

---

# AWS Security Operations

## Lab Guide

Sponsored by



v1.0

# Logistics

- **Format:** Overview presentation and lab setup, followed by paced exercises with a section recap.
- **Workshop Duration:** 2 hours
- **Target Audience:** Technical security users (security engineers, architects, DevOps) who have heard of Dome9 and know what Dome9 offers
- The organizing team is comprised of one speaker and 1-2 technical staff to help out and answer questions
- Participants bring their own laptops and have an AWS account setup (preferably beforehand) - **Please do this early on since it takes a few hours for a new AWS account to sync with a CFT template**
- Participants need to download the Cloudformation (**CFT**) template to run this lab
  - Please download CFT from the [Github here](#)

# Table of Contents

- 1. Lab Overview and AWS Setup (10 -15 min)**
  - a. Walkthrough of lab environment and exercises
  - b. Exercise 1.1: Setup AWS account
  - c. Exercise 1.2: Deploy sandbox environment in AWS
- 2. AWS Security Operations Lab (30 minutes)**
  - a. Exercise 2.1: Identify zombie security group (no instance, but permissive rule)
  - b. Exercise 2.2: Identify an exposed internal asset in the AWS environment - Part 1
  - c. Exercise 2.3: Identify an exposed internal asset in the AWS environment - Part 2
  - d. Section Wrap Up
- 3. Dome9 Overview and Onboarding (10 -15 min)**
  - i. Overview of Dome9
  - ii. Exercise 3.1: Connect Dome9 to your new AWS account
- 4. AWS Security Operations Lab with Dome9 (30 min)**
  - i. Exercise 4.1: Conduct inventory/asset review
  - ii. Exercise 4.2: Visualize security architecture
  - iii. Exercise 4.3: Identify zombie security group
  - iv. Exercise 4.4: Identify publicly accessible databases (2)
  - v. Section Wrap Up
- 5. Security Posture Management Lab (20 min)**
  - i. Exercise 5.1: Enforce security policy
  - ii. Exercise 5.2: Active protection – Security group level
  - iii. Exercise 5.3: Active protection – Region level
  - iv. Section Wrap Up

**6. S3 Security Lab (20 min)**

- i. Exercise 6.1: Identify publicly exposed S3 buckets (2) using Dome9
- ii. Exercise 6.2: S3 Access Controls (ACL)
- iii. Exercise 6.3: S3 Access Controls (Bucket policies)
- iv. Section Wrap Up

**7. Offboarding (5 min)**

---

# AWS Setup

## Exercise 1.1: Setup lab environment (Login to your lab AWS account)

The instructor should provide you with an AWS Account for this workshop. Ensure you have an AWS account setup before proceeding.

### Exercise Complete!

## Exercise 1.2: Deploy sandbox AWS environment

Navigate to AWS Cloudformation

The screenshot shows the AWS Management Console with the 'Services' tab selected. The 'Compute' section is expanded, listing EC2, Lightsail, Elastic Container Service, EKS, Lambda, Batch, and Elastic Beanstalk. The 'CloudFormation' service is highlighted in orange. Other sections visible include Storage (S3, EFS, Glacier, Storage Gateway), Database (RDS, DynamoDB, ElastiCache, Neptune, Amazon Redshift), Developer Tools (CodeStar, CodeCommit, CodeBuild, CodeDeploy, CodePipeline, Cloud9, X-Ray), Analytics (Athena, EMR, CloudSearch, Elasticsearch Service, Kinesis, QuickSight, Data Pipeline, AWS Glue), Customer Engagement (Amazon Connect, Pinpoint, Simple Email Service), Business Productivity (Alexa for Business, Amazon Chime, WorkDocs, WorkMail), Security, Identity & Compliance (IAM, Cognito, Secrets Manager, GuardDuty, Inspector, Amazon Macie, AWS Single Sign-On, Certificate Manager, CloudHSM, Directory Service, WAF & Shield, Artifact), Desktop & App Streaming (WorkSpaces, AppStream 2.0), and Internet Of Things (IoT Core, IoT 1-Click, IoT Device Management, IoT Analytics, Greengrass). A search bar at the top is empty, and a navigation bar at the bottom includes History, Console Home, CloudTrail, S3, CloudFormation, IAM, and various support links.

Click on create stack and select “upload a template to S3” and choose the CFT file that you downloaded and click next

The screenshot shows the 'Create stack' wizard in the AWS CloudFormation console. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, a bell icon, D9+Bootcamp2, N. Virginia, and Support. The current path is CloudFormation > Stacks > Create Stack.

**Select Template**

Specify Details

Options

Review

**Select Template**

Select the template that describes the stack that you want to create. A stack is a group of related resources that you manage as a single unit.

**Design a template** Use AWS CloudFormation Designer to create or modify an existing template. [Learn more.](#)

**Choose a template** A template is a JSON/YAML-formatted text file that describes your stack's resources and their properties. [Learn more.](#)

- Select a sample template
- Upload a template to Amazon S3
 

[Choose File](#)
- Specify an Amazon S3 template URL
 

[No file chosen](#)

Cancel **Next**

Create a stack name such as “<yourname>LoftLab” and select us-east1-a, 1-b, and 1-c for subnetAza, subnetAzb, subnetAzc and click next

**Create stack**

Select Template

**Specify Details**

Options

Review

**Specify Details**

Specify a stack name and parameter values. You can use or change the default parameter values, which are defined in the AWS CloudFormation template. [Learn more.](#)

**Stack name**

**Parameters**

<b>AmiName</b>	<input type="text" value="ami-14c5486b"/>	Name of the AWS AMI IN THIS REGION.
<b>SubnetAZa</b>	<input type="text" value="us-east-1a"/>	First Availability Zone of the Subnets
<b>SubnetAZb</b>	<input type="text" value="us-east-1b"/>	Second Availability Zone of the Subnets
<b>SubnetAZc</b>	<input type="text" value="us-east-1c"/>	Third Availability Zone of the Subnets

Cancel **Previous** **Next**

Click next twice and select “I acknowledge that AWS Cloudformation might create resources” and click create.

Rollback Triggers

No monitoring time provided

No rollback triggers provided

Advanced

Notification	
Termination Protection	Disabled
Timeout	none
Rollback on failure	Yes

Capabilities

**i** 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.

Quick Create Stack (Create stacks similar to this one, with most details auto-populated)

Cancel Previous Create

You will need to wait 5 minutes for the CFT to automatically deploy the environment in your AWS account. At the end you should see the below screen:

**Stack Name:** demo

**Created Time:** 2018-08-18 12:36:00 UTC-0700

**Status:** CREATE\_COMPLETE

**Description:**

Event Time	Status	Type	Logical ID	Status Reason
2018-08-18 12:38:50 UTC-0700	CREATE_COMPLETE	AWS::CloudFormation::Stack	demo	
2018-08-18 12:38:47 UTC-0700	CREATE_COMPLETE	AWS::ElasticLoadBalancingV2::LoadBalancer	Alb	
2018-08-18 12:38:20 UTC-0700	CREATE_COMPLETE	AWS::EC2::Instance	RabbitMQ1	
2018-08-18 12:37:40 UTC-0700	CREATE_COMPLETE	AWS::EC2::Instance	AgentService2	
2018-08-18 12:37:40 UTC-0700	CREATE_COMPLETE	AWS::EC2::Instance	monitoring2	
2018-08-18 12:37:33 UTC-0700	CREATE_COMPLETE	AWS::EC2::Instance	appserver2	
2018-08-18 12:37:33 UTC-0700	CREATE_COMPLETE	AWS::EC2::Instance	appserver1	
2018-08-18 12:37:27 UTC-0700	CREATE_COMPLETE	AWS::ElasticLoadBalancingV2::LoadBalancer	WebLB	

**Exercise Complete!** You have now deployed the sandbox AWS environment in your account.

## AWS Security Policy Lab

### Exercise 2.1: Identify zombie security group

In your AWS account, navigate to **N.Virginia region**. Explore EC2 instances, Security Groups, IAM and other services.

The screenshot shows the AWS CloudFormation console. On the left, there's a sidebar with navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, AMIs, and Bundle Tasks. Below that is a section for Elastic Block Store (Volumes, Snapshots), Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces), Load Balancing (Load Balancers, Target Groups), Auto Scaling (Launch Configurations, Auto Scaling Groups), Systems Manager Services (Run Command, State Manager, Configuration Compliance, Automations, Patch Compliance), and Feedback.

The main area displays a table of resources under "Resource Groups". The columns are Name, Group ID, Group Name, VPC ID, and Description. The table includes rows for various security groups like Application-Load-Balancer, App1\_Servers, App2\_ApplicationServers, App2\_DB, App2\_LoadBalancers, App2\_Web, Base SG, CloudFormer-WebServer..., CloudFormer-Roy-WebServer..., and Common SG. The "Common SG" row is selected, showing its details in the right panel.

