



Qualys Cloud Detection and Response

CDR Cloud Deployment

User Guide

May 05, 2023

Copyright 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 Cloud Detection and Response (CDR)	3
Getting Started	3
Deploying Qualys with Amazon Traffic Mirroring	3
Prerequisites	3
Deploying Qualys CDR	3
Steps for deploying Qualys CDR as a Standalone Amazon EC2.....	4
Deploying Qualys CDR in High-availability Load-Balanced Autoscaling Mode	7
Deploying Traffic Mirroring on Workloads	7
Setting up Traffic Mirror Session	8
Using Terraform Module	9
Using AWS Console or Customized Deployments.....	10
Next Steps	11
Advanced Steps for Setting Up Traffic Mirror Session	11
Mirroring Only Internet Traffic.....	11
Cross-Account Traffic Mirroring	12
Reference Architecture	12
About Integrations (AWS Security Hub and Lambda)	13
Creating Security VPC for HA Deployment	15

About Cloud Detection and Response (CDR)

With Qualys CDR, get deep visibility into your workloads from the network vantage point and secure them against advanced threats through the power of Deep Learning AI. For example, assume an attacker has discovered your Secure Shell (SSH) key in a public repository, runs a port scan to discover accessible instances of [Amazon Elastic Compute Cloud](#) (Amazon EC2), and tries to install Coin-miner malware on them. When this happens, Qualys CDR detects, in real-time,

- the port scan,
- the malicious Coin-miner payload transfer,
- and command and control (C2) communications to attacker-controlled known/unknown domains.

You can deploy Qualys CDR in minutes through CloudFormation and configure your Virtual Private Cloud (VPC) for agentless monitoring via AWS VPC Traffic Mirroring.

First, deploy Qualys CDR in either standalone or high-availability auto-scaling mode, then configure traffic mirroring for your VPCs, subnets, or tagged EC2/EKS instances.

Getting Started

Connecting Qualys to your AWS account(s) is the first step to protecting your cloud with Qualys Agentless Runtime Cloud Security powered by Deep Learning AI.

The steps below require a valid Qualys SaaS and deployment license. You can contact your Qualys TAM for a license. You can also request a trial license [here](#).

Your Qualys TAM will provide licenses and links to download Terraform scripts. Using AWS CloudShell, CDR can be deployed and configured easily.

Deploying Qualys with Amazon Traffic Mirroring

To deploy Qualys with Amazon Traffic Mirroring, follow these steps:

1. Qualys AMI is available as a private AMI. You must share your AWS account and region with TAM. Qualys will then share the BH AMI with your AWS account.
2. You can provision the stack using the CloudFormation Template (provided via S3).
3. You can configure your AWS console to mirror traffic.

Prerequisites

Deploying the Terraform module using AWS Cloud Shell is preferred since it already has some of the prerequisite tools installed. If you have already installed the prerequisites, you can switch to [this step](#).

Terraform

Install Terraform to create and manage the Qualys environment on AWS infrastructure.

<https://www.terraform.io/downloads.html>

Purpose: Creating and managing the Qualys environment on AWS infrastructure.

Deploying Qualys CDR

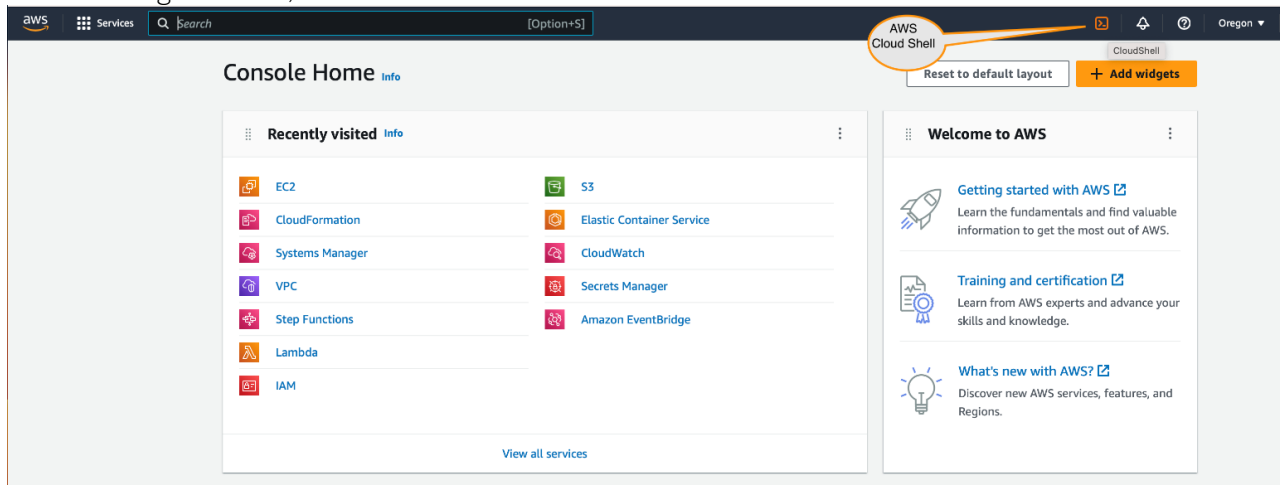
You can deploy Qualys CDR as a standalone Amazon EC2 virtual machine or in high availability autoscaling mode.

