



# AWS Immersion Day LAB 2

DISTRIBUTED FIREWALL FOR SECURE EGRESS



Aviatrix Systems  
Systems Engineering



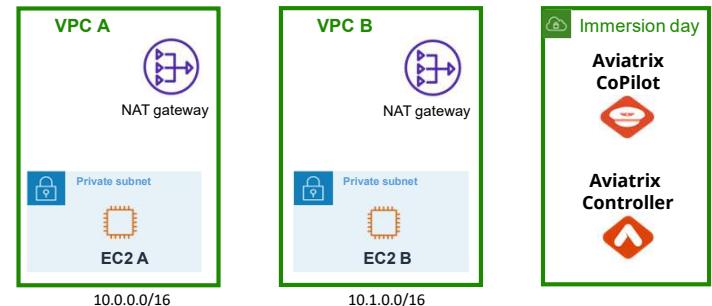
## Lab 1 Recap

In Lab 1 you created (3) AWS VPCs in the us-east-1 region.

VPCs A and B have EC2 instances on a private subnet and AWS NAT Gateway providing internet egress.

There is a 4<sup>th</sup> VPC called the “Immersion day” that contains your Aviatrix Controller and Aviatrix CoPilot as EC2 instances

In this lab you will deploy Aviatrix gateways to provide Secure Egress using the Aviatrix Distributed Firewall



**AWS us-east-1**

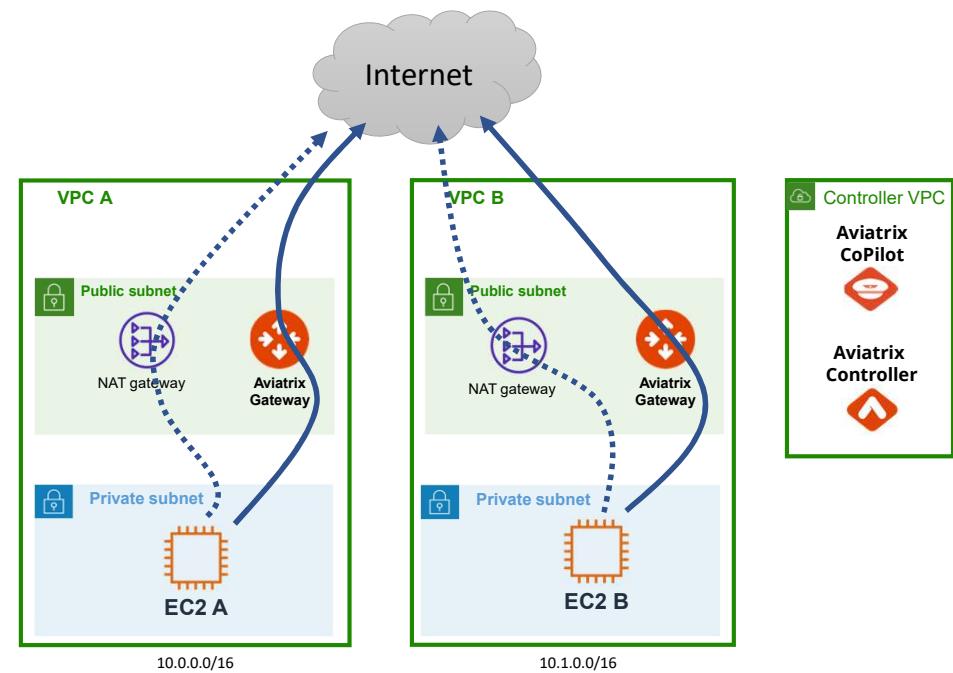


## Lab 2 Intro

In Lab 2 you are going to deploy Aviatrix Spoke Gateways in VPCs A & B.

You will create domain name filtering rules in the Aviatrix Distributed Firewall to secure your egress internet traffic.

After that you will seamlessly switch your egress traffic flows from AWS NAT Gateway to Aviatrix Gateways.



AWS us-east-1

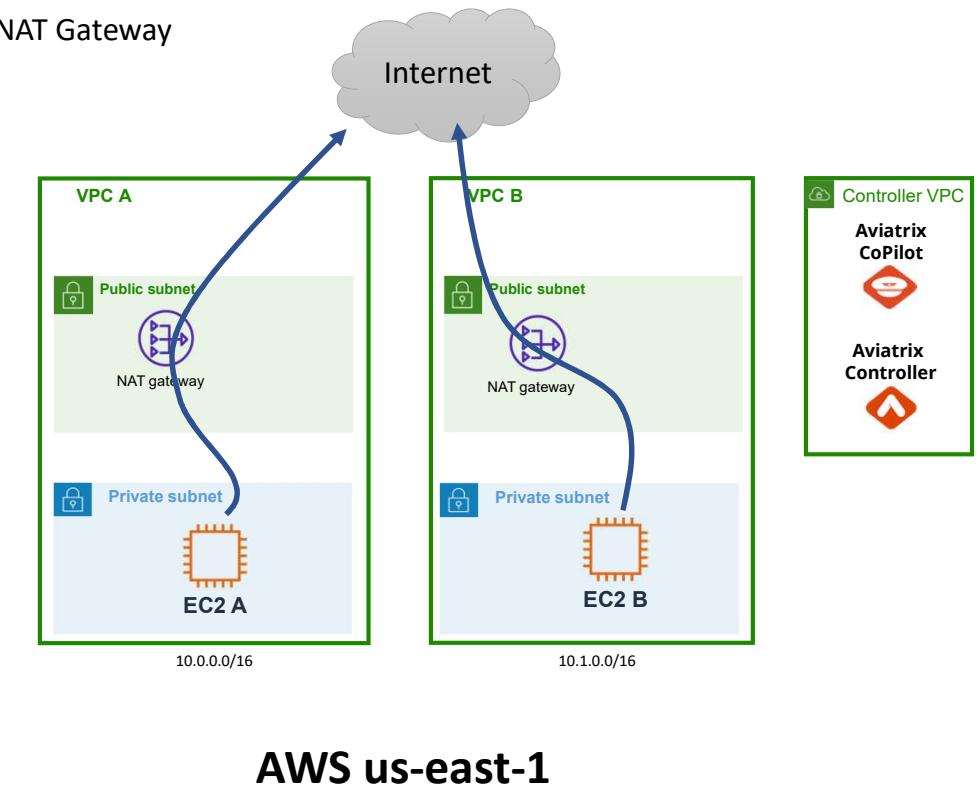
## Lab 2: Step 2

Test egress with AWS NAT Gateway

Before we do anything with Aviatrix, let's test the internet egress with AWS NAT Gateway.

In the following steps you will verify the default route is using AWS NAT Gateway.

After that you will connect to the console of instance EC2 A and test internet egress.





## Lab 2: Step 2.1

Observe the default route of your VPC A Private Route Table

Make sure your AWS Console is in the us-east-1 N. Virginia region.

