



# Container Security

## Release Notes for Sensor

Version 1.18

August 24, 2022 (Updated October 18, 2022)

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

[Scan Container Images in AWS Fargate \(ECS\)](#)

[New Option for Disabling Image Scans](#)

[Issues Addressed](#)

## Scan Container Images in AWS Fargate (ECS)

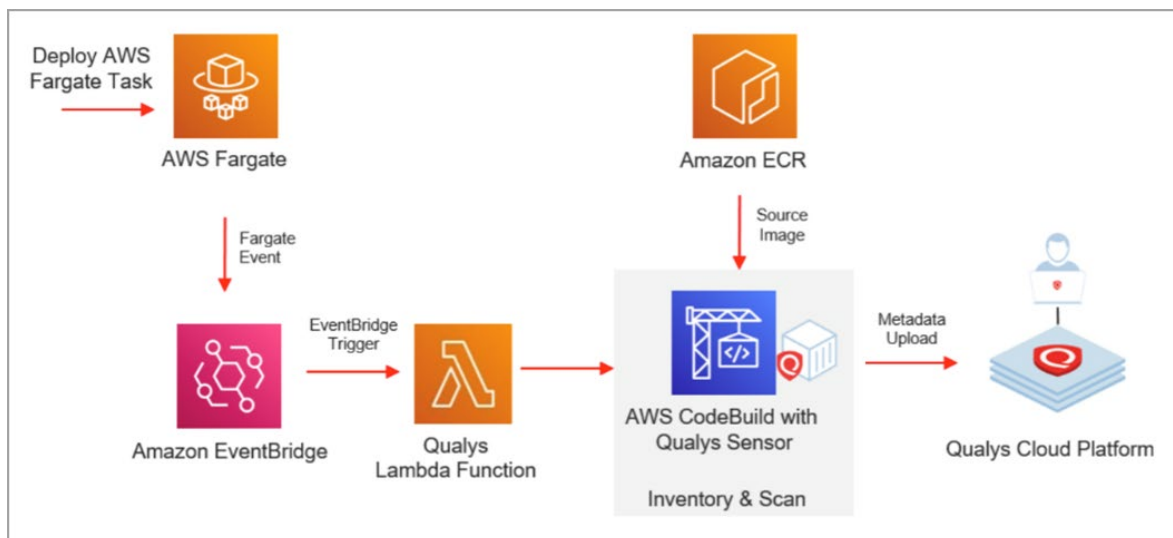
Qualys Container Security can now be used to secure AWS Fargate. AWS Fargate is a serverless compute engine for containers that works with Amazon Elastic Container Service (ECS). This feature allows you to know the containers running on AWS Fargate, perform vulnerability and compliance scanning on container 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 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 and API.



# Qualys AWS ECS Fargate Image Scan Stack Deployment Steps

Follow the steps outlined below to set up AWS ECS Fargate image scanning. You'll create a CloudFormation stack using a Qualys CloudFormation template and a Qualys Lambda function.

## Prerequisites

Before you begin, make sure you have the following items ready to successfully launch the CloudFormation AWS ECS Fargate image scanning stack.

- The AWS region where you want to deploy the stack
- Qualys CloudFormation template URL: <https://qualys-cs-image-scanning-cloud-formation-template.s3.amazonaws.com/qcs-ecs-fargate-image-scanning-cf.template>
- Environment details for your Qualys subscription: POD URL, Activation ID, Customer ID. To get the Activation ID and Customer ID auto-generated for your subscription, go to **Configurations > Sensors** in the UI, click **Download Sensor**. Then click any sensor type. The installation command on the **Installation Instructions** page contains your Activation ID and Customer ID.
- Qualys Lambda function (Zip file). You'll need the S3 bucket name and bucket key for configuring the ECS scanning Lambda function. See the following section to get the S3 bucket name and bucket key for each AWS region: [Qualys CS Lambda Function S3 Bucket Names and Keys](#)
- Qualys sensor image (version 1.18 or later). Refer to [How to Get the Qualys Sensor Image](#)

## How to Get the Qualys Sensor Image

You have these options for the Qualys Container Security Sensor image:

- Use from Docker Hub directly
- Use from Docker Hub but push the image to your ECR repository (public)
- Load from tar and push it to your ECR repository (public)

### Use from Docker Hub directly

You can use the sensor image directly from Docker Hub. The Container Security Sensor on Docker Hub is available as:

```
qualys/qcs-sensor: <tag>
qualys/qcs-sensor:latest
```

Look up the most recent tag in Docker Hub.

