Container Security

Release Notes for Sensor

Version 1.10.0
October 21, 2021

Here's what's new in the Container Security Sensor!

Sensor Updates for Compliance with CIS Benchmark for Docker

Qualys Container Security adheres to the CIS Benchmark for Docker for our Sensor image. This section provides guidance on how to use the Sensor image in a way that complies with the CIS Benchmark for Docker. We've provided instructions below for a number of controls so you can operate the Sensor in a compliant manner. We've also made changes to the image itself to pass certain controls.

For some controls, you'll use new command line arguments that have been introduced in this release (i.e. memory usage, cpu shares, no new privileges, pid limit) and for other controls you'll use existing command line arguments (e.g. running sensor in read only mode).

CIS Docker Benchmark	**5.9 Ensure that the host's network namespace is not shared (Automated)**
Qualys Control	CID 10811 "Status of the network mode set for the Docker containers on the host system"
Resolution	To meet compliance with this control, Qualys Container Sensor should run *without* the command line argument **--net=host**. However, it should be noted that when sensor is launched without **--net=host**, sensor will not be able to detect its host IP address. By default, the sensor container will be assigned a default IP from the pool assigned to the network. All the containers on the host will be mapped to the IP assigned to the sensor container and each container's host IP association will be missing. Hence, not to lose the asset/host association we recommend the sensor to be launched with the argument.
Installsensor.sh Command	Remove **--net=host** from the installsensor.sh script to run sensor without this command.
Docker Run Command	Do not specify **--net=host** as part of the Docker run command when deploying a sensor. `sudo docker run -d --restart on-failure -v /var/run/docker.sock:/var/run/docker.sock:ro -v /usr/local/qualys/sensor/data:/usr/local/qualys/qpa/data -e ACTIVATIONID=<Activation id> -e CUSTOMERID=<Customer id> -e POD_URL=<POD URL> --name qualys-container-sensor qualys/qcs-sensor:latest`
Kubernetes DaemonSet	For deployments in Kubernetes with Docker Runtime, remove **hostNetwork: true** from cssensor-ds.yml.

CIS Docker Benchmark	**5.10 Ensure that the memory usage for containers is limited (Automated)**
Qualys Control	CID 10812 "Status of the memory usage limitation for the Docker containers on the host system"
Resolution	To meet compliance with this control, Qualys Container Sensor should run with the command line argument **MemoryUsageLimit** for the installsensor.sh script or **-m** as part of Docker run command when

	deploying a sensor. The value should be formatted as <digit><unit> where unit can be any of the following: b (bytes), k (kilobytes), m (megabytes), g (gigabytes). The recommended value is 500m for 500 megabytes.
Installsensor.sh Command	Specify **MemoryUsageLimit** as a command line argument for installsensor.sh script. `sudo ./installsensor.sh ActivationId=<Activation id> CustomerId=<Customer id> Storage=/tmp/qualys/sensor/data MemoryUsageLimit=500m -s`
Docker Run Command	Specify **-m** as part of the Docker run command when deploying a sensor. `sudo docker run -d --restart on-failure -m 500m -v /var/run/docker.sock:/var/run/docker.sock:ro -v /usr/local/qualys/sensor/data:/usr/local/qualys/qpa/data -e ACTIVATIONID=<Activation id> -e CUSTOMERID=<CustomerId> -e POD_URL=<POD URL> --net=host --name qualys-container-sensor qualys/qcs-sensor:latest`

Kubernetes DaemonSet	For deployments in Kubernetes with Docker Runtime, add the **memory** usage limit to the **resources** section in the cssensor-ds.yml file, as shown below.
	
	`resources:`
	`  limits:`
	`  memory: "500M"`

CIS Docker Benchmark	**5.11 Ensure that CPU priority is set appropriately on containers (Automated)**
Qualys Control	CID 10813 “Status of the CPU share weighting set for the Docker containers on the host system”
Resolution	To meet compliance with this control, Qualys Container Sensor should run with the command line argument **CpuShares** for the installsensor.sh script or **--cpu-shares** as part of Docker run command when deploying a sensor. This defines the CPU shares for the sensor container. A valid value is a non-zero, positive integer other than 1024.
Installsensor.sh Command	Specify **CpuShares** as a command line argument for installsensor.sh script. `sudo ./installsensor.sh ActivationId=<Activation id> CustomerId=<Customer id> Storage=/tmp/qualys/sensor/data CpuShares=1023 -s`
Docker Run Command	Specify **--cpu-shares** as part of the Docker run command when deploying a sensor. `sudo docker run -d --restart on-failure --cpu-shares 1023 -v /var/run/docker.sock:/var/run/docker.sock:ro -v /usr/local/qualys/sensor/data:/usr/local/qualys/qpa/data -e ACTIVATIONID=<Activation id> -e CUSTOMERID=<Customer id> -e POD_URL=<POD URL> --name qualys-container-sensor qualys/qcs-sensor:latest`