Qualys Documentation

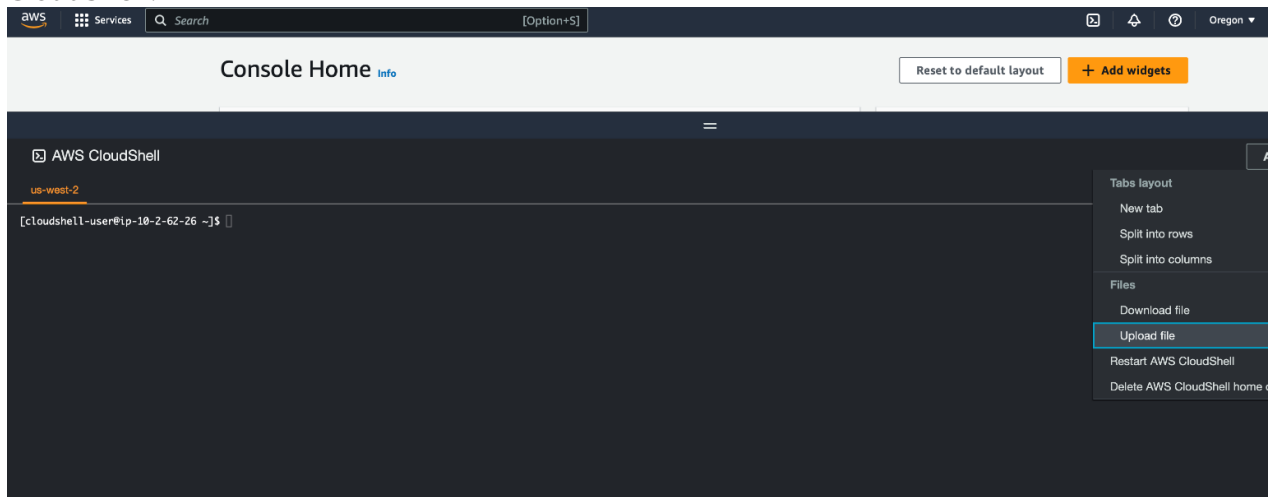
- Qualys CDR stack consists of one EC2 instance and the required Security Group and IAM role in standalone mode.
- AWS VPC Traffic Mirroring supports a maximum of ten mirror sources per EC2 instance configured as a mirroring target. You can use the high-availability load-balanced autoscaling mode for mirroring from more than ten sources to Qualys CDR.

Steps for deploying Qualys CDR as a Standalone Amazon EC2

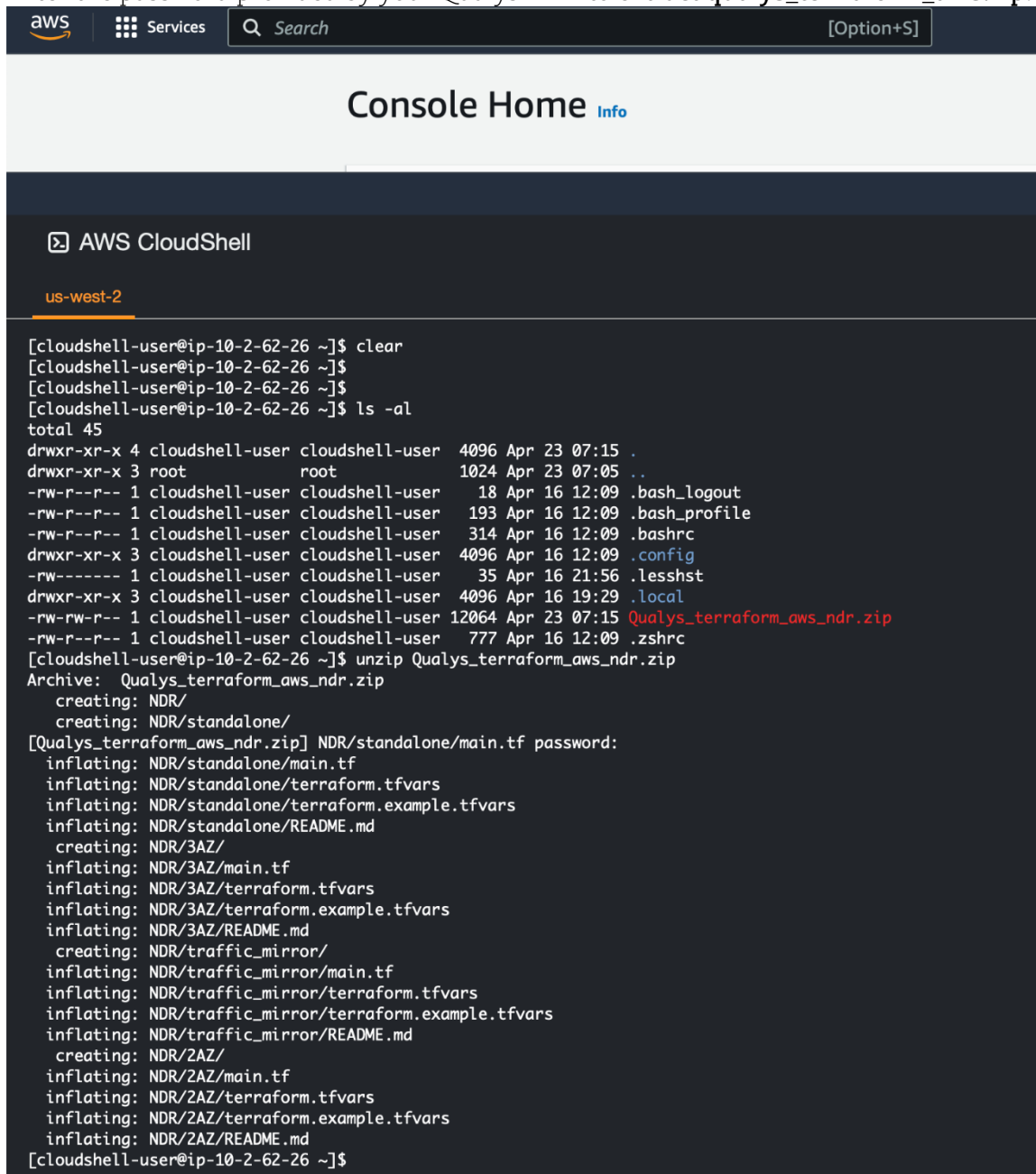
1. On the navigation bar, click **CloudShell**.