The right panel shows the details for the selected security group "sg-a3e455c8". It has tabs for Description, Inbound, Outbound, and Tags. The Inbound tab is selected, displaying a table of rules:

Type	Protocol	Port Range	Source	Description
All TCP	TCP	0 - 65535	sg-bde455cd (DB servers)	
Custom UDP Rule	UDP	151 - 162	212.25.105.39/32	
Custom UDP Rule	UDP	151 - 162	212.25.105.40/32	
Custom UDP Rule	UDP	151 - 162	sg-e4e455b1 (LB-Web)	
Custom TCP Rule	TCP	443	4.4.4.4/32	
Custom TCP Rule	TCP	443	50.50.50.50/32	
Custom TCP Rule	TCP	443	100.2.3.4/32	
Custom TCP Rule	TCP	443	4.5.77.88/32	
Custom TCP Rule	TCP	443	9.8.99.88/32	
Custom TCP Rule	TCP	443	77.66.1.2/32	
Custom TCP Rule	TCP	443	5.4.3.3/32	

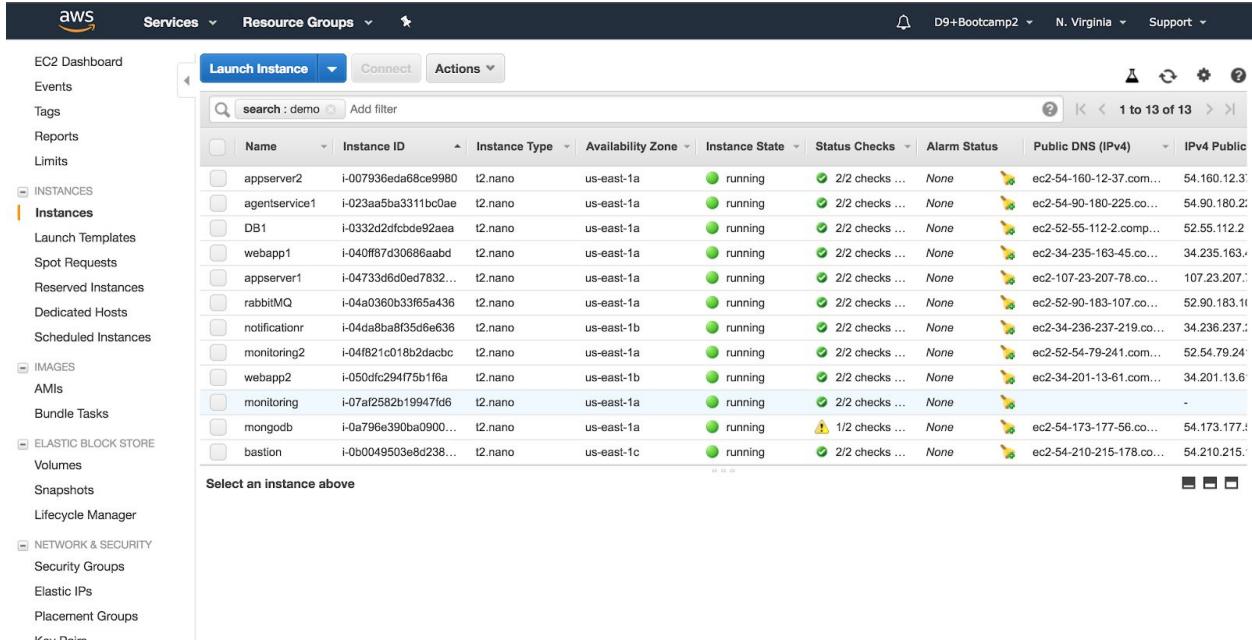
At the bottom of the page, there are links for Feedback, English (US), Copyright notice (© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.), Privacy Policy, and Terms of Use.

**Hint:** A zombie security group is a security group that has a permissive rule but has no instances tied to it!

**Exercise Complete!** You have now found your zombie policy!

## Exercise 2.2: Identify an exposed internal asset in the AWS environment - Part 1

## Dome9 Lab Guide

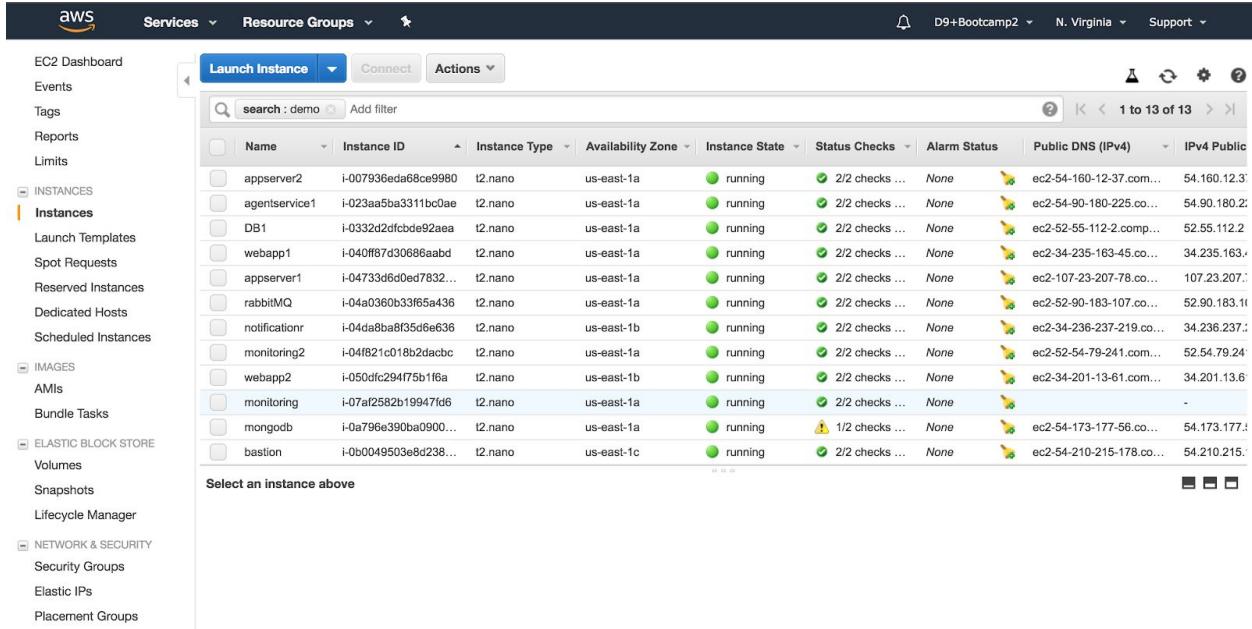


The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES (with Instances selected), Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Scheduled Instances, IMAGES (with AMIs selected), Bundle Tasks, and ELASTIC BLOCK STORE (with Volumes selected). The main area displays a table of 13 instances. The columns include Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS (IPv4), and IPv4 Public. Most instances have a green 'running' status and a green '2/2 checks ...' status check. One instance, 'mongodb', has a yellow '1/2 checks ...' status check. The Public DNS (IPv4) column shows various IP addresses, and the IPv4 Public column indicates whether they are publicly accessible.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public
appserver2	i-007936eda68ce9980	t2.nano	us-east-1a	running	2/2 checks ...	None	ec2-54-160-12-37.com...	54.160.12.3
agentservice1	i-023aa5ba3311bc0aae	t2.nano	us-east-1a	running	2/2 checks ...	None	ec2-54-90-180-225.co...	54.90.180.2
DB1	i-0332d2dfcbde92aea	t2.nano	us-east-1a	running	2/2 checks ...	None	ec2-52-55-112-2.com...	52.55.112.2
webapp1	i-040f87d30686aab0	t2.nano	us-east-1a	running	2/2 checks ...	None	ec2-34-235-163-45.co...	34.235.163
appserver1	i-04733d6d0ed7832...	t2.nano	us-east-1a	running	2/2 checks ...	None	ec2-107-23-207-78.co...	107.23.207
rabbitMQ	i-04a0360b33f85a436	t2.nano	us-east-1a	running	2/2 checks ...	None	ec2-52-90-183-107.co...	52.90.183.11
notificationr	i-04da8ba8f35d6e636	t2.nano	us-east-1b	running	2/2 checks ...	None	ec2-34-236-237-219.co...	34.236.237
monitoring2	i-04f821c018b2dacbc	t2.nano	us-east-1a	running	2/2 checks ...	None	ec2-52-54-79-241.com...	52.54.79.24
webapp2	i-050dcf294t75b1f6a	t2.nano	us-east-1b	running	2/2 checks ...	None	ec2-34-201-13-61.com...	34.201.13.6
monitoring	i-07af2582b19947fd6	t2.nano	us-east-1a	running	2/2 checks ...	None	-	-
mongodb	i-0a796e390ba0900...	t2.nano	us-east-1a	running	1/2 checks ...	None	ec2-54-173-177-56.co...	54.173.177
bastion	i-0b0049503e8d238...	t2.nano	us-east-1c	running	2/2 checks ...	None	ec2-54-210-215-178.co...	54.210.215

There is an internal asset that is exposed to the public. Can you find it?

### Exercise 2.3: Identify an exposed internal asset in the AWS environment - Part 2



This screenshot is identical to the one above, showing the AWS EC2 Instances page. It displays the same list of 13 instances with their respective details, including the instance named 'mongodb' which has a yellow '1/2 checks ...' status check and is publicly accessible.

There is another internal asset that is exposed to the public (this one is harder to find)  
Good luck!

**Exercise Complete!** You have now found your exposed assets!

## Dome9 Onboarding and Setup

### Exercise 3.1: Connect Dome9 to your new AWS account

#### Onboard an AWS account

Onboarding an AWS account involves creating policies and attaching it to roles for Dome9 to use. For simplicity, the policies and roles have already been created. Follow these steps to onboard your AWS account to Dome9.

1. On the Dome9 console navigate to **Cloud Inventory** and select Add AWS Account.



2. Select the Dome9 operation mode, **Read/Write** to be used for the account.

Select operation mode

**Monitor (Read-Only) Mode**

In the Monitor mode, Dome9 Arc can be used for visualization, monitoring and auditing, and will not modify or actively manage your cloud environment.

**Available in Monitor (Read-Only) Mode:**

- Dome9 Clarity for visualization of network security
- Change notifications
- Audit trail
- Compliance reports
- Alerts
- Policy reports

**When to Choose Monitor (Read-Only) Mode:**

- You have another source of automation to manage your policies
- You want to manage your security group rules directly, rather than delegating to Dome9

[GET STARTED!](#)

**Full-Protection (Read/Write) Mode**

In the Full-Protection(Read/Write) Mode, Dome9 Arc can be used to actively manage your security posture and enforce best practices.