From the AWS Console go to the VPC section of the console and select

**Route tables** 1

Select the **VPC A Private Route Table** 2

Select the **Routes** tab 3

Observe the 0.0.0.0/0 default route directing traffic to AWS NAT Gateway 4

The screenshot shows the AWS VPC Route Tables page. On the left, there's a sidebar with 'Virtual private cloud' and 'Your VPCs' sections, and a 'Route tables' tab which is highlighted with a yellow box and labeled '1'. Below the sidebar is a list of various VPC components: Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, and Peering connections. To the right of the sidebar, the main area shows a table titled 'Route tables (1/12)'. It lists three route tables: 'VPC C Route Table', 'VPC A Private Route Table' (which is selected, indicated by a checked checkbox and highlighted with a red box, labeled '2'), and 'VPC A Route Table'. The 'VPC A Private Route Table' row has a detailed view expanded below it. This view includes tabs for 'Details', 'Routes' (which is highlighted with a yellow box and labeled '3'), 'Subnet associations', 'Edge associations', and 'Route'. Under the 'Routes' tab, a table titled 'Routes (2)' shows two entries: one for '0.0.0.0/0' with a target of 'nat-061c67d74aa528ced' (highlighted with a red box and labeled '4') and another for '10.0.0.0/16' with a target of 'local'.



## Lab 2: Step 2.2

Connect to EC2 A instance console

Go the EC2 section of the AWS Console.

Select Instances

Find the EC2 A instance and select it. **1**

Click the Connect button **2**

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with 'EC2' selected. Under 'Instances', 'Instances' is also selected. The main area shows a table of instances:

Name	Instance ID	Instance State	Type
AviatrixController	i-087a07dea711bf121	Running	c5.xlarge
EC2 B	i-059e3a31c13a391dd	Running	t3.micro
<b>EC2 A</b>	<b>i-0b4f42221c25af84f</b>	<b>Running</b>	<b>t3.micro</b>
AviatrixCopilot	i-030b6278056a58402	Running	c5.xlarge

A yellow box labeled '1' highlights the 'EC2 A' row. Another yellow box labeled '2' highlights the 'Connect' button at the top right of the table header. The 'EC2' logo is also highlighted with a yellow box.



## Lab 2: Step 2.3

Connect to EC2 A instance console

EC2 > Instances > i-09ac71db6d325a3b8 > Connect to instance

### Connect to instance Info

Connect to your instance i-09ac71db6d325a3b8 (EC2 A) using any of these options

EC2 Instance Connect Session Manager 1 SSH client EC2 serial console

Session Manager usage:

- Connect to your instance without SSH keys or a bastion host.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager Preferences page.

Cancel Connect 2

Select the Session Manager tab 1

Click the Connect button 2



## Lab 2: Step 2.4

Connect to EC2 A instance console

Your browser should open a new tab giving you a CLI session.

Type the command:

**sudo su -l ec2-user** 1

*(dash lower case L)*

You are now logged on as the ec2-user and should see your private IP address in the hostname.

Next, let's test internet egress.

A screenshot of a web-based terminal window from AWS Systems Manager. The URL is 'us-east-1.console.aws.amazon.com/systems-manager/session-manager/i-09ac71db6d325a3b8?region=us-east-1'. The session ID is 'brad-000841717aa667ae4' and the instance ID is 'i-09ac71db6d325a3b8'. The terminal window shows a shell session with the prompt 'sh-4.2\$'. The user types the command 'sudo su -l ec2-user' (with the 'L' highlighted in a yellow box labeled '1'). The command is executed, and the user is prompted for a password. After logging in, the user sees their private IP address in the hostname: '[ec2-user@ip-10-0-2-10 ~]\$'.

```
sh-4.2$  
sh-4.2$  
sh-4.2$ sh-4.2$ sudo su -l ec2-user  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$
```



## Lab 2: Step 2.5

Test connection to google.com

Let's test a connection to google.com

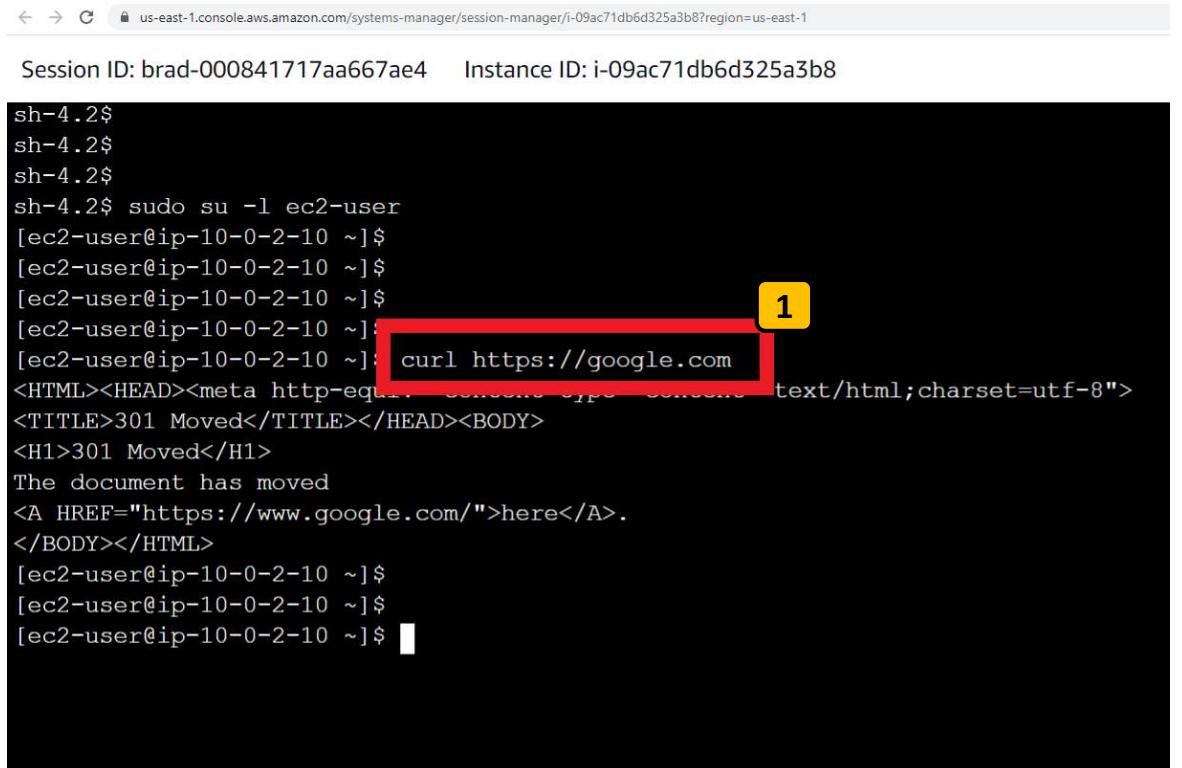
Type the command:

`curl https://google.com` 1

You should see your CLI return HTML code from google.com

Great! Internet is working..

Next, let's do a software update for this instance from AWS...