2. Download the Terraform module and qualys_terraform_aws.zip and upload them to **CloudShell**.



3. Enter the password provided by your Qualys TAM to extract **qualys_terraform_aws.zip**.



```
aws Services Search [Option+S]
```

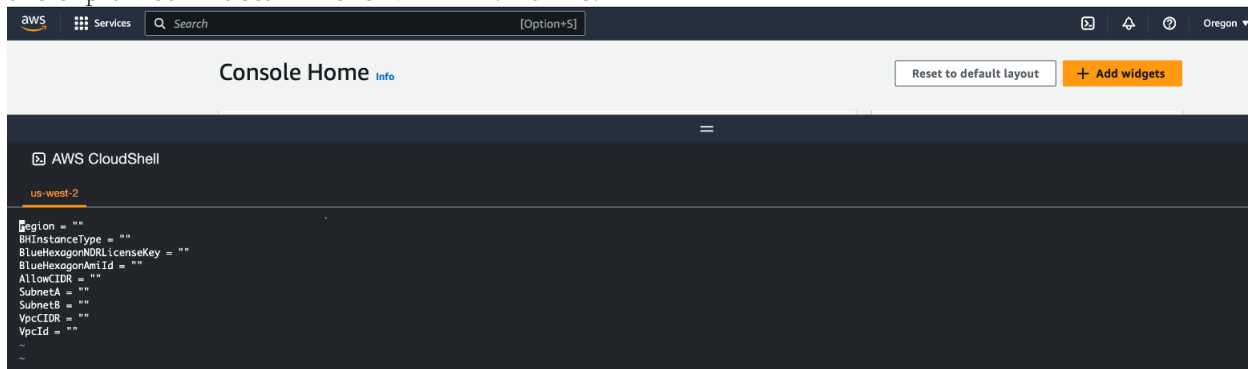
Console Home [Info](#)

AWS CloudShell

us-west-2

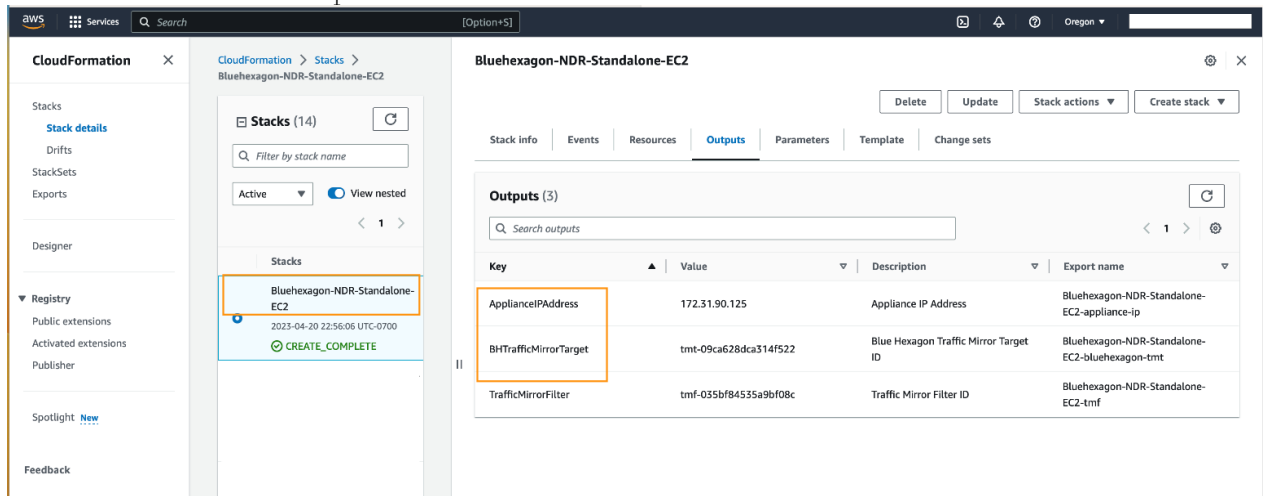
```
[cloudshell-user@ip-10-2-62-26 ~]$ clear
[cloudshell-user@ip-10-2-62-26 ~]$
[cloudshell-user@ip-10-2-62-26 ~]$
[cloudshell-user@ip-10-2-62-26 ~]$ ls -al
total 45
drwxr-xr-x 4 cloudshell-user cloudshell-user 4096 Apr 23 07:15 .
drwxr-xr-x 3 root root 1024 Apr 23 07:05 ..
-rw-r--r-- 1 cloudshell-user cloudshell-user 18 Apr 16 12:09 .bash_logout
-rw-r--r-- 1 cloudshell-user cloudshell-user 193 Apr 16 12:09 .bash_profile
-rw-r--r-- 1 cloudshell-user cloudshell-user 314 Apr 16 12:09 .bashrc
drwxr-xr-x 3 cloudshell-user cloudshell-user 4096 Apr 16 12:09 .config
-rw----- 1 cloudshell-user cloudshell-user 35 Apr 16 21:56 .lessht
drwxr-xr-x 3 cloudshell-user cloudshell-user 4096 Apr 16 19:29 .local
-rw-rw-r-- 1 cloudshell-user cloudshell-user 12064 Apr 23 07:15 Qualys_terraform_aws_nldr.zip
-rw-r--r-- 1 cloudshell-user cloudshell-user 777 Apr 16 12:09 .zshrc
[cloudshell-user@ip-10-2-62-26 ~]$ unzip Qualys_terraform_aws_nldr.zip
Archive: Qualys_terraform_aws_nldr.zip
  creating: NDR/
  creating: NDR/standalone/
[Qualys_terraform_aws_nldr.zip] NDR/standalone/main.tf password:
  inflating: NDR/standalone/main.tf
  inflating: NDR/standalone/terraform.tfvars
  inflating: NDR/standalone/terraform.example.tfvars
  inflating: NDR/standalone/README.md
  creating: NDR/3AZ/
  inflating: NDR/3AZ/main.tf
  inflating: NDR/3AZ/terraform.tfvars
  inflating: NDR/3AZ/terraform.example.tfvars
  inflating: NDR/3AZ/README.md
  creating: NDR/traffic_mirror/
  inflating: NDR/traffic_mirror/main.tf
  inflating: NDR/traffic_mirror/terraform.tfvars
  inflating: NDR/traffic_mirror/terraform.example.tfvars
  inflating: NDR/traffic_mirror/README.md
  creating: NDR/2AZ/
  inflating: NDR/2AZ/main.tf
  inflating: NDR/2AZ/terraform.tfvars
  inflating: NDR/2AZ/terraform.example.tfvars
  inflating: NDR/2AZ/README.md
[cloudshell-user@ip-10-2-62-26 ~]$
```