**Available in Full-Protection (R/W) Mode:**

- Dynamic Access Leases - time-limited, on-demand resource access
- Security group management console to edit policies in-place
- Tamper Protection and Region Lock for active enforcement
- Reusable policy objects such as IP Lists and DNS Objects
- Dome9 Clarity for visualization of network security
- Change notifications
- Audit trail
- Compliance reports
- Alerts
- Policy reports

**When to Choose Full-Protection (R/W) Mode**

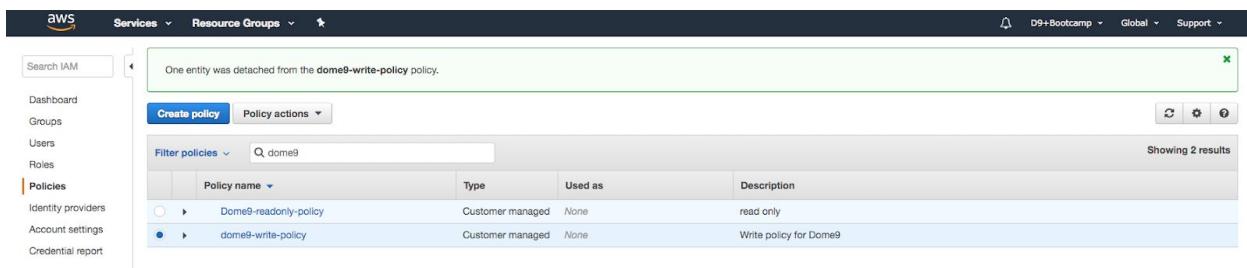
- You want to use Dome9 Arc as your system of authority for security management
- You want to use Dome9's active management and enforcement capabilities to maintain a closed-by-default security posture

Note that even when you are using Dome9 **Full-Protection (Read/Write) Mode** you'll still be able to set individual security groups to **Monitor (Read-Only) Mode**

[GET STARTED!](#)



3. Click next again
4. Sign to the AWS console ([aws.amazon.com](https://aws.amazon.com)) in a new browser tab or window  
(keep the Dome9 console open, as you will be switching between the two in the following steps).
5. Click **Services** and select the **IAM**
6. Select **Policies** and search for Dome9 and you should see two policies created.  
Click and review the policy for **dome9-write**



Policy name	Type	Used as	Description
Dome9-readonly-policy	Customer managed	None	read only
dome9-write-policy	Customer managed	None	Write policy for Dome9

7. In the AWS console, click **Roles** and “**Create new Role**”
8. Select Role Type: ‘**Another AWS Account**‘, under options mark the ‘**Required External ID**‘ option.
9. Enter the following:
  - AccountId: 634729597623
  - External ID: E+7NvdTUqNKZNoCSQ0L53@64
  - Require MFA: NOT checked

**Create role**

**Select type of trusted entity**

- AWS service** EC2, Lambda and others
- Another AWS account** Belonging to you or 3rd party
- Web identity** Cognito or any OpenID provider
- SAML 2.0 federation** Your corporate directory

Allows entities in other accounts to perform actions in this account. [Learn more](#)

**Specify accounts that can use this role**

Account ID\*  ⓘ

Options  Require external ID (Best practice when a third party will assume this role)

You can increase the security of your role by requiring an optional external identifier, which prevents "confused deputy" attacks. This is recommended if you do not own or have administrative access to the account that can assume this role. The external ID can include any characters that you choose. To assume this role, users must be in the trusted account and provide this exact external ID. [Learn more](#)

**External ID**

E+7NvdTUqNKZNoCSQ0L53@64

**Important:** The console does not support using an external ID with the Switch Role feature. If you select this option, entities in the trusted account must use the API, CLI, or a custom federation proxy to make cross-account iam:AssumeRole calls. [Learn more](#)

Require MFA ⓘ

\* Required

[Cancel](#) [Next: Permissions](#)

10. **READ-WRITE:** Make sure the following policies are selected:

- **SecurityAudit** (AWS managed policy).
- **AmazonInspectorReadOnlyAccess** (AWS managed policy).
- **dome9-write-policy**, that you created before. You can search for 'dome9' in the filter

11. Set Role Name with your choice ('Dome9-Connect' makes sense) and click on '**Create Role**'

**Create role**

**Review**

Provide the required information below and review this role before you create it.

**Role name\*** Dome9-Connect  
Use alphanumeric and '+=\_,@-' characters. Maximum 64 characters.

**Role description**

Maximum 1000 characters. Use alphanumeric and '+=\_,@-' characters.

**Trusted entities** The account 634729597623

**Policies**

- SecurityAudit
- AmazonInspectorReadOnlyAccess
- Dome9-readonly-policy

**Permissions boundary** Permissions boundary is not set

\* Required      Cancel      Previous      **Create role**

12. Copy the **Role ARN** value, and enter it in the **Role ARN** field in the Dome9 console.

## Dome9 Lab Guide

**Summary**

Policy Dome9-readonly-policy has been attached for the Dome9-Connect.

Role ARN	arn:aws:iam::371771556915:role/Dome9-Connect
Role description	Dome9 connect   Edit
Instance Profile ARNs	
Path	/
Creation time	2018-07-13 14:18 PDT
Maximum CLI/API session duration	1 hour Edit
Give this link to users who can switch roles in the console	
<a href="https://signin.aws.amazon.com/switchrole?roleName=Dome9-Connect&amp;account=371771556915">https://signin.aws.amazon.com/switchrole?roleName=Dome9-Connect&amp;account=371771556915</a>	

**Permissions**    **Trust relationships**    **Access Advisor**    **Revoke sessions**

▼ Permissions policies (3 policies applied)

Policy name	Policy type	X
Dome9-readonly-policy	Managed policy	X
SecurityAudit	AWS managed policy	X
AmazonInspectorReadOnlyAccess	AWS managed policy	X

Add AWS Cloud Account

Operation Mode > Prepare Policy > **Create Role** > Review

Onboarding Troubleshooting

Create IAM Role for Dome9

- Click on 'Roles' and 'Create new Role'
- Select: 'Role for Cross-Account Access' and select 'Provide access between your AWS account and a 3rd party AWS account'
- Enter
  - AccountID: 634729597623
  - External ID:
  - Require MFA: NOT checked
- Click on the 'Next Step' button
- Select the following policies:
  - 'SecurityAudit' (AWS managed policy)
  - 'dome9-readonly-policy' That we created before. You can search for 'dome9' in the filter
  - 'dome9-write-policy' That we created before
- Click on the 'Next Step' button
- Set Role Name with your choice ('Dome9-Connect' makes sense) and click on 'Create Role'
- On the search box look for the "Role name" you set in the previous step, and click on it.
- Copy the Role ARN and fill it in the Role ARN field
- Click on 'FINISH'

Display Name optional

Role ARN

External Id

Adding a GovCloud Account?

BACK FINISH

13. Click 'Finish'

14. Review the onboarded cloud account summary.
15. At the end of the onboarding process all the Security Groups will be in **Read-Write** mode.

**Exercise Complete!** You have now connected your Dome9 to your AWS account

## Security Architecture Review Lab

### Exercise 4.1: Conduct inventory/asset review

#### Switch into your Dome9 console for the remaining part of this module

Dome9 presents a single console view of all your assets, on all platforms, from which you can search or filter for specific assets of interest, and see details about their security posture. In this section of the Dome9 console, you can see a summary of all the assets in your VPCs that are protected by Dome9. These assets can include, for example, instances (such as EC2s), RDSs, and load balancers. Dome9 fetches information about these assets from the cloud platforms (AWS, Azure, Google) and presents it in a console view.

##### 1) View your Protected Assets

The main page shows a list of your assets that are protected by Dome9, organized by region. Filter the list using the filters on the left, or search for assets by name in the search bar.

The screenshot shows the Dome9 Cloud Inventory interface. At the top, there's a navigation bar with links for Compliance & Governance, Network Security, IAM Safety, and Administration. Below the navigation bar is a search bar and a filter section titled "Protected Assets". The filter section includes dropdowns for "Asset type" (AWS Instance, AWS Application Load Balancer, AWS ELB, AWS Lambda Function), "Tags" (Key and Value fields, checked filters for Server tags and VPC tags), and "Publicly Accessible" (Yes or No). The main area displays a table of assets under the heading "AWS > N. Virginia > vpc-034dca60511e63fdb". The table columns are Asset Type, Name, Instance Type, IP Address, and a small icon. There are 16 assets listed, including various AWS services like DB instances, Lambda functions, and EC2 instances.

Asset Type	Name	Instance Type	IP Address	Action
AWS Instance (13)	DB1 (i-0d01aeaf78636450)	t2.nano	35.153.69.123	
AWS Application Load Balancer (1)	Demo-Alb-17VXX5ZA6YMSG			
AWS ELB (1)	Demo-LambdaFunction-ENNM13XjYAHs (Demo-LambdaFunction-ENNM13XjYAHs)			
AWS Lambda Function (1)	bastion (i-045af36cd0177579)	t2.nano	52.91.193.105	
	i-0001dddee76c823f2	t2.nano	18.207.148.247	
	i-02d7d384ed59b788	t2.nano	18.206.98.146	
	i-042df90b73cde7307	t2.nano	54.205.191.132	
	i-04689aebcc4e90925	t2.nano	34.235.181.23	
	i-06a2c6dfcabca5e00	t2.nano	34.235.112.87	
	i-0a16b79a59b689265	t2.nano	35.173.254.41	
	i-0a432204a78cefced	t2.nano	54.157.1.94	
	i-0dcba333e73b9a5f28	t2.nano		
	notificationr (i-0d46241afedaad0c)	t2.nano	18.207.107.228	
	prod-web-lb			
	webapp1 (i-00fc7a9db5ecfd9fe)	t2.nano	54.144.35.221	
	webapp2 (i-013753ab74d09c193)	t2.nano	54.236.38.208	

For each asset in the list, the type, and its external IP address (if it has one) are shown.

Click on one of the assets in the list to see more details for it.

Please enter how many total EC2 instances are displayed in the google form.

## 2) View your Security Groups

The main page shows a list of all your managed security groups, in all your Dome9 managed accounts, on all cloud providers.

Filter the list using the search box or filter options on the left. You can filter by account, VPC, cloud region, protection method (full, read-only), and the number of instances or alerts.

The screenshot shows the Dome9 Cloud Inventory interface. At the top, there are tabs for Cloud Inventory, Compliance & Governance, Network Security, IAM Safety, and Administration. The Network Security tab is selected. Below the tabs, there's a search bar and a filter section with dropdowns for Region, VPC, and Tags. The main area displays two tables of security group details.