The screenshot shows a terminal window titled "Session ID: brad-000841717aa667ae4" and "Instance ID: i-09ac71db6d325a3b8". The URL "curl https://google.com" is highlighted with a red box and labeled with a yellow box containing the number "1". The terminal output shows the command being run and the resulting HTML response from Google.

```
sh-4.2$  
sh-4.2$  
sh-4.2$  
sh-4.2$ sudo su -l ec2-user  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]: curl https://google.com1  
<HTML><HEAD><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  
<TITLE>301 Moved</TITLE></HEAD><BODY>  
<H1>301 Moved</H1>  
The document has moved  
<A href="https://www.google.com/">here</A>.  
</BODY></HTML>  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$
```



## Lab 2: Step 2.6

Run a software update

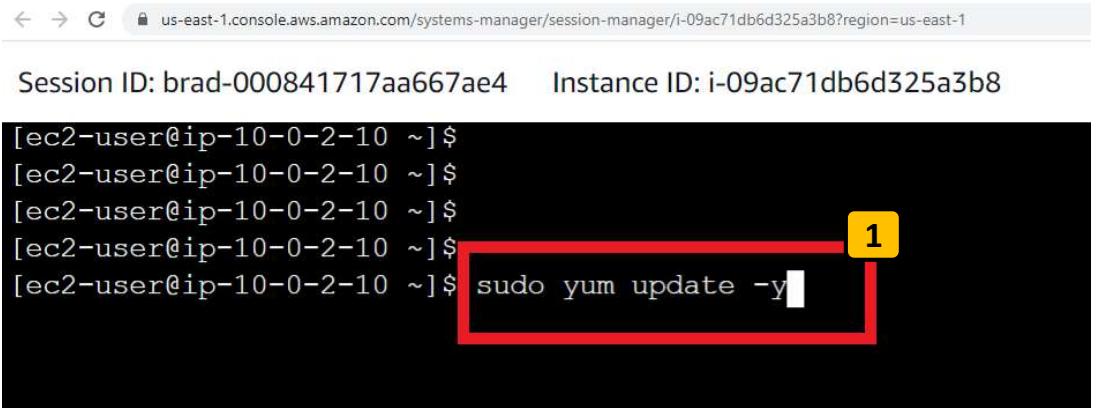
Let's run a software update from the AWS repositories

Run the command:  
**sudo yum update -y** 1

This will connect to ***amazonaws.com*** domains to download software updates.

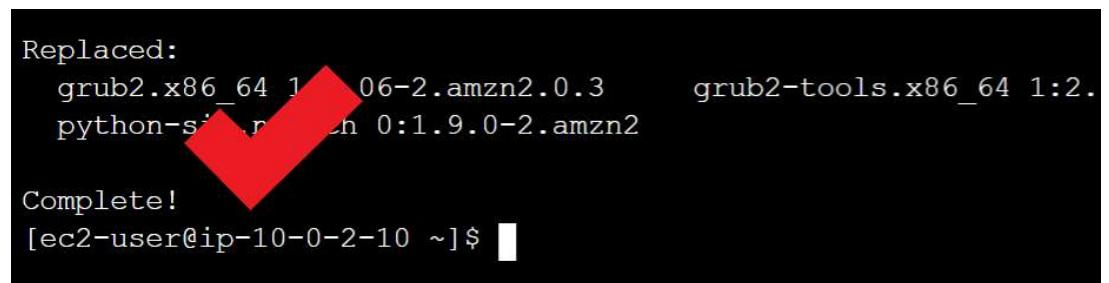
Great! Software updates are working..

Next, let's see if this instance can connect to unwanted or potentially harmful domains...



Session ID: brad-000841717aa667ae4    Instance ID: i-09ac71db6d325a3b8

```
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$  
[ec2-user@ip-10-0-2-10 ~]$ sudo yum update -y1
```



```
Replaced:  
grub2.x86_64 1:06.2.amzn2.0.3      grub2-tools.x86_64 1:2.  
python-sip._src._rhel 0:1.9.0-2.amzn2  
  
Complete!  
[ec2-user@ip-10-0-2-10 ~]$
```



## Lab 2: Step 2.7

Test connections to potentially harmful domains

Test that your EC2 instance has unfiltered internet access, to even potentially harmful domains.

Run the commands:

`curl https://ransomware.org` 1

`curl https://malware.net` 2

`curl https://botnet.com` 3

For each command you should see the CLI return HTML code from those domains ... a successful connection 4

The screenshot shows three separate terminal sessions on an AWS EC2 instance. Each session has a red box highlighting the command being run, and a yellow box numbered 1, 2, or 3 indicating the sequence of the commands. The sessions are as follows:

- Session 1 (Top):** The command `curl https://ransomware.org` is highlighted. The output shows the terminal prompt [`[ec2-user@ip-10-0-2-10 ~]$`] followed by the command itself.
- Session 2 (Middle):** The command `curl https://malware.net` is highlighted. The output shows the terminal prompt [`[ec2-user@ip-10-0-2-10 ~]$`] followed by the command itself.
- Session 3 (Bottom):** The command `curl https://botnet.com` is highlighted. The output shows the terminal prompt [`[ec2-user@ip-10-0-2-10 ~]$`] followed by the command itself. A large red arrow points from the bottom of Session 3 towards the bottom of Session 1, indicating a visual connection between the two bottom sessions.

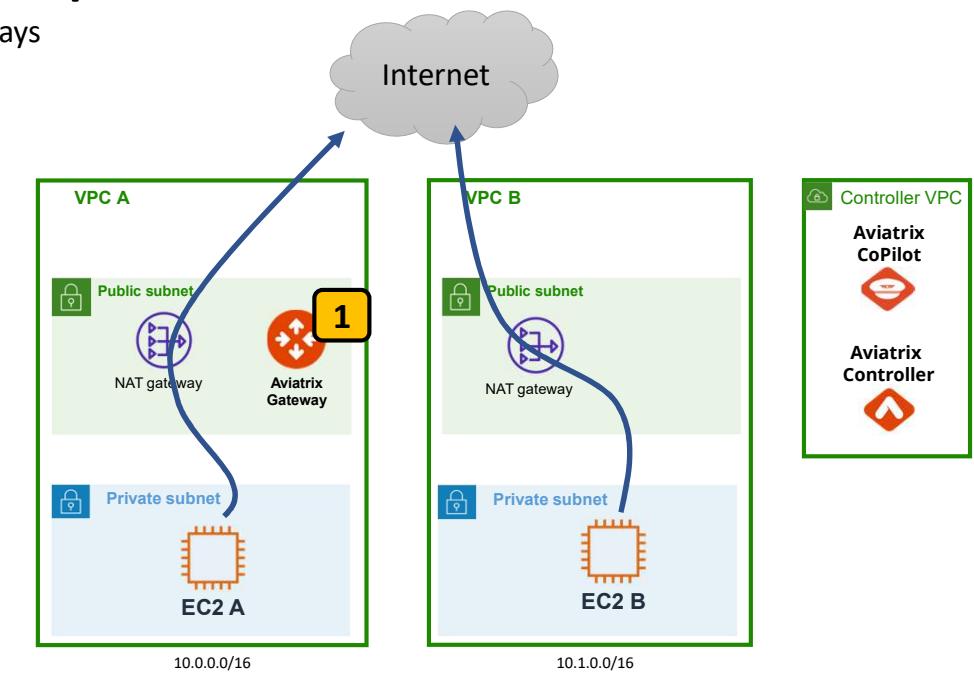
Each session also displays the session ID and instance ID at the top.