4. Modify the Terraform .tfvars file to manage deployment variables. Terraform variables are explained in detail in the README.md file.



Qualys has shared the latest AMI for your deployment region with your registered AWS account. You can find the latest Qualys CDR AMI [here](#) as well.

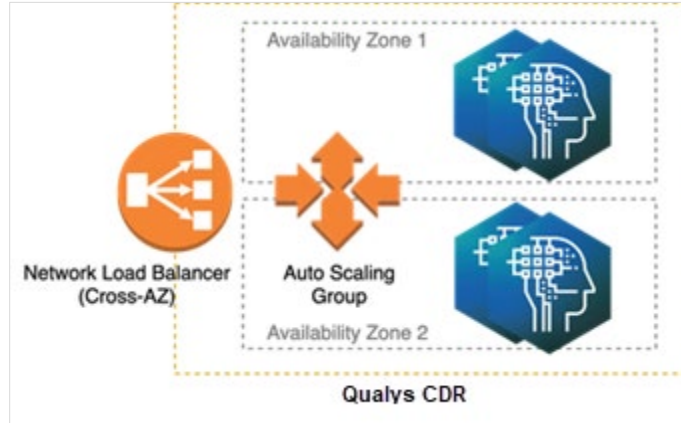
5. In the deployment wizard, specify the correct AMI ID for the region. Terraform is ready to deploy the CDR in your security account's VPC once Terraform .tfvar files are set.
6. Run the following commands to deploy the module to each AWS subscription as needed.
 - terraform init
 - terraform apply -auto-approve
7. After terraform apply runs successfully and the application registers with Qualys, a CloudFormation stack should be created in the AWS account. This stack should show the resources needed to set up a traffic mirror session.



Deploying Qualys CDR in High-availability Load-Balanced Autoscaling Mode

For Qualys to run in high-availability mode, you must have a VPC with at least two private subnets distributed across multiple AZs (Availability Zones).

If you do not have a VPC, follow the steps outlined in the section [Create Security VPC for HA Deployment](#).



Qualys CDR can be deployed in high availability autoscaling mode using the Terraform module provided for two availability zones (2AZ) and three availability zones (3AZ).

As shown in the image above,

- for fault tolerance, the stack consists of a Network Load Balancer with cross-Availability Zone load balancing,
- an Auto Scaling group that automatically scales based on monitored traffic bandwidth,
- and EC2 instances launched in the Auto Scaling group in multiple Availability Zones for deep learning-based threat detection and visibility.

Deploying Traffic Mirroring on Workloads

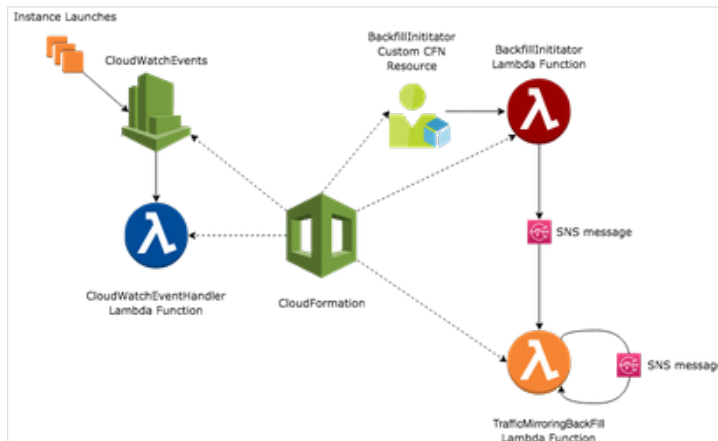
VPC Traffic Mirroring is supported on network interfaces attached to EC2 and EKS instances. You can find a full list of instance types that support VPC traffic mirroring and other considerations [here](#). AWS offers a serverless application for automating traffic mirroring based on VPCs, subnets, or tags as input.

The following image shows the application architecture. From the zip file you downloaded in the [Deploying Qualys CDR](#) section, you can launch the application using the Terraform module for traffic mirroring, which Qualys has packaged in an easy-to-use CloudFormation template.

By specifying the VPCs or subnets to monitor, the serverless application will set up traffic mirroring sessions on existing instances or future instances in the selected VPCs or subnets. Also, you can specify instance tags so that the serverless application mirrors traffic across instances with matching tags (existing or future). Terraform's traffic mirroring module simplifies deployment using AWS's in-built features.