AWS > N. Virginia > productionVPC		Instances	Alerts
Demo-AgentServiceSG-1VHM88JQ2ZOB		2	2
Demo-DBServersSG-NFSU0AOWT5RY		1	0
Demo-DevopSG-13SQSHJNV9JV		0	1
Demo-LambdaSG-1LAMFLCQQ8CJ6		0	0
Demo-MQSG-1UHK4GOBMVPP		1	1
Demo-MonitoringSG-1OE9BZ4APK6R4		1	2
Demo-NotificationServerSG-VPSD01C706WE		1	0
Demo-WebMonitorSG-55MGJUL28QDG		1	1
Demo-WebappSG-SYSUWBJB2BRR		2	1
Demo-appserverSG-169ORD4N0Y8JE		2	1
Demo-default-1AN3SSLBMTN72B		1	0
default		0	0
prod-alb-sg		0	0
prod-bastion-je-sg		1	0
prod-mongo-sg		1	1
prod-webapp-elb-sg		0	0

AWS > N. Virginia > vpc-a39d63d9		Instances	Alerts
default		0	0

Please enter how many total SGs are displayed in the google form.

**Exercise Complete!** You have now explored your AWS instances and reviewed your inventory within the protected asset view of Dome9.

## Exercise 4.2: Visualize security architecture

Dome9 Clarity gives a graphical visualization of the security groups in your cloud environment, and their effects on the cloud assets in the environment. It shows the security groups, traffic sources, and permitted traffic paths in the cloud network. The view is organized logically, according to the level of exposure of the Security Group to the external world. You can use Clarity to analyze your cloud network for security issues such as access to sensitive components from the internet, or to troubleshoot it for connectivity issues such as blocked paths to components.

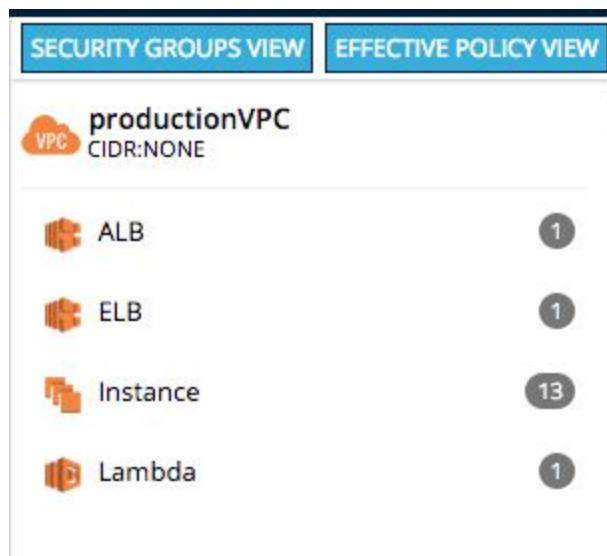
## 1) Select a cloud environment

Select **Clarity** from the main menu. A list of your cloud accounts is shown on the left.

## 2) View an environment with the Security Group view

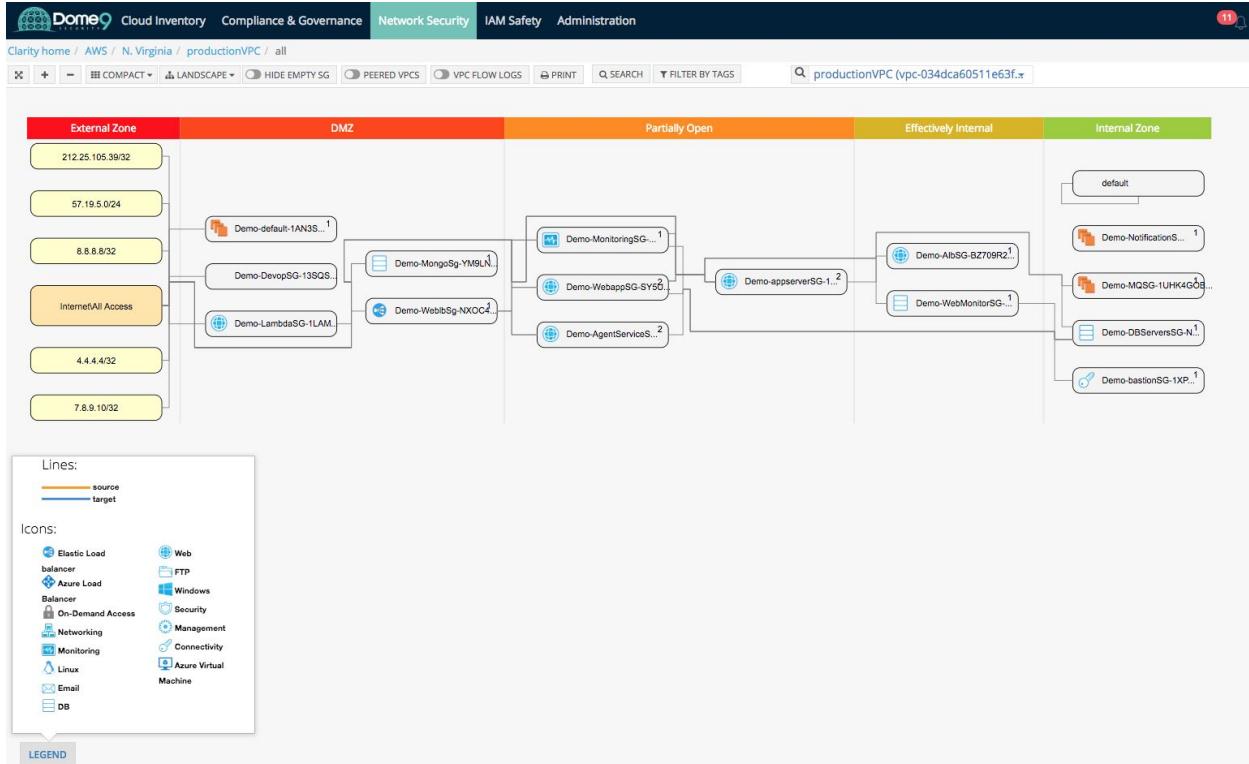
In this step, an environment will be visualized with the Security Group view. This view shows all the Security Groups.

- In Clarity, select a cloud environment in one of your accounts (in the previous section), and then select **Security Groups** from the list in the menu bar on the upper right.

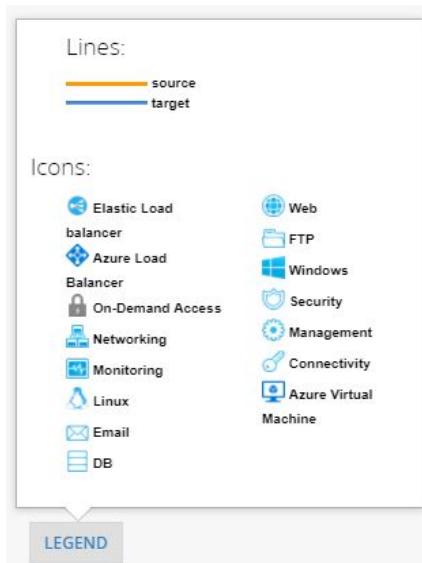


You can tell how the security groups (SG) interact with each other and can now understand whether any internal assets are communicating with the public world based on their inbound and outbound rules. The different swimlanes (which are auto classified by Dome9, so customers don't need to manually create them) represent security groups with various levels of exposure to the public.

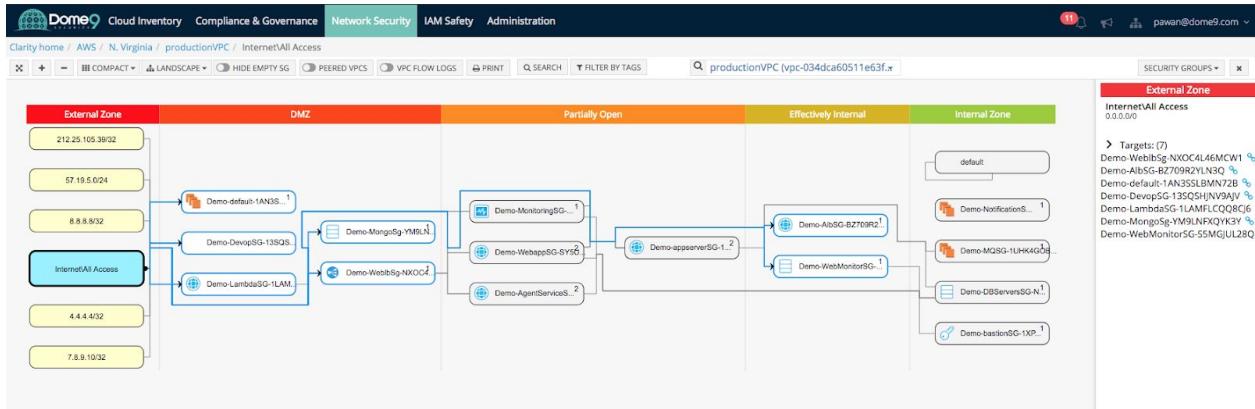
## Dome9 Lab Guide



b) Click the **Legend** button to show what each icon represents.



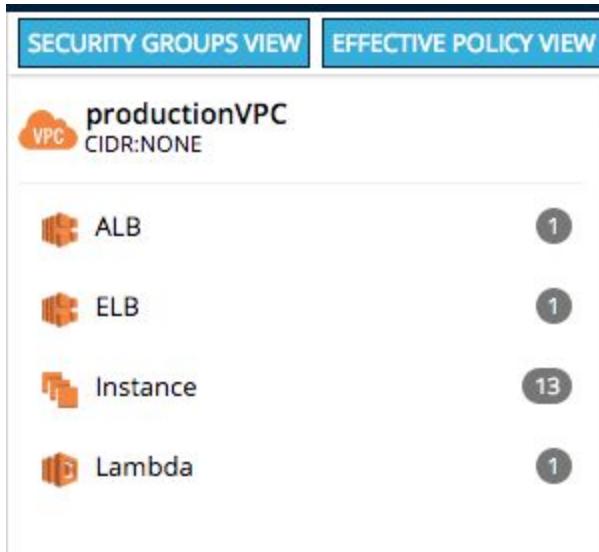
c) Click on **Internet/All Access** block. The source block is highlighted in the view, and the Security Groups that affect this source are highlighted.