## Lab 2: Checkpoint 1

Deploy Aviatrix Gateways

At this point you've verified unfiltered egress through the AWS NAT Gateway

**In the next steps you will log on to the Aviatrix UI and deploy an Aviatrix Gateway in the public subnet of VPC A**



AWS us-east-1



## Lab 2: Step 2.8

Log in to your Aviatrix CoPilot

Go the us-east-1 N. Virginia region in your AWS Console 1

Select the Aviatrix-Copilot EC2 instance 2

On the Details tab, find the Public IP address and click on “open address” 3

This will open a browser tab to log on to the Aviatrix Copilot UI

The screenshot shows the AWS EC2 Instances page. At the top right, the region is set to "United States (N. Virginia)" with a red box around it and a yellow box labeled "1". The main table lists four instances:

Name	Instance ID	Instance state	Instance type	Status
AviatrixController	i-087a07dea711bf121	Running	c5.xlarge	3/3 ch
EC2 B	i-059e3a31c13a391dd	Running	t3.micro	3/3 ch
EC2 A	i-0b4f42221c25af84f	Running	t3.micro	3/3 ch
<b>AviatrixCopilot</b>	<b>i-030b6278056a58402</b>	<b>Running</b>	<b>c5.xlarge</b>	<b>3/3 ch</b>

A yellow box labeled "2" highlights the "AviatrixCopilot" row. The instance details page for "i-030b6278056a58402 (AviatrixCopilot)" is shown below. The "Details" tab is selected. A red box highlights the "Public IPv4 address" field, which contains "54.146.248.233 | open address" with a yellow box labeled "3" next to it.

**Instances (1/4) Info**

Last updated 3 minutes ago

Find Instance by attribute or tag (case-sensitive)

All states

Actions

EC2 Instances

EC2 > Instances

EC2

Dashboard

EC2 Global View

Events

Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

Network & Security

i-030b6278056a58402 (AviatrixCopilot)

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary

Instance ID: i-030b6278056a58402

Public IPv4 address: 54.146.248.233 | [open address](#)

IPv6 address: -

Instance state: Running

Private IPv4 addresses: 10.8.0.154

Public IPv4 DNS: -



## Lab 2: Step 2.9

Log in to your Aviatrix CoPilot

Acknowledge the Certificate warning  
by clicking Advanced, then Proceed  
(Chrome browser) **1**

You should see the Aviatrix CoPilot  
logon page **2**

Username = lab\_student

Password = ImmersionDay123# **3**

The screenshot shows a Chrome browser window with two main parts. On the left, the Aviatrix CoPilot logon page is displayed. It has fields for 'Username' (lab\_student) and 'Password' (ImmersionDay123#), a 'Log In' button, and checkboxes for 'Remember Me' and 'Forgot Password'. A yellow box labeled '2' points to the Aviatrix logo at the top of the page. A yellow box labeled '3' points to the password field. On the right, a certificate warning dialog is shown over the browser's address bar. The dialog title is 'Your connection is not private'. It states: 'Attackers might be trying to steal your information from 34.239.54.147 (for example, passwords, messages, or credit cards). Learn more'. Below this is the error code 'NET::ERR\_CERT\_AUTHORITY\_INVALID'. At the bottom of the dialog, there are buttons for 'Advanced' (highlighted with a red box and labeled '1'), 'Back to safety', and 'Proceed to 34.239.54.147 (unsafe)' (also highlighted with a red box).

NOTE: You may see messages regarding patch updates and alerts. Please acknowledge these alerts and continue.



## Lab 2: Step 2.10

Log in to your Aviatrix CoPilot

You should now see  
your Aviatrix CoPilot UI.

Take a minute to  
checkout the main  
Dashboard page and all  
the info it provides.

Click on any widget to  
get more info 1



The screenshot shows the Aviatrix CoPilot dashboard interface. On the left is a dark sidebar menu with options: CoPilot, Dashboard, Cloud Fabric, Networking, Security, Groups, Cloud Resources, Monitor, Diagnostics, Administration, and Settings. The main area features a world map with several red dots representing gateway locations. Below the map are three main dashboard cards:

- Secure Egress Traffic:** A card with a yellow header containing a gear icon and the text "Secure Egress Traffic". It includes a brief description: "Improve Security and reduce Cloud Cost with an Egress solution that embeds Application visibility and Threat detection." and "Distributed Cloud Firewall for Egress supports automated, rapid deployment, be it at global scale, a single region, or just one application at a time." A blue "Start" button is at the bottom.
- Gateways Health:** A card with a yellow header containing a gear icon and the text "Gateways Health". It displays three categories: Transit, Spoke, and Specialty, each with a green "All Up" status indicator. A blue "1" is highlighted in a yellow box over the "All Up" button for the Transit category.
- Connection Health:** A card with a yellow header containing a gear icon and the text "Connection Health". It includes a "Gateway Connections" section with a green "All Up" status indicator and a "View in Topology" link.

At the top right of the dashboard are several small icons: a bell, a magnifying glass, a user profile, and a refresh symbol.



## Lab 2: Step 2.11

Deploy Aviatrix Spoke Gateway in VPC A Public Subnet

From the left-hand navigation in Aviatrix CoPilot...

Under Cloud Fabric select **Gateways** 1

Select the **Spoke Gateways** tab 2

Click the **+Spoke Gateway** 3 button to deploy a new Spoke Gateway.

The screenshot shows the Aviatrix CoPilot interface. On the left, there is a dark sidebar with the Aviatrix logo at the bottom. The sidebar has a 'Cloud Fabric' section with 'Topology' and 'Gateways' (which is highlighted with a red box and a yellow '1')) options. Below that are 'Edge', 'Scaling', 'Networking', and 'Network Segmentation'. On the right, the main area has a header with tabs: 'Gateways', 'Overview', 'Transit Gateways', 'Spoke Gateways' (which is highlighted with a red box and a yellow '2')) and 'Spec'. Below the header is a search bar and some filter icons. A large red box highlights the '+ Spoke Gateway' button, which is also labeled with a yellow '3'. The main table lists three existing Spoke Gateways: 'aws-us-west-2-spoke-1', 'aws-us-west-2-spoke-2', and 'aws-us-west-2-spoke-3', each with details like Region, VPC/VNet, and Subnet CIDR. At the bottom of the table, it says 'Total 3 Spoke Gateways'.