This solution uses Network Load Balancer (NLB), Auto-Scaling Group (ASG), and AWS Console. Alternatively, you can use third-party solutions.

Note: For help with deployment, contact your Qualys Technical Account Manager (TAM).



Note: Tags, subnets, and VPCs are matched independently. Traffic mirroring will still be set up on an instance with a matching tag, even if it does not belong to any subnets or VPCs specified. In the same way, if an instance is launched in a selected VPC, traffic mirroring will be set up, regardless of whether the instance is in any of the chosen subnets or matches any tags.

The Qualys Terraform module for CDR deployment also includes a module to deploy traffic mirroring stacks in traffic source accounts. With the CDR deployment module:

- You can use the mirroring Terraform module.
- You can also create your Terraform module or use CFT or AWS console to add traffic mirror sessions.

To automate the process of adding mirroring sessions for workloads deployed in the account, the script deploys the CFT described above.

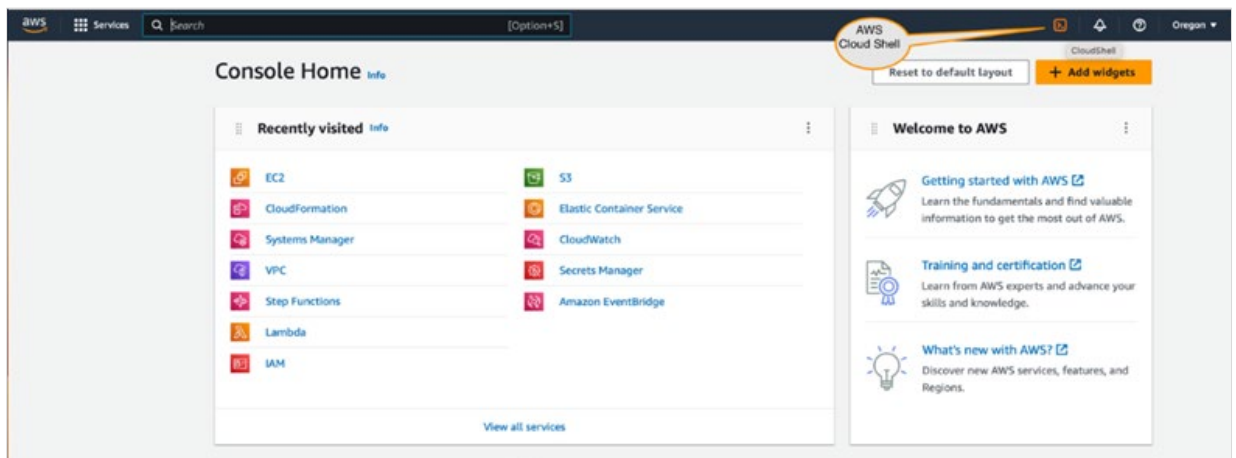
Setting up Traffic Mirror Session

You can do this in two ways, by:

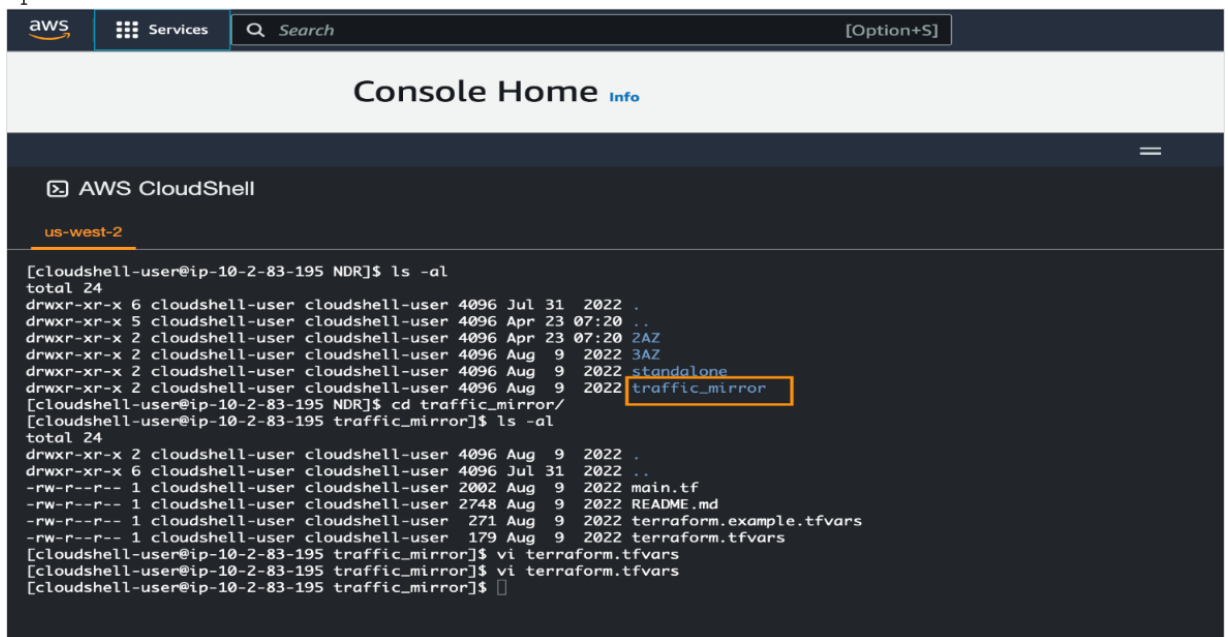
- [Using Terraform Module](#)
- [Using AWS Console or Customized Deployments](#)