CIS Docker Benchmark	5.12 Ensure that the container's root filesystem is mounted as read only (Automated)
Qualys Control	CID 10825 "Status of the read-only filesystem for the Docker containers on the host system"
Resolution	To meet compliance with this control, Qualys Container Sensor should run with the command line argument **--read-only** for the installsensor.sh script or as part of Docker run command when deploying a sensor.
Installsensor.sh Command	Specify **--read-only** as a command line argument for installsensor.sh script. `sudo ./installsensor.sh ActivationId=<Activation id> CustomerId=<Customer id> Storage=/tmp/qualys/sensor/data --read-only -s`
Docker Run Command	Specify **--read-only** as part of the Docker run command when deploying a sensor. `sudo docker run -d --restart on-failure --read-only -v /var/run/docker.sock:/var/run/docker.sock:ro -v /usr/local/qualys/sensor/data:/usr/local/qualys/qpa/data -e ACTIVATIONID=<Activation id> -e CUSTOMERID=<Customer id> -e POD_URL=<POD URL> --name qualys-container-sensor qualys/qcs-sensor:latest`

Kubernetes DaemonSet	For deployments in Kubernetes with Docker Runtime, add **readOnlyRootFilesystem: true** in the **securityContext** section of the cssensor-ds.yml file.
	
	`securityContext:`
	`  readOnlyRootFilesystem: true`

CIS Docker Benchmark	5.25 Ensure that the container is restricted from acquiring additional privileges (Automated)
Qualys Control	CID 10855 "Status of the 'no-new-privileges' security option set for the Docker containers on the host system"
Resolution	To meet compliance with this control, Qualys Container Sensor should run with the command line argument **--security-opt=no-new-privileges** as part of the Docker run command when deploying a sensor.
Installsensor.sh Command	There is no new option for the installsensor.sh script. Support for installer script for this argument will be added in a future release. Alternatively, you can use docker run command as specified below to meet this control.
Docker Run Command	Specify **--security-opt=no-new-privileges** as part of the Docker run command when deploying a sensor. `sudo docker run -d --restart on-failure --security-opt=no-new-privileges -v /var/run/docker.sock:/var/run/docker.sock:ro -v /usr/local/qualys/sensor/data:/usr/local/qualys/qpa/data -e`

	`ACTIVATIONID=<Activation id> -e CUSTOMERID=<Customer id> -e POD_URL=<POD URL> --name qualys-container-sensor qualys/qcs-sensor:latest`

CIS Docker Benchmark	5.28 Ensure that the PIDs cgroup limit is used (Automated)
Qualys Control	CID 10829 "Status of the Process ID (PID) cgroup limit for Docker containers"
Resolution	To meet compliance with this control, Qualys Container Sensor should run with the command line argument **PidLimit** for installsensor.sh script or **--pids-limit** as part of Docker run command when deploying a sensor. This defines the Pid limit for the sensor container. The value provided must be a positive integer.
Installsensor.sh Command	Specify **PidLimit** as a command line argument for installsensor.sh script. `sudo ./installsensor.sh ActivationId=<Activation id> CustomerId=<Customer id> Storage=/tmp/qualys/sensor/data PidLimit=100 -s`
Docker Run Command	Specify **--pids-limit** as part of the Docker run command when deploying a sensor. `sudo docker run -d --restart on-failure --pids-limit 100 -v /var/run/docker.sock:/var/run/docker.sock:ro -v /usr/local/qualys/sensor/data:/usr/local/qualys/qpa/data -e ACTIVATIONID=<Activation id> -e CUSTOMERID=<Customer id> -e POD_URL=<POD URL> --name qualys-container-sensor qualys/qcs-sensor:latest`