### Use from Docker Hub but push the image to your ECR repository (public)

Use the following commands to push the qualys sensor image to the ECR public repository:

```
sudo docker pull qualys/qcs-sensor:latest
sudo docker tag qualys/qcs-sensor:latest <URL to push image to ECR public repository>
sudo docker push <URL to push image to ECR public repository>
```

For example:

```
sudo docker pull qualys/qcs-sensor:latest
sudo docker tag c3fa63a818df public.ecr.aws/y4h7m2t8/qualys/sensor:latest
sudo docker push public.ecr.aws/y4h7m2t8/qualys/sensor:latest
```

### Load from tar and push the image to your ECR repository (public)

Download the **QualysContainerSensor.tar.xz** file from Qualys Cloud Portal on a Linux computer. In the Container Security UI, download the Binary (tar.xz) file by going to **Configurations > Sensors > Download Sensor** and click any sensory type. Then pick **Linux** and the **Binary (tar.xz)** tab. Click **Download Now** to get the tar file.

Untar the sensor package:

```
sudo tar -xvf QualysContainerSensor.tar.xz
```

Use the following commands to push the qualys sensor image to the ECR public repository:

```
sudo docker load -i qualys-sensor.tar
sudo docker tag <IMAGE NAME/ID> <URL to push image to ECR public repository>
sudo docker push <URL to push image to ECR public repository>
```

For example:

```
sudo docker load -i qualys-sensor.tar
sudo docker tag c3fa63a818df public.ecr.aws/y4h7m2t8/qualys/sensor:latest
sudo docker push public.ecr.aws/y4h7m2t8/qualys/sensor:latest
```

## How to deploy the stack using AWS Console

We use AWS CloudFormation for scanning container images in AWS Fargate ECS.

Follow these deployment instructions:

- 1) Log into your AWS Console.
- 2) Go to **CloudFormation**, click **Create Stack** and select **With new Resources**.
- 3) Under **Specify template**, in the **Amazon S3 URL** field, enter the Qualys CloudFormation Template S3 URL (see URL in [Prerequisites](#) section). Then, click **Next** to continue to the template configuration.

**Specify template**  
A template is a JSON or YAML file that describes your stack's resources and properties.

**Template source**  
Selecting a template generates an Amazon S3 URL where it will be stored.

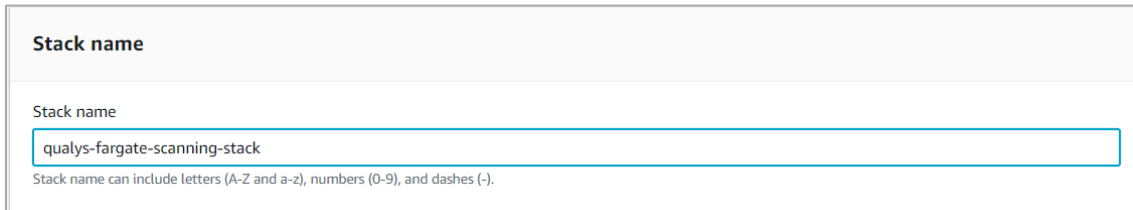
☒ Amazon S3 URL ☐ Upload a template file

**Amazon S3 URL**  
  
Amazon S3 template URL

S3 URL: Will be generated when URL is provided View in Designer

Cancel Next

4) Under **Stack name**, enter a name for the Qualys AWS Fargate scanning stack, such as “qualys-fargate-scanning-stack”.



The screenshot shows a form titled "Stack name". Below the title is a text input field containing the text "qualys-fargate-scanning-stack". Below the input field is a small note: "Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-)." The entire form is enclosed in a light gray border.

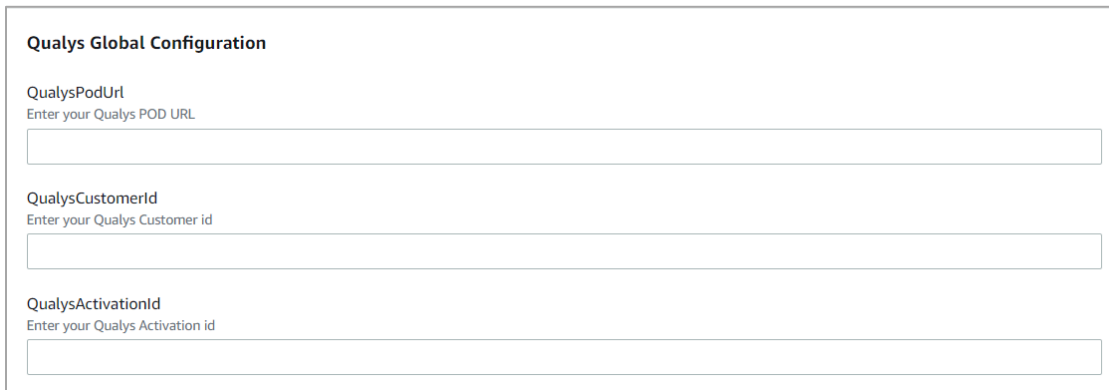
5) Under **Qualys Global Configuration**, provide environment details for your subscription, including POD URL, Activation ID, and Customer ID.