Using Terraform Module

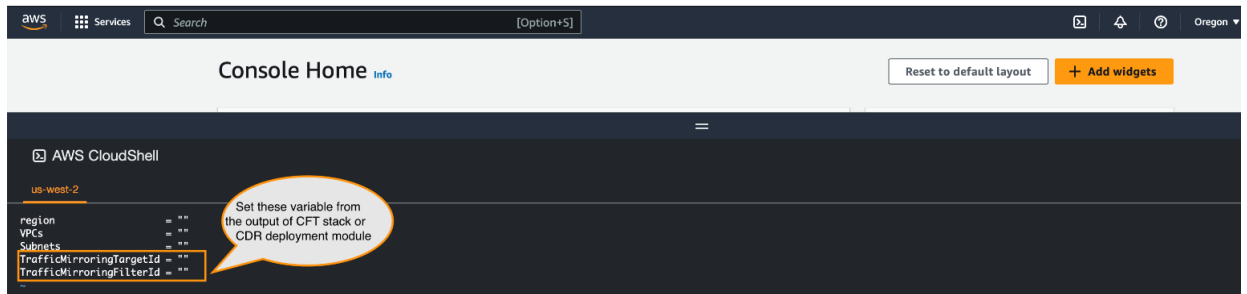
1. On the navigation bar, click the **CloudShell** icon.



2. Go to the traffic mirroring Terraform module in the CDR deployment module you uploaded earlier to CloudShell.

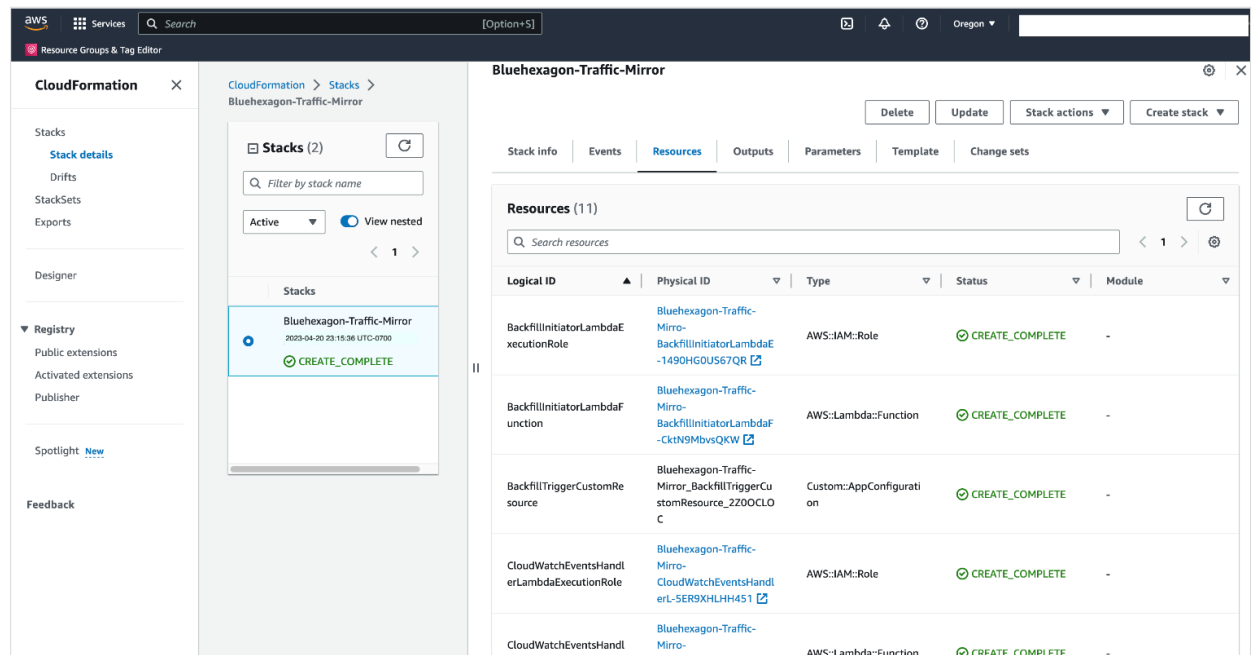


3. Modify the Terraform .tfvars file to manage deployment variables. Terraform variables are explained in detail in the README.md file.



CDR deployment module, as described here, creates traffic mirror targets and traffic mirror filters. You must use the module output to set the variables shown in the diagram above.

4. Run the following commands to deploy the module to each AWS subscription as needed.
 - `terraform init`
 - `terraform apply-auto-approve`
5. After `terraform apply` runs successfully and creates the stack, it will add resources to listen to events for workload activation. It will also add functions to update the traffic mirror session.



Using AWS Console or Customized Deployments

You can configure the "Traffic mirror session" in the VPC console according to your specifications and control the mirror traffic scope using the AWS console. Also, you can customize the Terraform module to suit your deployment needs.

To delete *all* traffic mirroring sessions in your account, use the command below.

```
aws ec2 describe-traffic-mirror-sessions | jq
.TrafficMirrorSessions[].TrafficMirrorSessionId | xargs -I {} aws ec2 delete-
traffic-mirror-session --traffic-mirror-session-id {} --dry-run
```

Run the command without `--dry-run` to actually delete your session.

Next Steps

Qualys CDR virtual appliances will begin inspecting your network traffic as soon as traffic mirroring is enabled on your workloads, providing deep L3-L7 visibility and threat detection, surfacing security findings, and validating threats in the [portal](#).

Advanced Steps for Setting Up Traffic Mirror Session

Mirroring Only Internet Traffic

If you wish to mirror only internet traffic or “North-South” traffic and not internal “East-West” traffic, you can either:

- update the Qualys CDR CloudFormation template with the below traffic mirror filter rules, or
- instantiate the template and update the VPC Traffic Mirroring Filter created via the AWS Console.