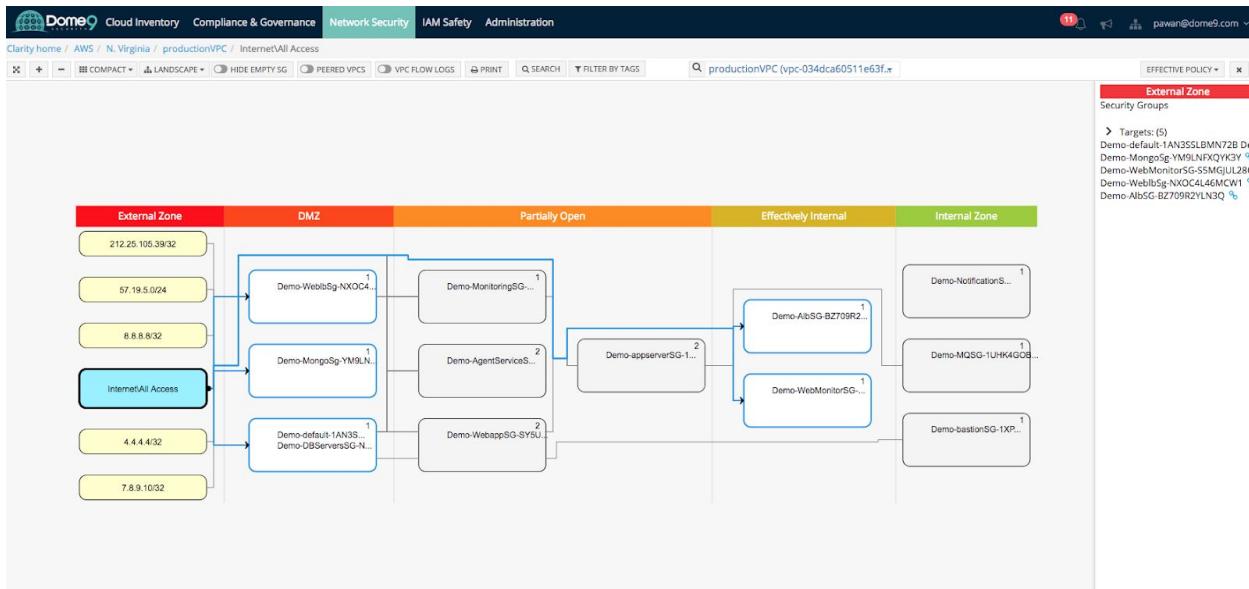
### 3) View an environment in the Effective Policy view

The Effective Policy view groups Security Groups that affect a common asset, and hides those that do not affect any assets.

- Select **Effective Policy** in the list in the menu bar at the upper right.



- This shows the VPC in the Effective Policy view



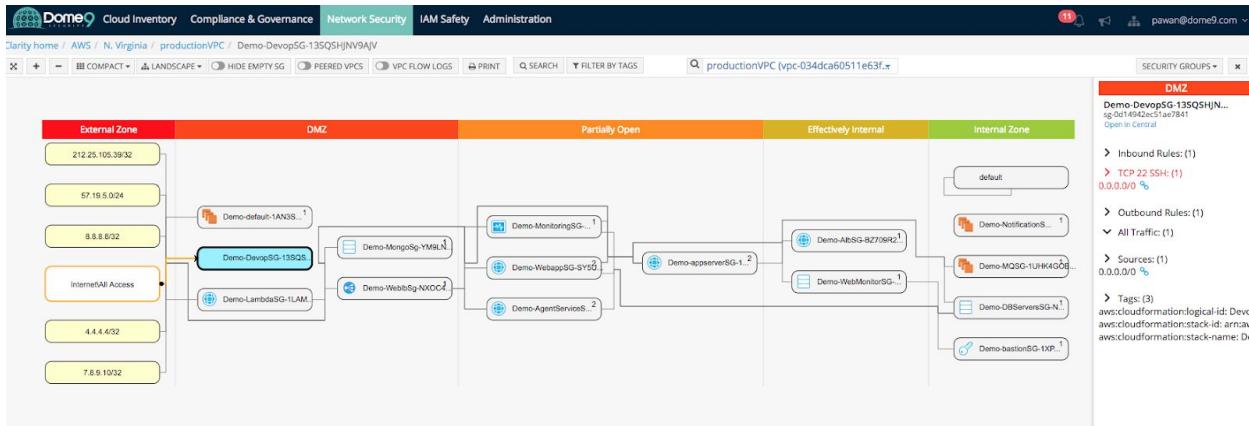
- c) This view also shows the Security Groups and Sources, organized by zone. In this view, however, the Security Groups that affect the same asset are grouped together. Security Groups that affect a number of assets may appear several times in the view. Security Groups that do not affect any assets are hidden.

**Exercise Complete!** You have visualized your AWS security configurations in the VPC from a logical firewall view (SG view) and instance policy view (effective policy view)

### Exercise 4.3: Identify zombie security group

Let's go back into Clarity's security group view. See if you can identify a zombie security group. This is a SG that has no instance attached to it but has an exposed policy (TCP 22 0.0.0.0/0) to the public.

**Answer: Don't peek just yet..**



This screenshot shows the detailed configuration of the security group 'Demo-DevopSG-13SQSHJNV9AJV'. The 'Tags' section lists three tags: 'aws:cloudformation:logical-id' (Value: DevopSG), 'aws:cloudformation:stack-id' (Value: arn:aws:cloudformation:us-east-1:290175429794:stack/Demo/052a6500-95af-11e8-a469-500c286e1a36), and 'aws:cloudformation:stack-name' (Value: Demo). The 'Inbound Services' table shows one rule: 'SSH' on 'TCP:22' from '0.0.0.0/0' with a status of 'Open'. The 'Outbound Services' section shows 'All traffic'. The 'Group Members' and 'Referencing Security Groups' sections are currently empty. The sidebar on the left provides navigation links for Visualize in Clarity, VPC Flow Logs, and Alerts.

This is a low severity issue yet is important to be aware of as it is just one click away from exposing your internal servers. With Clarity you can find such issues and take appropriate actions to fix such misconfigurations.

**Exercise Complete!** You have now identified the zombie security group exposure

#### Exercise 4.4: Identify publicly accessible databases

In this section, we will investigate and find 2 different database exposures. Let's jump into Clarity.

## Challenge 1: Detect DB Exposure – part 1 (easy)

In Clarity, try to find an exposed DB:

**Answer:** Here you see that this SG has 0.0.0.0/0 on port 27017 associated with the mongoDB instance. This is a high severity issue as one of your internal DB servers is now wide open. Click on the SG for more details.

The screenshot shows a network diagram with nodes representing AWS services and security groups. The nodes are color-coded by zone: External Zone (yellow), DMZ (orange), Partially Open (light orange), Effectively Internal (light green), and Internal Zone (light blue). A specific security group, 'Demo-MongoSg-YM9LNFXQYK3Y', is highlighted in the DMZ zone. The diagram illustrates various traffic paths and security rules between these components.

