docker with the installsensor.sh script or docker run command will be updated automatically (unless the --disable-auto-update option was used for the install script). Sensors are not updated automatically for Kubernetes deployments. Refer to "Update the sensor deployed in Kubernetes" in the [Qualys Container Security Sensor Deployment Guide](#) for instructions.

For sensors installed from Docker Hub

The Qualys Container Sensor image hosted on Docker Hub does not support auto update. See "Upgrading the sensor" in the section "Installing the sensor from Docker Hub" in the [Qualys Container Security Sensor Deployment Guide](#) for instructions.

How to uninstall sensor

The `QualysContainerSensor.tar.xz` file (which you download for sensor installation from Qualys Cloud Platform) has the script **`uninstallsensor.sh`** for uninstalling the sensor.

To uninstall a sensor:

If the docker host is configured to communicate over `docker.sock`, use the following command:

```
./uninstallsensor.sh -s
```

If the docker host is configured to communicate over TCP socket, then provide the address on which the docker daemon is configured to listen:

```
./uninstallsensor.sh DockerHost=<<IPv4 address or FQDN>:<Port#>> -s
```

Example:

```
./uninstallsensor.sh DockerHost=10.11.12.13:1234 -s
```

Follow the on-screen prompts to uninstall the sensor. Qualys recommends not to clear the persistent storage.