The **traffic mirror filter rules** below should be updated to include or exclude your “internal” VPC CIDR blocks (e.g., 172.16.0.0/12, 192.168.0.0/16). NATed public load balancer traffic, DNS traffic (to an internal server), etc., can also be included with appropriate rules. For more information, contact your TAM.

```
QCDRTrafficMirrorFilterRuleIngressRejectLocal:
  Type: "AWS::EC2::TrafficMirrorFilterRule"
  Properties:
    Description: "Qualys Traffic Mirror Filter Rule"
    TrafficMirrorFilterId: !Ref QCDRTrafficMirrorFilter
    TrafficDirection: "ingress"
    RuleNumber: 10
    DestinationCidrBlock: "10.0.0.0/8"
    SourceCidrBlock: "10.0.0.0/8"
    RuleAction: "reject"
```

```
QCDRTrafficMirrorFilterRuleEgressRejectLocal:
  Type: "AWS::EC2::TrafficMirrorFilterRule"
  Properties:
    Description: "Qualys Traffic Mirror Filter Rule"
    TrafficMirrorFilterId: !Ref QCDRTrafficMirrorFilter
    TrafficDirection: "egress"
    RuleNumber: 10
    DestinationCidrBlock: "10.0.0.0/8"
    SourceCidrBlock: "10.0.0.0/8"
    RuleAction: "reject"
```

```
QCDRTrafficMirrorFilterRuleIngress:
```

```
Type: "AWS::EC2::TrafficMirrorFilterRule"
Properties:
  Description: "Qualys Traffic Mirror Filter Rule"
  TrafficMirrorFilterId: !Ref QCDRTrafficMirrorFilter
  TrafficDirection: "ingress"
  RuleNumber: 20
  DestinationCidrBlock: "0.0.0.0/0"
  SourceCidrBlock: "0.0.0.0/0"
  RuleAction: "accept"
```

```
QCDRTrafficMirrorFilterRuleEgress:
Type: "AWS::EC2::TrafficMirrorFilterRule"
Properties:
  Description: "Qualys Traffic Mirror Filter Rule"
  TrafficMirrorFilterId: !Ref QCDRTrafficMirrorFilter
  TrafficDirection: "egress"
  RuleNumber: 20
  DestinationCidrBlock: "0.0.0.0/0"
  SourceCidrBlock: "0.0.0.0/0"
  RuleAction: "accept"
```

Cross-Account Traffic Mirroring

Mirroring traffic from VPC B in Account B to a (Qualys) traffic mirror target in VPC A in Account A (created by deploying the templates in earlier steps):

1. Utilize [AWS Resource Access Manager \(RAM\)](#) to share the Qualys traffic mirror target.
2. Configure a [VPC Peering](#) or [Transit Gateway](#) to route traffic between VPCs B and A.
3. Ensure the `AllowCIDR` parameter in the Qualys CDR template includes the CIDR address block of VPC B. This is so that packets are delivered to the Qualys CDR instances without being blocked by the respective security groups.

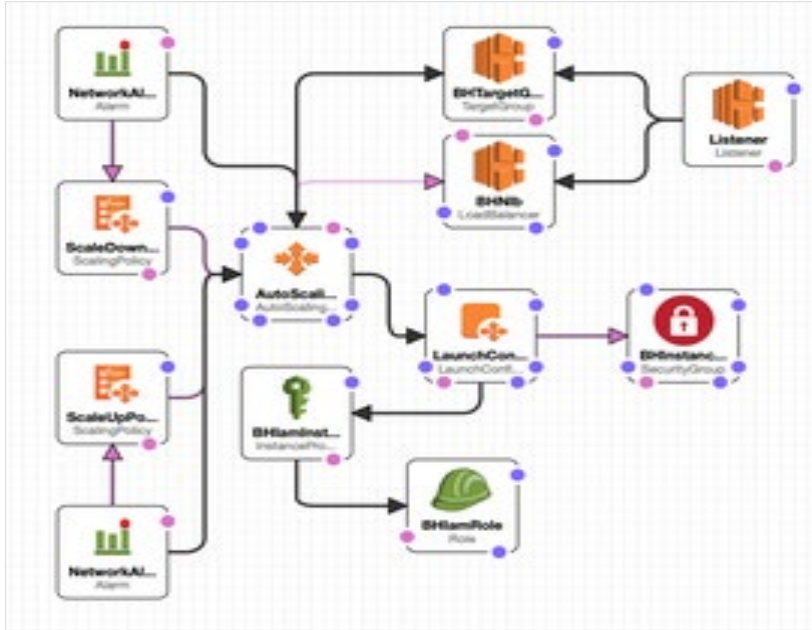
Reference Architecture

Traffic Mirroring in AWS is a virtual TAP that provides direct access to all raw packets flowing through a VPC. To perform traffic analysis, this traffic will be forwarded to a Qualys Virtual Appliance deployed in your VPC.

This deployment architecture involves AWS native solutions:

- Network Load Balancer
- Auto Scaling Group
- IAM Policies
- Security Groups

Following is a logical diagram of all the components involved in the architecture:



Here are the components of the architecture:

- A Network Load Balancer (NLB) is deployed in a VPC's private subnet. The NLB is not internet-facing.
- An Auto Scaling Group (ASG) is deployed. It is responsible for the auto-deployment and scaling of the CDR AMI.
 - A minimum of 1 instance is deployed.
 - The maximum is set at 6.
 - If the *Average Network Bytes In*, in a 10-minute window, exceeds 500 MB, the number of running instances is increased by 1.
 - If the *Average Network Bytes In*, in a 10-minute window, stays below 500 MB, the number of running instances is decreased by 1.
 - Running Instance count cannot be 0.
- Instances are associated with an Instance Profile, which contains the following policies:
 - `arn:aws:iam::aws:policy/SecurityAudit`
 - `arn:aws:iam::aws:policy/ViewOnlyAccess`
- The NLB FQDN will be registered as a Traffic Mirror Target.