**AWS Security Group: Demo-MongoSg-YM9LNFXQYK3Y (sg-079fdbbeb93b019f17)**

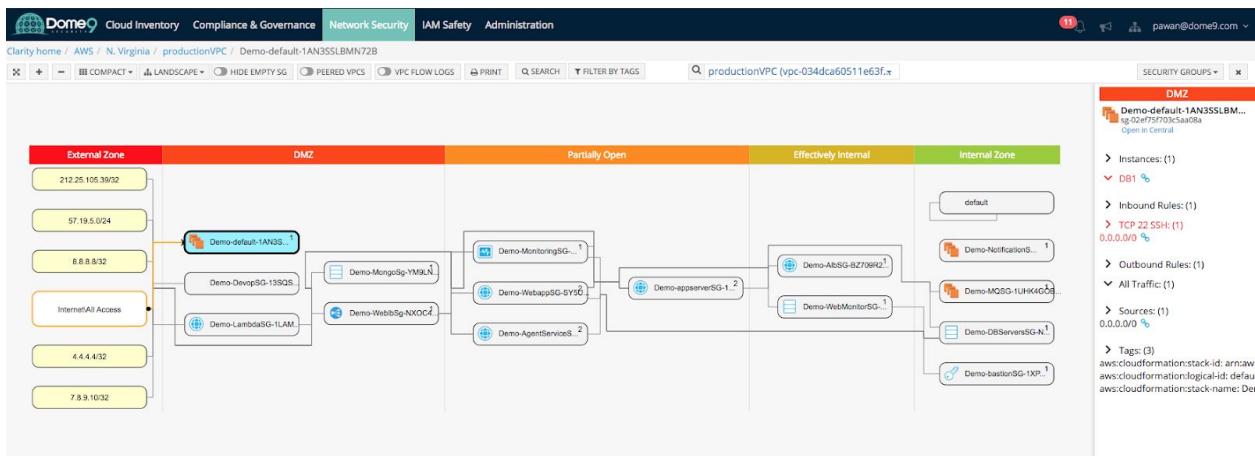
Account	AWS	Region	N. Virginia (us_east_1)	VPC	productionVPC (vpc-034dca60511e63fb)	Group Description	MongoDB security group	Related Links	<a href="#">Visualize in Clarity</a>	<a href="#">VPC Flow Logs</a>	<a href="#">Alerts</a>																																					
History (Last 7 Days)	10:19 AM - 10:24 AM	Aug 1, 2018	Showing 1/1																																													
<b>Tags</b> <table border="1"> <thead> <tr> <th>Key</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>env</td> <td>prod</td> </tr> <tr> <td>aws:cloudformation:stack-id</td> <td>arn:aws:cloudformation:us-east-1:290175429794:stack/Demo/052a6500-95af-11e8-a469-500c286e1a36</td> </tr> <tr> <td>Name</td> <td>prod-mongo-sg</td> </tr> <tr> <td>aws:cloudformation:logical-id</td> <td>MongoSg</td> </tr> <tr> <td>aws:cloudformation:stack-name</td> <td>Demo</td> </tr> </tbody> </table> <b>Inbound Services</b> <table border="1"> <thead> <tr> <th>Name</th> <th>Dome9 Description</th> <th>Protocol/Port</th> <th>State</th> <th>Allowed Sources</th> </tr> </thead> <tbody> <tr> <td>Custom TCP (27017)</td> <td></td> <td>TCP:27017</td> <td>Open</td> <td></td> </tr> <tr> <td>MS-SQL</td> <td></td> <td>TCP:1433</td> <td>Partial</td> <td>Demo-LambdaSG-1LAMFLCQQ8CJ6</td> </tr> </tbody> </table> <b>Outbound Services: Default</b> <table border="1"> <thead> <tr> <th>Group Members</th> </tr> </thead> <tbody> <tr> <td> <table border="1"> <thead> <tr> <th>State</th> <th>Name</th> <th>Type</th> <th>IP Address</th> </tr> </thead> <tbody> <tr> <td>Up</td> <td>(i-042df90b73cde7307)</td> <td>t2.nano</td> <td>54.205.191.132 10.10.4.46</td> </tr> </tbody> </table> </td> </tr> </tbody> </table> <b>Referencing Security Groups: No referencing</b>												Key	Value	env	prod	aws:cloudformation:stack-id	arn:aws:cloudformation:us-east-1:290175429794:stack/Demo/052a6500-95af-11e8-a469-500c286e1a36	Name	prod-mongo-sg	aws:cloudformation:logical-id	MongoSg	aws:cloudformation:stack-name	Demo	Name	Dome9 Description	Protocol/Port	State	Allowed Sources	Custom TCP (27017)		TCP:27017	Open		MS-SQL		TCP:1433	Partial	Demo-LambdaSG-1LAMFLCQQ8CJ6	Group Members	<table border="1"> <thead> <tr> <th>State</th> <th>Name</th> <th>Type</th> <th>IP Address</th> </tr> </thead> <tbody> <tr> <td>Up</td> <td>(i-042df90b73cde7307)</td> <td>t2.nano</td> <td>54.205.191.132 10.10.4.46</td> </tr> </tbody> </table>	State	Name	Type	IP Address	Up	(i-042df90b73cde7307)	t2.nano	54.205.191.132 10.10.4.46
Key	Value																																															
env	prod																																															
aws:cloudformation:stack-id	arn:aws:cloudformation:us-east-1:290175429794:stack/Demo/052a6500-95af-11e8-a469-500c286e1a36																																															
Name	prod-mongo-sg																																															
aws:cloudformation:logical-id	MongoSg																																															
aws:cloudformation:stack-name	Demo																																															
Name	Dome9 Description	Protocol/Port	State	Allowed Sources																																												
Custom TCP (27017)		TCP:27017	Open																																													
MS-SQL		TCP:1433	Partial	Demo-LambdaSG-1LAMFLCQQ8CJ6																																												
Group Members																																																
<table border="1"> <thead> <tr> <th>State</th> <th>Name</th> <th>Type</th> <th>IP Address</th> </tr> </thead> <tbody> <tr> <td>Up</td> <td>(i-042df90b73cde7307)</td> <td>t2.nano</td> <td>54.205.191.132 10.10.4.46</td> </tr> </tbody> </table>	State	Name	Type	IP Address	Up	(i-042df90b73cde7307)	t2.nano	54.205.191.132 10.10.4.46																																								
State	Name	Type	IP Address																																													
Up	(i-042df90b73cde7307)	t2.nano	54.205.191.132 10.10.4.46																																													

You can see the open port in this view. We will fix these issues in the next lab.

## Challenge 2: Detect DB Exposure – part 2 (hard)

Find another exposed DB, this one is a bit tricky.

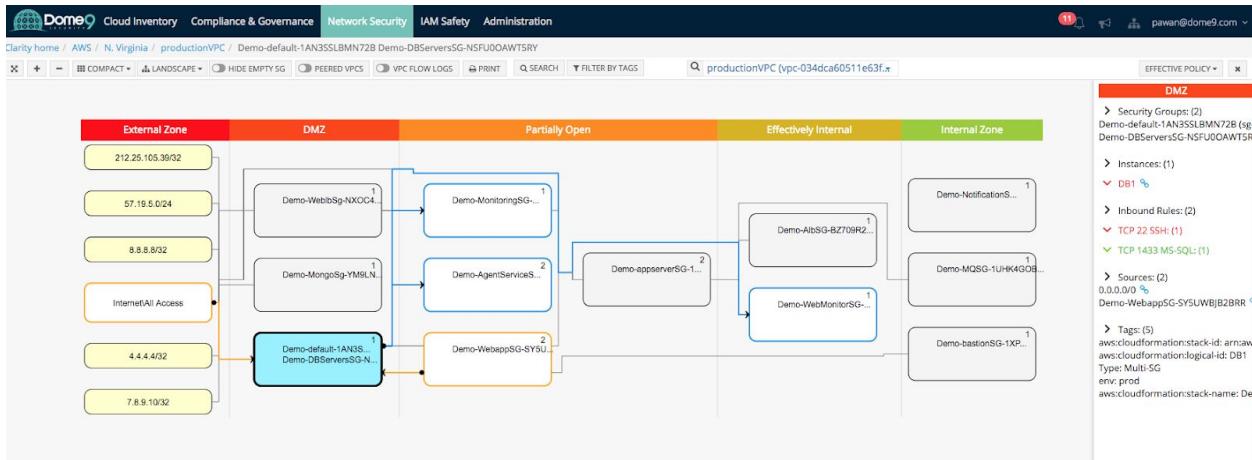
**Answer:** Click the default SG and hover over the right. Here you see default SG with SSH wide open and that DB1 is associated with this default SG making the database exposed to the public



In the effective policy view below – you can see default SG is in the DMZ zone and has one instance assignment (DB1, which is also part of the DB servers group).

Digging deeper, we realize that even if the rules associated with the security group are correct, an instance can still be assigned to the wrong security group. This is due to a misconfiguration that occurred due to assignment of multiple security groups, and specifically an incorrect default SG to DB1 instance.

The effective policy view brings this to light and tells you what security groups an instance belongs to, and therefore what the effective security policy is. Now it is clear – DB1 is exposed because it belongs to not only the internal DB Servers SG, but also to the Default SG which is in the DMZ.



**Exercise Complete!** You have now investigated both database exposures in Clarity

## Security Posture Management Lab

### Exercise 5.1: Enforce security policy

Fix misconfiguration in default security group by deleting the open SSH rule. For the purpose of this lab, lets keep it completely closed.

Now back in **Dome9 console** - go to Clarity, click the default SG we saw as the culprit, and turn on full protection mode at the top right in the entity explorer view:

The screenshot shows the AWS Security Group configuration page for 'Demo-default-1AN3SSLBMN72B'. The page includes the following sections:

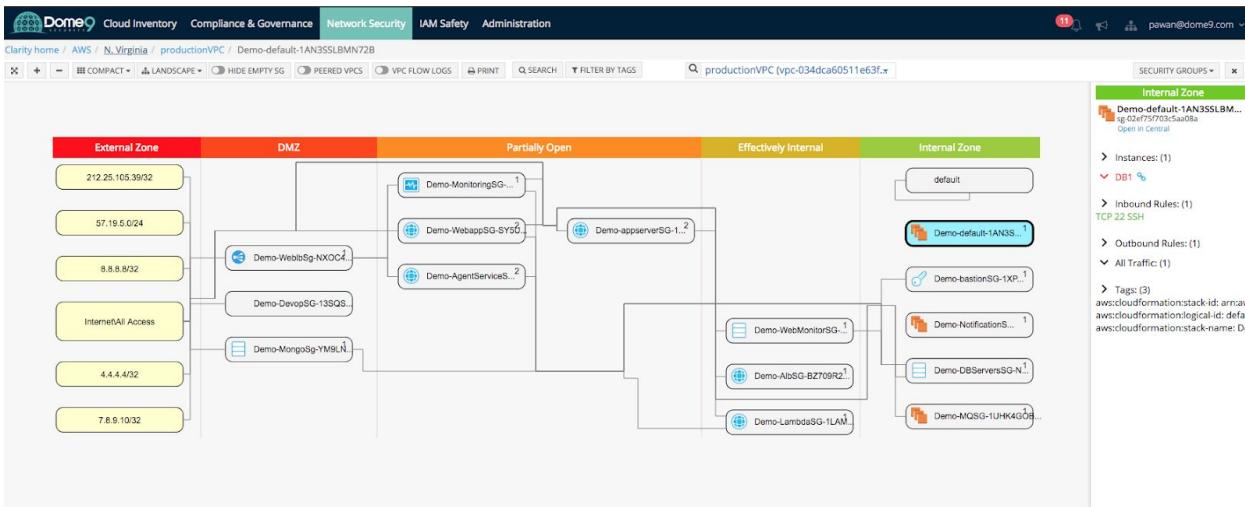
- Tags:** Shows tags for the security group, such as 'aws:cloudformation:stack-id' and 'aws:cloudformation:logical-id'.
- Inbound Services:** Shows an inbound rule for 'SSH' on 'TCP.22' with 'Open' as the state.
- Group Members:** Shows a member named 'DB1' (with ID '0d01aeaa78636450') which is an 't2.nano' instance with IP '35.153.69.123' and '10.10.4.151'.

A green success message at the top right says 'Protection Mode Changed Successfully'.