To get the Activation ID and Customer ID auto-generated for your subscription, go to **Configurations > Sensors** in the UI, click **Download**, and then click any sensor type. The installation command on the **Installation Instructions** page contains your Activation ID and Customer ID.

**QualysPodUrl:** Enter the Container Security Server URL for the Qualys Platform where your account is located (e.g. <https://cmsqagpublic.qg2.apps.qualys.com/ContainerSensor> for US2 Platform). If you are not sure of the URL, refer to the following link to identify your Qualys platform and get the Container Security Server URL: <https://www.qualys.com/platform-identification/#container-security-servers>

**QualysCustomerId:** Enter your Qualys subscription’s customer ID.

**QualysActivationId:** Enter your Qualys subscription’s activation ID.



The screenshot shows a form titled "Qualys Global Configuration". It contains three input fields, each with a label and a placeholder text: "QualysPodUrl" with "Enter your Qualys POD URL", "QualysCustomerId" with "Enter your Qualys Customer id", and "QualysActivationId" with "Enter your Qualys Activation id". Each input field is a light gray rectangle. The entire form is enclosed in a light gray border.

6) Under **Qualys CS Lambda function Configuration**, provide Lambda S3 bucket name, key and log level. See the following link to get the bucket name and key values: [Qualys CS Lambda Function S3 Bucket Names and Keys](#)

**QualysLambdaFunctionS3BucketName:** Enter the S3 bucket name for the Qualys Lambda function.

**QualysLambdaFunctionS3BucketKey:** Enter the S3 bucket key as the name of the Qualys Lambda function (e.g., qcslambda-1.0.0-34-PUBLIC.zip).

**QualysLambdaLogLevel:** Select a log level for the Qualys Lambda function. The default value is “info”. Keep the default value or select another log level if more verbose logging is needed.

**Qualys CS Lambda function Configuration**

**QualysLambdaFunctionS3BucketName**  
Enter the S3 bucket name for Qualys lambda function deployment package

**QualysLambdaFunctionS3BucketKey**  
Enter the S3 bucket key for Qualys lambda function deployment package

**QualysLambdaLogLevel**  
Enter the log level for Qualys lambda function

7) Under **Qualys CS Sensor Configuration**, provide the following details:

**QualysSensorImage**: Enter the name of the CS 1.18.0 sensor image.

**QualysSensorLogLevel**: Select a log level (0-5) for the Qualys sensor. The default value is 3 (Information). Keep the default value or select another log level if more verbose logging is needed.

**QualysSensorCLIParameters**: You can keep this field empty.

**Qualys CS Sensor Configuration**

**QualysSensorImage**  
Enter the Qualys CS sensor image name with tag

**QualysSensorLogLevel**  
Enter the log level for Qualys Container Security Scanner

**QualysSensorCliParameters**  
Enter the list of CLI parameters that needs to configured for Qualys CS sensor

8) Click **Next** to continue through the workflow. On the final page, you will need to select the **I acknowledge that AWS CloudFormation might create IAM resources** check box.

**Capabilities**

**The following resource(s) require capabilities: [AWS::IAM::Role]**

This template contains Identity and Access Management (IAM) resources that might provide entities access to make changes to your AWS account. Check that you want to create each of these resources and that they have the minimum required permissions. [Learn more](#)

☒ I acknowledge that AWS CloudFormation might create IAM resources.

Cancel Previous Create change set **Create stack**

9) Click **Create stack**. That's it!