About Integrations (AWS Security Hub and Lambda)

You can configure Qualys to publish findings to an AWS SNS topic. The finding triggers an AWS Lambda for either;

- forwarding the finding to AWS Security Hub and/or,
- remediating the threat by stopping or quarantining the infected EC2 instance.

Your TAM will provide the CloudFormation Template (CFT) to deploy the integration. When deployed, the CFT creates the following resources:

- AWS SNS notification topic: All Qualys findings are published to the AWS SNS notification topic.
- AWS Lambda: Adds additional context and metadata to the finding, publishes it to Security Hub, and/or remediates the threat.

CloudFormation > Stacks > SECURITYHUB-NOTIFICATION

Stacks (1)

SECURITYHUB-NOTIFICATION

Active < 1 >

SECURITYHUB-NOTIFICATION

SECURITYHUB-NOTIFICATION

Stack actions

Stack info | Events | Resources | **Outputs** | Parameters | Template | Change sets

Outputs (3)

Key	Value	Description	Export name
BlueHexagonNotificationTopicARN	arn:aws:sns:us-west-2:██████████:SECURITYHUB-NOTIFICATION-notification-topic	Blue Hexagon alert notification topic	BlueHexagonNotificationTopic
LambdaFunctionARN	arn:aws:lambda:us-west-2:██████████:function:bluehexagon-lambda-integration	Lambda function ARN.	BlueHexagonLambdaARN
LambdaRoleARN	arn:aws:iam:██████████:role/bluehexagon-lambda-integration-role	Role for Lambda execution.	LambdaRole

To receive findings in Security Hub from Qualys, in the **Security Hub** page, click **Integrations** > **Accept findings**.

aws Services Resource Groups Oregon Support

Security Hub

Summary
Security standards
Insights
Findings
Integrations
Settings
What's new 7

Blue Hexagon: Blue Hexagon for AWS

Description
Blue Hexagon for AWS is a deep learning based, real time threat detection platform for AWS workloads

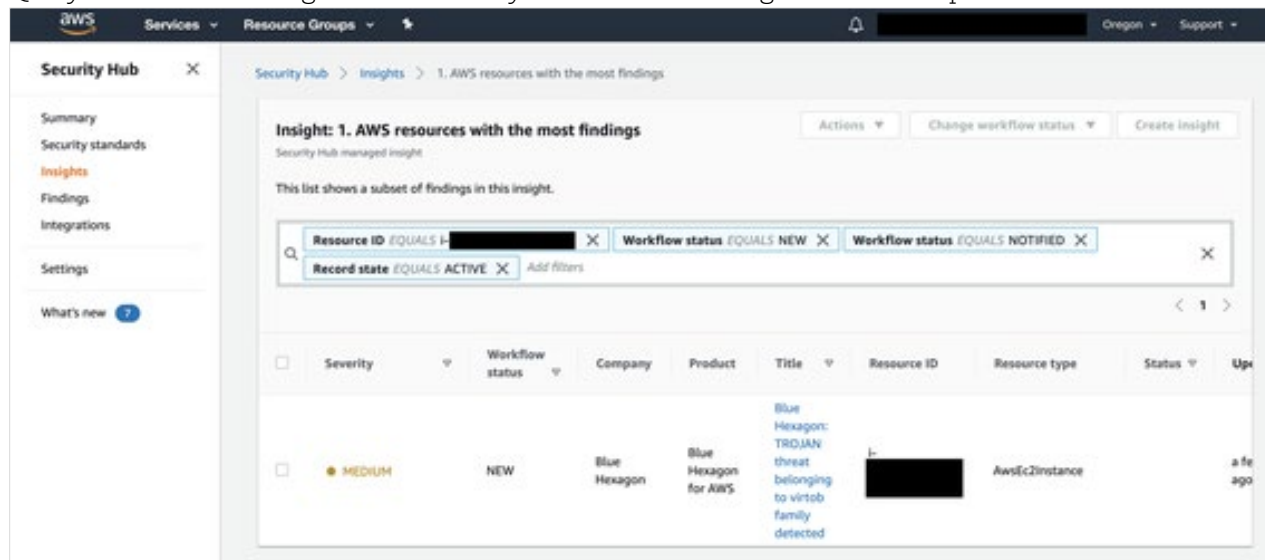
Type of integration
Sends findings to Security Hub

Categories
Network Intrusion Detection Systems (IDS), Network Intrusion Protection System (IPS), AV Scanning and Sandboxing

How to receive findings from this integration
1. Purchase a subscription to this product
2. Choose **Accept findings**

Status
☒ Not accepting findings

Qualys will send findings to the Security Hub once the integration is set up.



Creating Security VPC for HA Deployment

For Qualys HA high-availability deployment, Qualys virtual appliances should be deployed within VPCs with private subnets across several AZs (Availability Zones). If you do not already have a VPC, deploy the stack below to create a VPC with two Availability Zones and a pair of public and private subnets. This stack is an adaptation of this AWS CloudFormation template. The stack deploys an internet gateway with a default route on a public subnet. Qualys appliances are also registered with the Qualys cloud through NAT gateways (one in each AZ) and default routes in the private subnets. The stack is a faithful adaptation of [this AWS CloudFormation template](#).

Stacks in the image create only VPCs, subnets, and associated networking. As described in [Deploying Qualys CDR](#), the actual Qualys virtual appliances are deployed to this VPC.