Fix the issue by closing the SG for now

The screenshot shows the AWS Security Groups console. A specific security group, "Demo-default-1AN3SSLBMN72B", is selected. The inbound rules tab is active, showing one rule: "SSH" (Protocol: TCP, Port Number: 22, Port Range: 22, Open for All: Limited, Source: On-Demand). The "Allow On-Demand connections from authorized users (Dynamic-Access)" checkbox is checked.

You can also check Clarity to see the correct topology below:



**Exercise Complete!** You have now fixed the misconfigured security group.

### Exercise 5.2: Active protection – SG level

Let's try to mess with this by making changes in the AWS console. Make SG change in AWS by going to the default SG.

The screenshot shows the AWS EC2 Dashboard with the 'Security Groups' section selected. A table lists 17 security groups, each with a checkbox, Name, Group ID, VPC ID, and Description. The 'Default security group' (sg-02ef75f703c5aa08a) is highlighted. Below the table, a modal window titled 'Edit inbound rules' is open, showing a single rule for SSH traffic from 0.0.0.0/0 to port 22.

Name	Group ID	Group Name	VPC ID	Description
sg-01b79ee69511b6bfd	sg-01b79ee69511b6bfd	Demo-MonitoringSG-1OE9B...	vpc-034dca60511e63fdb	Monitoring security group
sg-02b4dba458cc41f67	sg-02b4dba458cc41f67	Demo-WebMonitorSG-S5M...	vpc-034dca60511e63fdb	DB Servers security group
sg-02ef75f703c5aa08a	sg-02ef75f703c5aa08a	Demo-default-1AN3SSLBM...	vpc-034dca60511e63fdb	Default security group
sg-0382fe9f809d57409	sg-0382fe9f809d57409	Demo-MQSG-1UHK4GOBM...	vpc-034dca60511e63fdb	MQ security group
sg-04bce6c6c60628e42	sg-04bce6c6c60628e42	Demo-NotificationServerSG-...	vpc-034dca60511e63fdb	Notification server security group
sg-079430790d9f62542	sg-079430790d9f62542	Demo-AgentServiceSG-1VH...	vpc-034dca60511e63fdb	Agent Service security group
sg-0a2d3e1098f39a7bf	sg-0a2d3e1098f39a7bf	Demo-appserverSG-169OR...	vpc-034dca60511e63fdb	Agent Service security group
sg-0a66e77d558bb9d28	sg-0a66e77d558bb9d28	default	vpc-034dca60511e63fdb	default VPC security group
sg-0bf44ec5380e4a6e	sg-0bf44ec5380e4a6e	Demo-DBServersSG-NSFU...	vpc-034dca60511e63fdb	DB Servers security group
sg-0bf8009d10955602c	sg-0bf8009d10955602c	Demo-WebappSG-SY5UWB...	vpc-034dca60511e63fdb	WebApp security group
sg-0d14942ec51ae7841	sg-0d14942ec51ae7841	Demo-DevopsSG-13SQSHJN...	vpc-034dca60511e63fdb	Devops security group
sg-0eba43cc935f98779	sg-0eba43cc935f98779	Demo-LambdaSG-1LAMFL...	vpc-034dca60511e63fdb	Lambda Function security group
sg-a2ca3fe8	sg-a2ca3fe8	default	vpc-a39d63d9	default VPC security group

**Security Group: sg-02ef75f703c5aa08a**

**Inbound**

Type: SSH | Protocol: TCP | Port Range: 22 | Source: 0.0.0.0/0 | Description: e.g. SSH for Admin Desktop

This security group has no rules

And now enable 0.0.0.0/0 SSH rule for default SG

The dialog box 'Edit inbound rules' is open. It shows a table with columns: Type, Protocol, Port Range, Source, and Description. A new rule is being added with the following details:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom	0.0.0.0/0

**Add Rule**

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

You should now be able to see the change in Dome9 console. Dome9 also rolls configuration back and adds the user event in the audit trail for further analysis.

You can see the activity trail in the audit log which is extracted context from CloudTrail

Events				
Timestamp	Origin	User / IP Address	Description	
2018-07-12 19:07:56	Dome9 Audit	system	<b>Security group tamper detected and handled</b> Security group tamper detected (Group modified). Security group: 'Demo-DBServersSG-1OXTU4VY4ZFT (sg-0bd91e7cf547a00ef)' of 'AWS>us_east_1>vpc-03de4c22bec2c90d3'. Handling method: Dome9 system has overriden the new settings with the approved policy. The following inbound rules were discovered: TCP-22-22: 0.0.0.0/0.	<a href="#">Details</a>

You should see the configuration reversed back to the gold standard configuration

The screenshot shows the AWS CloudFormation console for a security group named 'Demo-default-13LNEK50ZUU60' (sg-049f4f60ef0f24aab). The configuration includes:

- Tags:**
  - aws:cloudformation:logical-id: default
  - aws:cloudformation:stack-id: arn:aws:cloudformation:us-east-1:426895540081:stack/Demo3aae12d0-8637-11e8-94c9-500c20fcfa99
  - aws:cloudformation:stack-name: Demo
- Inbound Services:**

Name	Dome9 Description	Protocol/Port	State	Allowed Sources
SSH		TCP:22	Closed	<a href="#">GET ACCESS</a> <a href="#">EDIT</a> <a href="#">DELETE</a>
- Outbound Services:** Default
- Group Members:**

State	Name	Type	IP Address
Open	DB1 (i-0bda72c518af8bc00)	t2.nano	18.209.214.249 10.10.4.33

You can also confirm it in the AWS console

The screenshot shows the AWS EC2 Dashboard with the 'Create Security Group' button highlighted. On the left, a sidebar lists various AWS services like Instances, AMIs, and Network & Security. Under Network & Security, 'Security Groups' is selected. The main pane displays a table of security groups:

Name	Group ID	Group Name	VPC ID	Description
sg-01b79ee69511b6bf		Demo-MonitoringSG-10EB9...	vpc-034dca60511e63fdb	Monitoring security group
sg-02b4dba458cc41f67		Demo-WebMonitorSG-S5M...	vpc-034dca60511e63fdb	DB Servers security group
sg-02ef75f703c5aa08a		Demo-default-1AN3SSLBM...	vpc-034dca60511e63fdb	Default security group
sg-0382fe9f809d57409		Demo-MQSG-1UHK4GOBM...	vpc-034dca60511e63fdb	MQ security group
sg-04bce6cc60628e42		Demo-NotificationServerSG...	vpc-034dca60511e63fdb	Notification server security group
sg-079430790d9f62542		Demo-AgentServiceSG-1VH...	vpc-034dca60511e63fdb	Agent Service security group
sg-0a2d3e1098f39a7bf		Demo-appserverSG-1690R...	vpc-034dca60511e63fdb	Agent Service security group
sg-0a66e77d65bb9d28		default	vpc-034dca60511e63fdb	default VPC security group
sg-0b4f44ec5380e4a6e		Demo-DBServersSG-NSFU...	vpc-034dca60511e63fdb	DB Servers security group
sg-0b8009410955602c		Demo-WebappSG-SY5UWB...	vpc-034dca60511e63fdb	WebApp security group
sg-0d14942ec51ae7841		Demo-DevopSG-13SQSHJN...	vpc-034dca60511e63fdb	Devops security group
sg-0eba43cc93598779		Demo-LambdaSG-1LAMFL...	vpc-034dca60511e63fdb	Lambda Function security group
sg-a2ca3fe8		default	vpc-a39d63d9	default VPC security group

Below the table, a message says "This security group has no rules". The 'Inbound' tab is highlighted.

**Exercise Complete!** You have now enabled enforcement of security policies and eliminate configuration drift at a security group level in AWS

### Exercise 5.3: Active protection – Region level

Let's turn on Region Lock first in Dome9 (Lock down Sao Paolo region)

Go to the cloud inventory and select cloud accounts and navigate to Sao Paolo region.

## Dome9 Lab Guide



**Account Number**  
290175429794

**Added At**  
Aug 1, 2018 10:24 AM

**Total Instances**  
13

**Total Full Protection Security Groups**  
1

**Total Read-Only Security Groups**  
31

Search for Region				
Region	Detection Mode	Instances	Security Groups	
Canada Central	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
Frankfurt	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
Ireland	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
London	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
Mumbai	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
N. California	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
N. Virginia	Read-Only (Monitor mode) ⓘ	13	1 Full protection 16 Read-Only	<button>Edit</button>
Ohio	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
Oregon	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
Paris	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
Seoul	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
Singapore	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
Sydney	Read-Only (Monitor mode) ⓘ	0	0 Full protection 1 Read-Only	<button>Edit</button>
São Paulo	Read-Only (Monitor mode) ⓘ	0	0 Full protection 2 Read-Only	<button>Edit</button>

Let's assume this would be an unused foreign region where you would not expect activity happening. You can turn on Region Lock to ensure no changes are made to this region unless its from within Dome9.

## Dome9 Lab Guide

Edit Region - AWS - São Paulo

Newly Detected Groups Behavior



**Read-Only (Monitor mode)**

Newly detected Security Groups and their rules will be imported and will be set as 'Read-Only' Security Groups. These groups will be monitored by Dome9 for changes but can still be changed outside Dome9.



**Full protection (Dome9 managed)**

Newly detected Security Groups will be imported with their initial rules to the Dome9 service, and thereafter be treated as Full Protection (Dome9 Managed) Security Groups. Any further security policy changes should be made using the Dome9 console or API. Dome9 Policy Tamper Protection will be added to the Security Group and will be activated after the import.



**Region lock**

Newly detected Security Groups will be imported and will immediately have their rules cleared (deleted) on both ingress and egress. This mode ensures that no network changes will be made to the region, unless they are performed on the Dome9 console.

**Note:** This will delete all ingress and egress rules from any newly discovered Security Groups. This mode is recommended for highly secured setups that are not allowed to be changed outside of the Dome9 system. This mode is also highly recommended for inactive regions.