Name	Region	VPC/VNet	Subnet CIDR
aws-us-west-2-spoke-1	us-west-2	vpc-0dff... vpc-04fd9...	10.51.0.32/28 10.52.0.32/28
aws-us-west-2-spoke-2	us-west-2	vpc-0270...	10.53.0.32/28
aws-us-west-2-spoke-3	us-west-2	vpc-0270...	10.53.0.32/28



## Lab 2: Step 2.12

Deploy Aviatrix Spoke Gateway in VPC A Public Subnet

Create Spoke Gateway

1 Name: aws-us-east-1-SpokeA

2 Cloud: AWS Standard

3 Account: aws-account

4 Region: us-east-1 (N. Virginia)

5 VPC/VNet: VPC A

6 Instance Size: t3.medium

High Performance Encryption: Off

Attach To Transit Gateway: Optional

Name the gateway **aws-us-east-1-SpokeA** 1

Select the cloud AWS standard 2

Select the account **aws-account** 3

Select the region **us-east-1** 4

Select VPC A 5

Select the **t3.medium** instance size 6



## Lab 2: Step 2.13

Deploy Aviatrix Spoke Gateway in VPC A Public Subnet

Instances

+ Instance

Attach to Subnet

1 10.0.0.0/24~us-east-1a~~VPC A - AZ1

Public IP

Allocate New Static Public IP

Cancel Save

Select the 10.0.0.0/24 subnet in **AZ1** **7**

Upon clicking Save, Aviatrix CoPilot will begin the gateway deployment

Leave the Public IP at the default selection of Allocate New

Click **Save** **8**



## Lab 2: Step 2.14

Monitor the gateway deployment Task

The screenshot shows the Aviatrix CoPilot interface. On the left, there's a sidebar with 'CoPilot' at the top, followed by 'Monitor' (with 'Notifications' and 'Tasks' under it), 'Settings', 'Resources', and 'Task'. A red box labeled '1' highlights the search bar in the 'Monitor' section where 'task' is typed. A yellow box labeled '2' highlights the 'Notifications / Tasks' link in the sidebar. The main area has tabs for 'Notifications', 'Alerts', 'Alerts Configuration', 'System Messages', 'Tasks' (which is selected and highlighted in black), and 'Recipients'. Below the tabs, there are two buttons: 'Tasks' and 'Active Gateway Operations'. A red box labeled '3' highlights the first task in the list, which is 'Create spoke gateway: aws-us-east-1-SpokeA'. This task is listed under the 'Tasks' tab. The task details show it's completed with a green progress bar.

Name	Entity	Status	Progress
Create spoke gateway: aws-us-east-1-SpokeA		Completed	[Green Progress Bar]
Create primary gateway		Completed	[Green Progress Bar]

Total 1 task

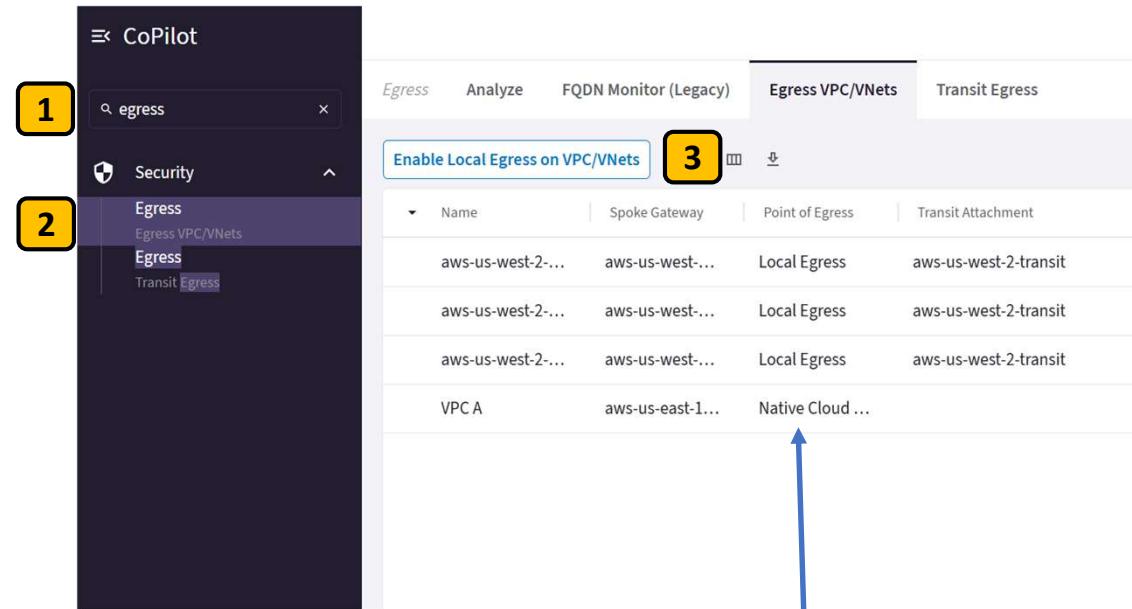
From the CoPilot search bar type **task** 1

Click the search result **Notifications / Tasks** 2

Observe the spoke gateway creation Task and wait for it to complete 3

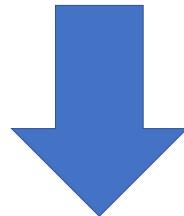
## Lab 2: Step 2.15

Enable Egress on the Aviatrix Spoke Gateway



Name	Spoke Gateway	Point of Egress	Transit Attachment
aws-us-west-2...	aws-us-west-...	Local Egress	aws-us-west-2-transit
aws-us-west-2...	aws-us-west-...	Local Egress	aws-us-west-2-transit
aws-us-west-2...	aws-us-west-...	Local Egress	aws-us-west-2-transit
VPC A	aws-us-east-1...	Native Cloud ...	

Now let's configure your new Aviatrix Spoke Gateway to do Local Egress for VPC A, so we can use it for egress NAT from our private subnets.



From the CoPilot search bar type **egress** **1**

Click the search result **Egress VPC/VNets** **2**

Click the **Enable Local Egress on VPC/VNets** button **3**

Notice how CoPilot is telling us that VPC A is currently using "Native Cloud Egress"