CIS Docker Benchmark	Multiple Controls
Installsensor.sh Command	This sample includes all the new command line arguments for installsensor.sh for meeting compliance, including MemoryUsageLimit, CpuShares, PidLimit, and --read-only (previously available). Note that --net=host is excluded. `sudo ./installsensor.sh ActivationId=<Activation id> CustomerId=<Customer id>  Storage=/tmp/qualys/sensor/data MemoryUsageLimit=500m CpuShares=1023 PidLimit=100 -s --read-only`
Docker Run Command	This sample includes all the new command line arguments for the Docker run command for meeting compliance, including --security-opt=no-new-privileges, --cpu-shares, --pids-limit, -m, and --read-only (previously available). Note that --net=host is excluded. `sudo docker run -d --restart on-failure --read-only --security-opt=no-new-privileges --cpu-shares 1023 --pids-limit 100 -m 500m -v /var/run/docker.sock:/var/run/docker.sock:ro -v /usr/local/qualys/sensor/data:/usr/local/qualys/qpa/data -e ACTIVATIONID=<Activation id> -e CUSTOMERID=<Customer id> -e POD_URL=<POD URL> --name qualys-container-sensor qualys/qcs-sensor:latest`

Updates to YAML Files for Mounting the Socket

We made updates related to mounting the socket (for docker.sock, containerd.sock, crio.sock) when deploying the sensor in a Kubernetes cluster environment. We made the following changes to the Kubernetes sensor deployment YAML files:

- Mount socket directly instead of the entire directory (/var/run)

- Mount in Read-Only mode

docker.sock

Here are the updated sections of the YAML file (**cssensor-ds.yml**):

```
volumeMounts:
- mountPath: /var/run/docker.sock
  name: socket-volume
  readOnly: true

volumes:
- name: socket-volume
  hostPath:
    path: /var/run/docker.sock
    type: Socket
```

containerd.sock

Here are the updated sections of the YAML file (**cssensor-containerd-ds.yml**):

```
volumeMounts:
- mountPath: /var/run/containerd/containerd.sock
  name: socket-volume
  readOnly: true

volumes:
- name: socket-volume
  hostPath:
    path: /var/run/containerd/containerd.sock
    type: Socket
```

crio.sock

Here are the updated sections of the YAML files (**cssensor-crio-ds.yml, cssensor-openshift-crio-ds.yml**):

```
volumeMounts:
- mountPath: /var/run/crio/crio.sock
  name: socket-volume
  readOnly: true

volumes:
- name: socket-volume
  hostPath:
    path: /var/run/crio/crio.sock
    type: Socket
```

Updates to Persistent Volume Claim YAML File

We made updates to the **cssensor-ds_pv_pvc.yml** file, which is used for PersistentVolumeClaim (PVC) to request for storage of specific size from the gross Persistent Volume specified when deploying a sensor in Kubernetes with Docker Runtime.

We added support for features that were introduced in previous Sensor releases and already updated in other YAML files, but not updated in **cssensor-ds_pv_pvc.yml**, including:

- Deploy sensor in 'qualys' namespace only

- Docker RBAC Support

- Collection of Kubernetes cluster attributes

- Jobs API now used to create image scanning pods

To use the updated yaml file, follow these steps:

1) Uninstall your existing sensor.

2) Download the latest **cssensor-ds_pv_pvc.yml** file directly from https://github.com/Qualys/cs_sensor. This will be available on github once the latest sensor is available from Docker Hub.

3) Deploy the sensor.

Change to OS Name for Google Distroless Images Without Shell

When scanning Google distroless images without shell, the Operating System (OS) value returned for the image will now include "Distroless" in the name so you can more easily identify your distroless images. For example, the OS name "Debian Linux 10.10" in a previous release will now appear as "Distroless Debian Linux 10.10" starting in this release.

Environment Variable QUALYS_SCANNING_CONTAINER_SCOPECLUSTER Being Deprecated

Node affinity is enforced by default, and this means the image scanning takes place on the same node as the sensor that initiated the scan. This is the recommended setting.

The environment variable QUALYS_SCANNING_CONTAINER_SCOPECLUSTER which allowed users to disable node affinity is being deprecated and is not recommended for use. This environment variable will be removed in Container Sensor 1.11. At that time, the environment variable will no longer be supported, and users will no longer be allowed to disable node affinity.

Issues Addressed

- We made a fix for registry sensor to honor scope returned by registry in WWW-Authenticate header instead of using default catalog scope while getting auth token from registry.
- We fixed an issue where image scanning pods were not being deployed when the YAML file included Toleration keys other than “key: node-role.kubernetes.io/master”. Now, regardless of the Tolerations specified in the YAML file, the image scanning pods will still be deployed.
- We fixed an issue where in some cases, the sensor was sending unsupported controller types for containers to the Container Security backend and as a result we could not display all container, image and sensor details in the Container Security UI.