São Paulo	select entire region	select entire region
cloud productionVPC (vpc-09bee5c22c0fd16f5)	<input type="radio"/> Read-Only (Monitor mode) <small>(?)</small> <input type="radio"/> select all	<input type="radio"/> Full protection (Dome9 managed) <small>(?)</small> <input type="radio"/> select all
cloud permissive-sg (sg-07ed9e3c937021411)	<input type="radio"/>	<input type="radio"/>
cloud default (sg-032df64f273446774)	<input type="radio"/>	<input type="radio"/>
cloud Demo-SandBox-bastionSG-2ZQ77PNH4SBA (sg-0170baf9218b4da06)	<input type="radio"/>	<input type="radio"/>
cloud vpc-c896f0af	<input type="radio"/> Read-Only (Monitor mode) <small>(?)</small> <input type="radio"/> select all	<input type="radio"/> Full protection (Dome9 managed) <small>(?)</small> <input type="radio"/> select all
cloud default (sg-ff45a086)	<input type="radio"/>	<input type="radio"/>
cloud SG1 (sg-0357eeab77f385426)	<input type="radio"/>	<input type="radio"/>

**CLOSE** **SAVE**

Go to the AWS console and create a new SG in São Paulo

AWS Services Dashboard

Create Security Group

Security group name:  Description:  VPC:

Security group rules:

**Inbound** **Outbound**

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

**Add Rule**

**Create**

Go back to Dome9 console, you should see this new SG's inbound/outbound rules deleted.

**AWS Security Group: permissive-sg (sg-07ed9e3c937021411)**

Account: AWS  
Region: São Paulo (sa-east-1)  
VPC: productionVPC (vpc-09bee5c22c0fd16f5)  
Group Description: test  
Related Links: Visualize in Clarity, VPC Flow Logs, Alerts  
History (Last 7 Days): Jul 12, 2018 7:35 PM  
Showing 1/1

Tags: No tags  
Inbound Services: There are no Inbound rules defined.  
Outbound Services: There are no Outbound rules defined. Members connected to this security group will not be able to initiate connections.  
Group Members: No members attached  
Referencing Security Groups: No referencing

You can also confirm it in the AWS console in São Paulo

Name	Group ID	Group Name	VPC ID	Description
sg-0170ba9218b4d06	Demo-SandBox-bastionSG...	vpc-09bee5c22c0fd16f5	Bastion security group	
sg-032df64f273446774	default	vpc-09bee5c22c0fd16f5	default VPC security group	
sg-0357eeab77f385426	SG1	vpc-c896f0af	launch-wizard-1 created 2018-06-13T15:50:10.469-07:00	
sg-07ed9e3c937021411	permissive-sg	vpc-09bee5c22c0fd16f5	test	
sg-f45a086	default	vpc-c896f0af	default VPC security group	

Description    Inbound    Outbound    Tags

Edit

Type    Protocol    Port Range    Source    Description

This security group has no rules

**Exercise Complete!** You have now enabled enforcement of security policies and eliminate configuration drift at a region level in AWS.

# S3 Security Lab

The goal of this exercise is to help you understand how to control S3 bucket access. We will focus on how to ensure specific role can interact with S3 and thereby not allowing any anonymous/outside users have access to sensitive data in the bucket. We will also look at least privilege concept, where you as an admin decide who has access (whitelist) to the most sensitive S3 operations and deny everyone else.

## Exercise 6.1: Detecting Exposed Buckets

### 1. Navigate to Compliance Engine Tab

The screenshot shows the Dome9 Compliance Engine interface. On the left, there's a sidebar with a timeline of events from June 22, 2023, including several security group tamper detections and assessment events. The main dashboard has tabs for 'Compliance Policies' (selected), 'Dashboard', 'Continuous Compliance', 'Playground', and 'Assessment History'. A central box displays 'Compliance & Governance' with a message about checking compliance with bundles. Below this are sections for 'AWS Prod' (AWS NIST 800-53 Rev 4 (FED...)), 'Network', 'SELECT RELEVANT ACCOUNTS AND BUNDLES', and 'SELECT RELEVANT ACCOUNTS AND BUNDLES'. The 'Network' section shows 150 Default Security Groups with network policies, 38 Security Groups with admin ports exposed, 31 Security Groups with SSH admin port exposed, and 10 Instances not configured within a VPC. The 'IAM' section shows 58 IAM Users with console password without MFA enabled, 0 Accounts without enforced Password Policy, 10 IAM Users with Inline IAM Policies applied, and 27 IAM Users enabled while unused for 90 days or more. The top right corner shows a summary of accounts: 1 GCP Accounts (WS Redshift: 0, Azure VM: 2, Azure ELB: 10, GCP VM: 3). The bottom right corner shows 'NETWORK POLICY REPORTS'.

### 2. Explore Compliance Engine page

## Dome9 Lab Guide

The screenshot shows the Dome9 Compliance Engine interface. At the top, there's a navigation bar with links for Cloud Inventory, Compliance & Governance, Network Security, IAM Safety, and Administration. The main area is titled "AWS CIS Foundations v. 1.1.0" and displays a summary: "Automated Validation of AWS CIS V1.1.0. For additional reference: https://d0.awsstatic.com/whitepapers/compliance/AWS\_CIS\_Foundations\_Benchmark.pdf https://s3.amazonaws.com/quickstart-reference/enterprise-accelerator/cis/benchmark/latest/doc/cis-benchmark-on-the-aws-cloud.pdf". Below this, there's a search bar and a dropdown menu for "Validation Types" (Medium, High, Low) and "Compliance Section" (List<CloudTrail>, IamUser, Iam, CloudTrail, SecurityGroup, IamPolicy, Region, S3Bucket, KMS, VPC). The main content area shows two sections: "Avoid the use of the 'root' account" (High risk) and "Enforce Password Policy" (High risk). Each section has a "Description", "Remediation", and a "GSL" (Groovy Script Language) snippet.

Dome9 Compliance Engine monitors and scans your AWS/Azure/GCP infrastructure to ensure alignment with compliance standards such as PCI-DSS, NIST 800-53, GDPR, CIS etc.

### 3. Select CIS Benchmarks and run bundle assessment

## Dome9 Lab Guide

Select your AWS environment in the cloud account section. Dome9 also provides the option to BYOC (Bring your own CFT) and scan your template before pushing to production.

## 5. Analyze the report findings

Within a few seconds, the Compliance Engine runs an assessment and provides detailed findings in an easy to use dashboard. You can see the number of test cases passed vs failed, as well as test cases segmented geographically, across regions and by test severities.

## 6. Filter Analyze key test results for **S3 buckets**

1. Ensure buckets are not publicly accessible

How many buckets are publicly accessible?

**Exercise Complete!** You have now detect publicly exposed buckets!

### Exercise 6.2: S3 Access Controls (exposed ACLs)

Navigate to the S3 console in AWS console. We will start with bucket 1

Bucket name	Access	Region	Date created
awsloft-s3bucket1-rohw1yk24pf	Public	US East (N. Virginia)	Aug 21, 2018 5:38:51 PM GMT-0400
awsloft-s3bucket2-si1m6gzuusrf	Public	US East (N. Virginia)	Aug 21, 2018 5:38:50 PM GMT-0400
awsloft-s3bucket3-e3vvx6gt6uf	Not public *	US East (N. Virginia)	Aug 21, 2018 5:38:50 PM GMT-0400
awsloft-s3bucket4-1jpyh3l02a98z	Not public *	US East (N. Virginia)	Aug 21, 2018 5:38:51 PM GMT-0400
awsloft-s3bucket5-4mp0szrrf0kn	Not public *	US East (N. Virginia)	Aug 21, 2018 5:38:51 PM GMT-0400
cf-templates-ob34rlv71lzd-us-east-1	Not public *	US East (N. Virginia)	Aug 19, 2018 2:03:30 PM GMT-0400

\* Objects might still be publicly accessible due to object ACLs. [Learn more](#)

Fix <yourCFT>S3Bucket1

### Exercise 6.3: S3 Access Controls (Bucket policies)

Now let's move to <yourCFT>S3Bucket2. Write JSON policy (start from the existing configuration seen below)

The screenshot shows the AWS S3 Bucket Policy editor for the bucket 'demo-s3bucket2-1fras4mi5msbu'. The 'Bucket Policy' tab is selected, showing the following JSON policy:

```

1 {
2   "Version": "2012-10-17",
3   "Id": "MyPolicy",
4   "Statement": [
5     {
6       "Sid": "Put",
7       "Effect": "Allow",
8       "Principal": "*",
9       "Action": "s3:PutObject",
10      "Resource": "arn:aws:s3:::demo-s3bucket2-1fras4mi5msbu/*"
11    }
12  ]
13 }

```

Below the policy editor, there are links for 'Documentation' and 'Policy generator'.

Bucket Policy: Access Management - Only Allow **S3-Role** to put and get objects in this bucket.

Sample snippet for Principal parameter:

```

"Principal": {
  "AWS": "arn:aws:iam:<ACCOUNT NUMBER>:role/<INSERT S3 ROLE NAME HERE>"
},

```

**Hint:** copy/paste the Principal parameter to include S3 role name. It can be found by navigating to **Services -> IAM -> Roles** -> **<your S3 Role Name>**

Bucket Policy: Least Privilege enforcement - Ensure deletion of S3 bucket can only be done based on your specific AWS account id (this is important for sensitive buckets that may have other accounts accessing it, and you want to make sure the most critical operations are whitelisted)

Sample JSON to add in your bucket policy (don't forget to add , after previous statement)

```
,
```

```
{
```

```
    "Sid": "Stmt1526361042800",
```

```
    "Effect": "Deny",
```

```
    "Principal": "*",
```

```
    "Action": "s3>DeleteBucket",
```

```
    "Resource": "arn:aws:s3:::<copy your S3 bucket resource name in existing policy>",
```

```
    "Condition": {
```

```
        "StringNotLike": {
```

```
            "aws:userid": "YOURAWSACCOUNTID"
```

```
        }
```

```
    }
```

```
}
```

**Exercise Complete!** You have now ensured no buckets are publicly exposed!

## Offboarding

Please navigate to the Cloudformation and click the stack and delete stack. You are now done!