Because this VPC is currently using AWS NAT Gateway

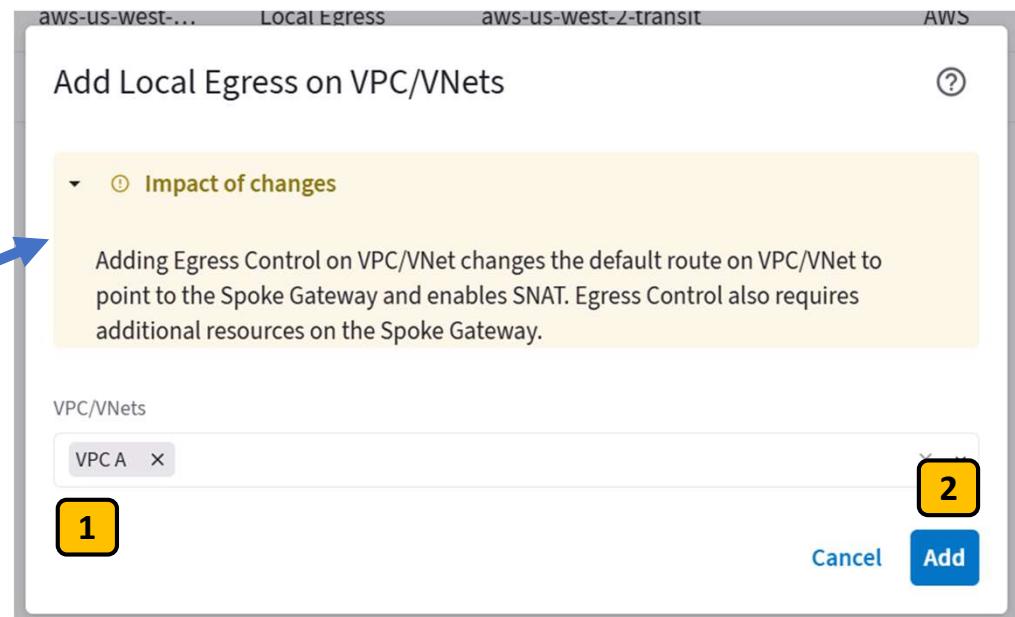
## Lab 2: Step 2.16

Enable Egress on the Aviatrix Spoke Gateway

In the Add Local Egress on VPC/VNets pop-up, select your new **aws-us-east-1-SpokeA** gateway from the VPC/VNets drop down **1**

Click **Add** **2**

After you click Add, Copilot will change the VPC default route associated to all private subnets in VPC A to point to your new Aviatrix Gateway





## Lab 2: Step 2.17

Observe the default route of your VPC A Private Route Table

After enabling Egress on the Aviatrix Gateway, the Aviatrix Controller changed the default route in your VPC private route table.

From the AWS Console go to the VPC section of the console and select **Route tables** 1

Select the **VPC A Private Route Table** 2

Select the **Routes** tab

Observe the 0.0.0.0/0 default route directing traffic to Aviatrix Gateway 3

The screenshot shows the AWS VPC dashboard with the 'Route tables' section selected. A yellow box labeled '1' highlights the 'Route tables' link in the left sidebar. Another yellow box labeled '2' highlights the 'VPC A Private Route Table' in the list of route tables. A third yellow box labeled '3' highlights the 'Routes' tab in the detailed view of the selected route table. The detailed view shows two routes: one for 0.0.0.0/0 pointing to the Aviatrix gateway (eni-0f8168d817292c1fa) and another for 10.0.0.0/16 pointing to the local subnet.

Name	Route table ID	Last updated
VPC C Route Table	rtb-0b5a73fe6c63e1094	1 minute ago
immersion-day-rtb	rtb-0fb243760d77f8b2d	
<b>VPC A Private Route Table</b>	<b>rtb-0fb243760d77f8b2d</b>	
-	rtb-094cf51a4c8078233	
-	rtb-0d38dae85884dabf7	

Destination	Target
0.0.0.0/0	eni-0f8168d817292c1fa
10.0.0.0/16	local

## Lab 2: Checkpoint 2

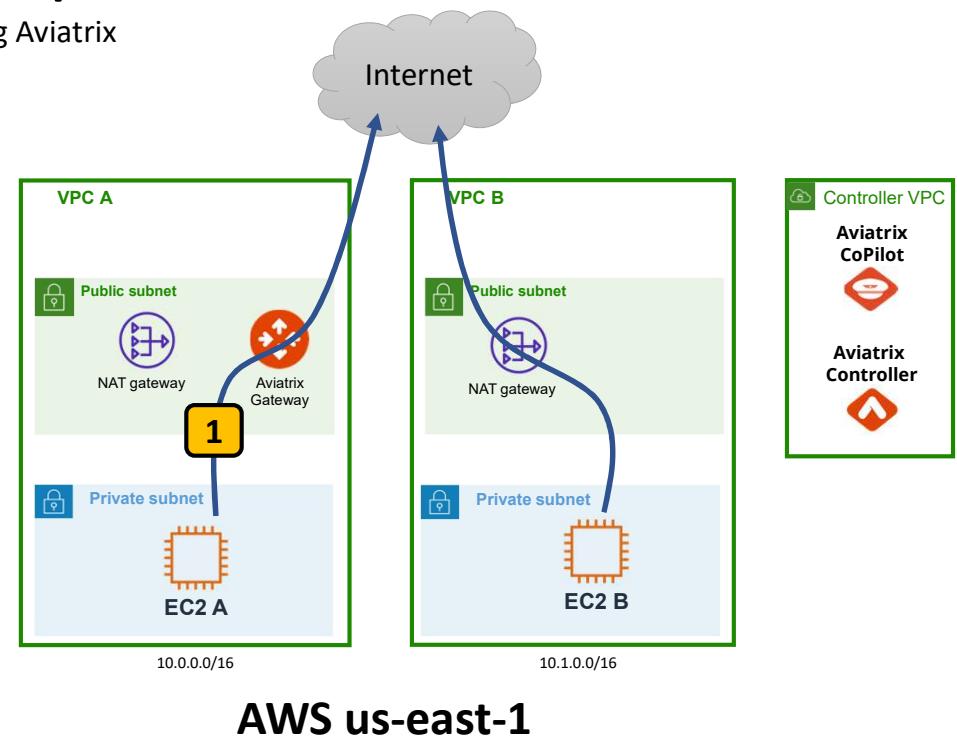
Simple egress NAT using Aviatrix

At this point you've deployed an Aviatrix Gateway in VPC A

You configured the Aviatrix Gateway for Egress

You observed the VPC A Private Route Table for private subnets has been changed to use the Aviatrix Gateway for egress. **1**

**Next, let's configure security rules for egress traffic using Aviatrix Distributed Firewall**



**Note:** Aviatrix does not charge data processing charges for NAT. The Aviatrix cost for simple egress NAT is \$0.14 /hr per gateway (plus the EC2 gateway instance charges)