## Resources Created

When the stack creation is successful, several resources are created and they'll appear in the **Resources** section, as shown below. In this example, the resources were created for a stack named "fargate-demo".

fargate-demo						
<div>DeleteUpdateStack actions ▼Create stack ▼</div>						
Stack infoEventsResourcesOutputsParametersTemplateChange sets						
Resources (6) <div>Search resources</div>						
Logical ID ▲	Physical ID ▼	Type ▼	Status ▼	Status reason ▼	Module ▼	
QualysECSFargateImageScanningBuildProject	QualysECSFargateImageScanning	AWS::CodeBuild::Project	UPDATE_COMPLETE	-	-	
QualysECSFargateImageScanningInvokeLambdaPermission	fargate-demo-QualysECSFargateImageScanningInvokeLambdaPermission-1OM4H7PQZVL8G	AWS::Lambda::Permission	CREATE_COMPLETE	-	-	
QualysECSFargateImageScanningLambda	<a href="#">QualysECSFargateImageScanningLambda</a>	AWS::Lambda::Function	UPDATE_COMPLETE	-	-	
QualysECSFargateImageScanningLambdaRole	<a href="#">fargate-demo-QualysECSFargateImageScanningLambdaRole-1OTJJD585AXX</a>	AWS::IAM::Role	CREATE_COMPLETE	-	-	
QualysECSFargateImageScanningRule	<a href="#">QualysECSFargateImageScanningRule</a>	AWS::Events::Rule	CREATE_COMPLETE	-	-	
QualysECSFargateImageScanningServiceRole	<a href="#">fargate-demo-QualysECSFargateImageScanningServiceRole-SP6940MZM9LU</a>	AWS::IAM::Role	CREATE_COMPLETE	-	-	

Here's another look at the resources created.

Logical ID	Type
QualysECSFargateImageScanningBuildProject	AWS::CodeBuild::Project
QualysECSFargateImageScanningInvokeLambdaPermission	AWS::Lambda::Permission
QualysECSFargateImageScanningLambda	AWS::Lambda::Function
QualysECSFargateImageScanningLambdaRole	AWS::IAM::Role
QualysECSFargateImageScanningRule	AWS::Events::Rule
QualysECSFargateImageScanningServiceRole	AWS::IAM::Role

## Qualys CS Lambda Function S3 Bucket Names and Keys

### CS Lambda Function S3 Bucket Name

Refer to the table below to get the Qualys CS Lambda function S3 bucket name for your region.

Please note that we will add support for additional AWS regions in the future. For the most current list of supported regions and bucket names, refer to the “Scan Container Images in AWS Fargate (ECS)” section of the [Sensor Deployment Guide](#).

AWS Region	Bucket Name
us-east-1	qualys-cs-image-scanning-lambda-function-us-east-1
us-east-2	qualys-cs-image-scanning-lambda-function-us-east-2
us-west-1	qualys-cs-image-scanning-lambda-function-us-west-1
us-west-2	qualys-cs-image-scanning-lambda-function-us-west-2
me-south-1	qualys-cs-image-scanning-lambda-function-me-south-1
eu-central-1	qualys-cs-image-scanning-lambda-function-eu-central-1
eu-west-2	qualys-cs-image-scanning-lambda-function-eu-west-2
eu-north-1	qualys-cs-image-scanning-lambda-function-eu-north-1

### CS Lambda Function S3 Bucket Key

The Qualys CS Lambda function S3 bucket key is the same across all AWS regions.

The bucket key is: **qcslambda-1.0.0-34-PUBLIC.zip**



## New Option for Disabling Image Scans

We've introduced a new argument called "--disableImageScan". When you install the General sensor with this new argument, the sensor will not perform dynamic or static scanning of container images. Image scanning will be disabled. This new option is available for General sensor type only, and is available for all Runtimes (Docker, CRI-O and Containerd).

### InstallSensor.sh Command

Specify --disableImageScan as a command line argument for installSensor.sh script.

```
sudo ./installSensor.sh ActivationId=<Activation id> CustomerId=<Customer id>  
Storage=/usr/local/qualys/sensor/data -s --disableImageScan
```

### Docker Run Command

Specify --disableImageScan 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> --net=host --name qualys-  
container-sensor qualys/qcs-sensor:latest --disableImageScan
```

### Yaml Argument

Add --disableImageScan argument in args section of yaml. See examples below for different Runtimes.

#### Docker Runtime:

```
args: ["--k8s-mode", "--disableImageScan"]
```

#### Containerd Runtime:

```
args: ["--k8s-mode", "--container-runtime", "containerd", "--disableImageScan"]
```

#### CRI-O Runtime:

```
args: ["--k8s-mode", "--container-runtime", "cri-o", "--disableImageScan"]
```

## Issues Addressed

- We fixed an issue where an inaccurate Created date (shown in Created On column on Containers list) and Running timeframe (shown in State column on Containers list) was reported by the sensor for containers on CRI-O Runtime.