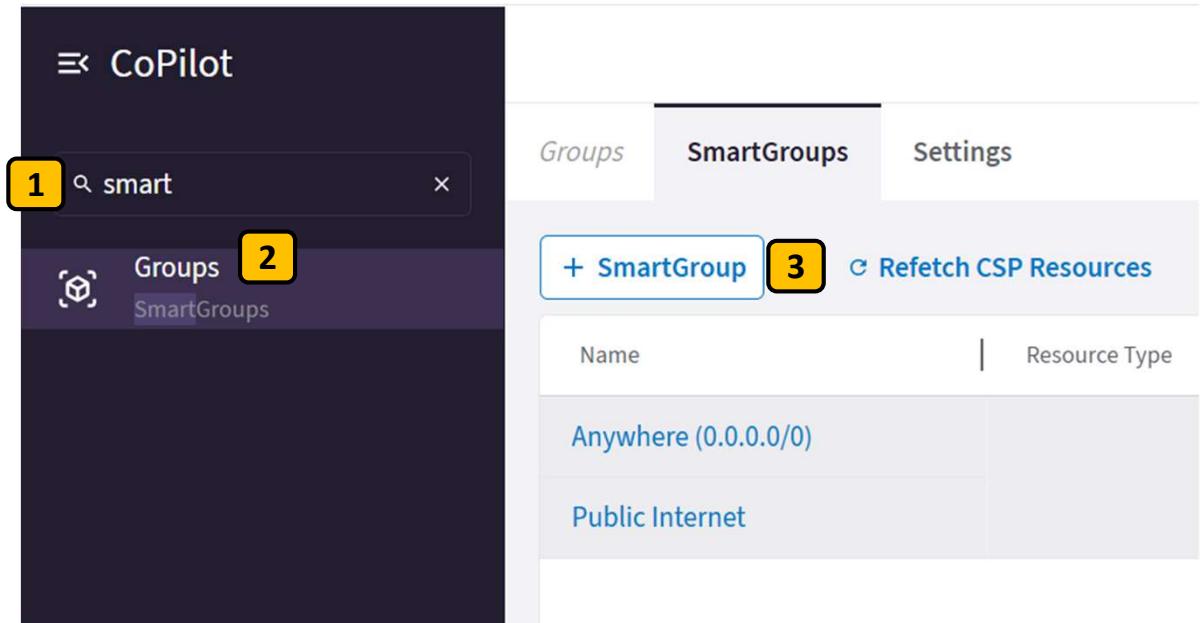
## Lab 2: Step 2.18

Create a SmartGroup to use for firewall rules

From the CoPilot search bar type **smart** 1

Click the search result **SmartGroup** 2

Click the **+SmartGroup** button 3





## Lab 2: Step 2.19

Create a SmartGroup to use for firewall rules

Name the SmartGroup **DEV** 1

We want all EC2 instances tagged **Environment = Development** to be in this SmartGroup. 2

The instances EC2 A and EC2 B have already been tagged this way.

Select the CSP Tag key **Environment**

Select the value **Development** 3

Click on the **Preview** toggle switch 4

Create SmartGroup ?

Name 1  
DEV

Resource Selection Preview (2)  4

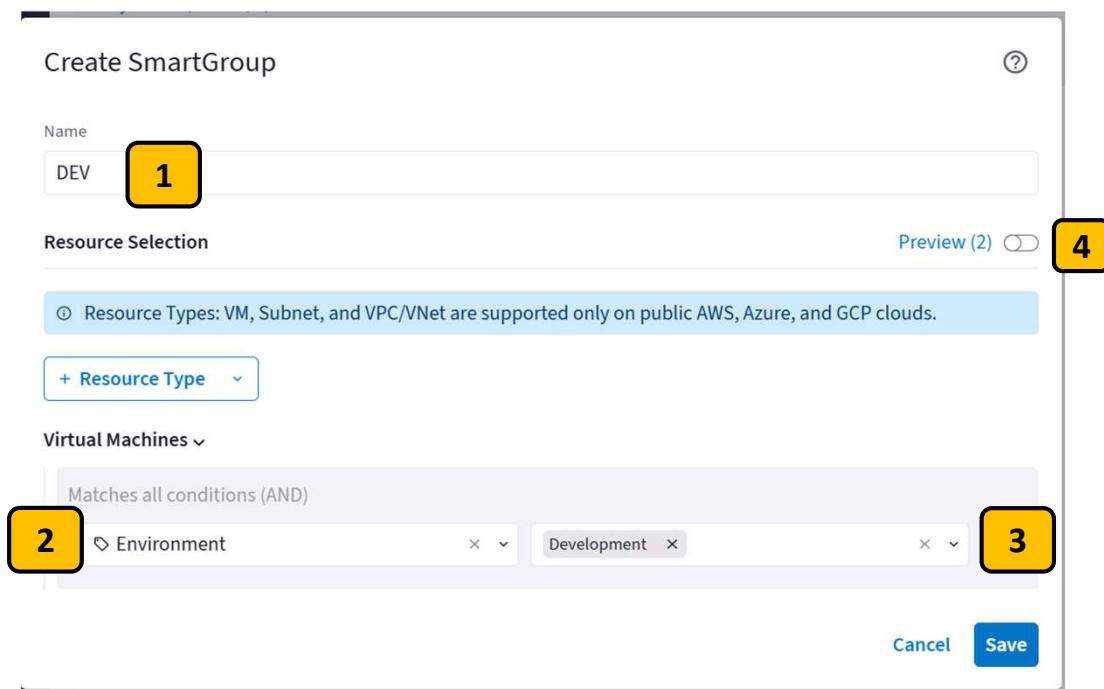
Resource Types: VM, Subnet, and VPC/VNet are supported only on public AWS, Azure, and GCP clouds.

+ Resource Type 2

Virtual Machines Matches all conditions (AND)

Environment 3 Development 3

Cancel Save





## Lab 2: Step 2.20

Create a SmartGroup to use for firewall rules

Aviatrix CoPilot already knows that your instances EC2 A and EC2 B were tagged with the EC2 tags

**Environment = Development**

Confirm that both EC2 A and EC2 B instances match the SmartGroup **1**

Click **Save** to create the SmartGroup **2**

We'll use this SmartGroup to create firewall rules

Name	Type	Cloud	Region
EC2 A	VM	AWS	us-east-1
EC2 B	VM	AWS	us-east-1



## Lab 2: Step 2.21

Create a SmartGroup to use for firewall rules

Create another SmartGroup and name it **PROD** 1

We want all EC2 instances tagged **Environment = Production** to be in this SmartGroup.

Select the CSP Tag key **Environment** 2

Select the value **Production** 3

Click **Resource Selection** to confirm 4

The screenshot shows the 'Create SmartGroup' dialog box. Step 1: The 'Name' field contains 'PROD'. Step 2: The 'Resource Selection' dropdown is open, showing 'Environment' selected. Step 3: The 'Environment' dropdown shows 'Production' selected. Step 4: The 'Preview (3)' button is highlighted.



## Lab 2: Step 2.22

Create a SmartGroup to use for firewall rules

Aviatrix CoPilot already knows your instances tagged with the EC2 tags  
**Environment = Production**

Confirm that both there are three instances named SAP in the us-west-2 region in the SmartGroup **1**

Click **Save** to create the SmartGroup **2**

Create SmartGroup ?

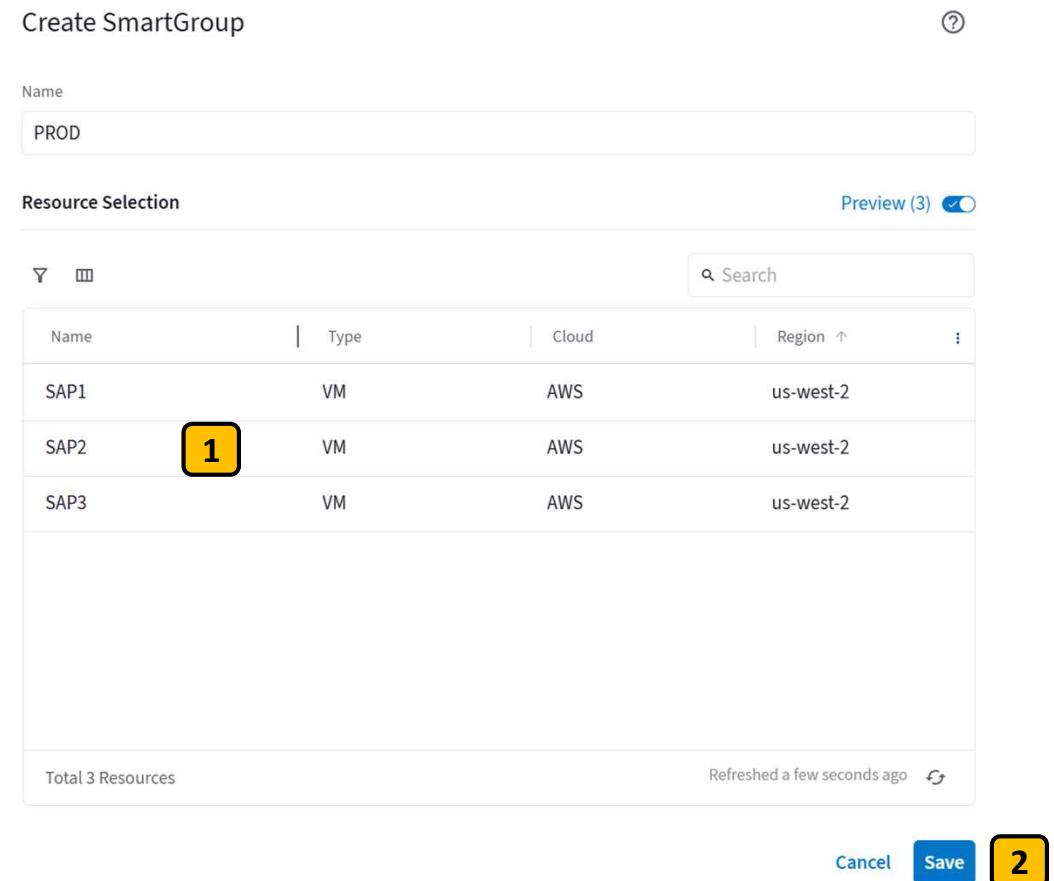
Name

Resource Selection Preview (3)

Name	Type	Cloud	Region
SAP1	VM	AWS	us-west-2
SAP2	VM	AWS	us-west-2
SAP3	VM	AWS	us-west-2

Total 3 Resources Refreshed a few seconds ago

Save **2**





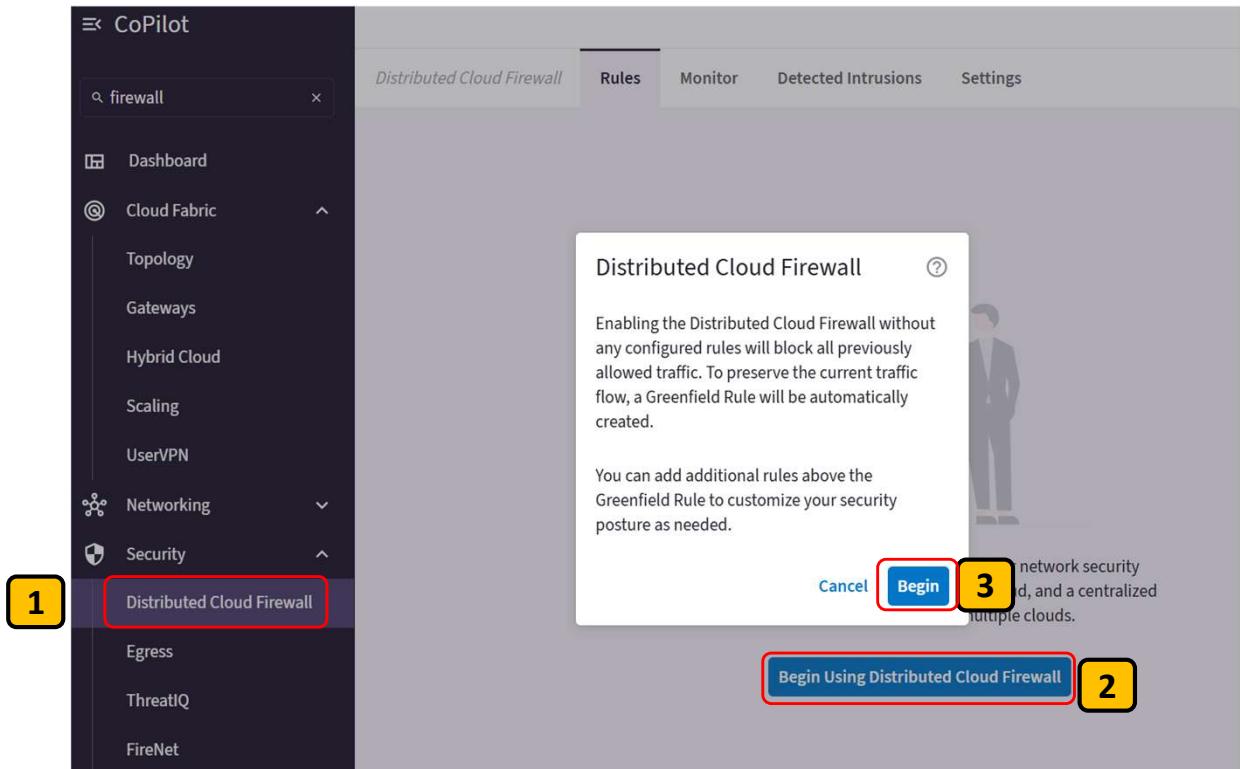
## Lab 2: Step 2.23

Enable Aviatrix Distributed Firewalling

From the CoPilot panel  
navigate to **Security > Distributed Cloud Firewall**

Click **Begin Using Distributed Firewalling**

In the pop-up dialog click  
**Begin**





## Lab 2: Step 2.24

Create a WebGroup to use for firewall rules

The screenshot shows the Aviatrix CoPilot interface. On the left, there is a navigation panel with a search bar containing 'firewall'. Below the search bar are several menu items: 'Groups' (highlighted with a yellow box labeled '1'), 'Cloud Resources', 'Monitor', 'FlowIQ', and 'Performance'. In the center, there are tabs for 'Groups', 'SmartGroups', 'WebGroups' (highlighted with a yellow box labeled '2'), and 'Settings'. A yellow banner at the top states: '⚠ URL Type WebGroups is in Preview. Preview features are not safe for deployment in production environments.' with a 'Learn More' link. Below the banner is a table with columns: Name, Type, Domains/URLs, and Rule References. One row is visible: 'All-Web' (Type: Predefined WebGroup). At the bottom of the table are edit and delete icons. A search bar and a 'Default View' dropdown are also present.

On the CoPilot navigation panel scroll down and select  
**Groups>WebGroup**

Click the button **+ WebGroup** to create a new WebGroup



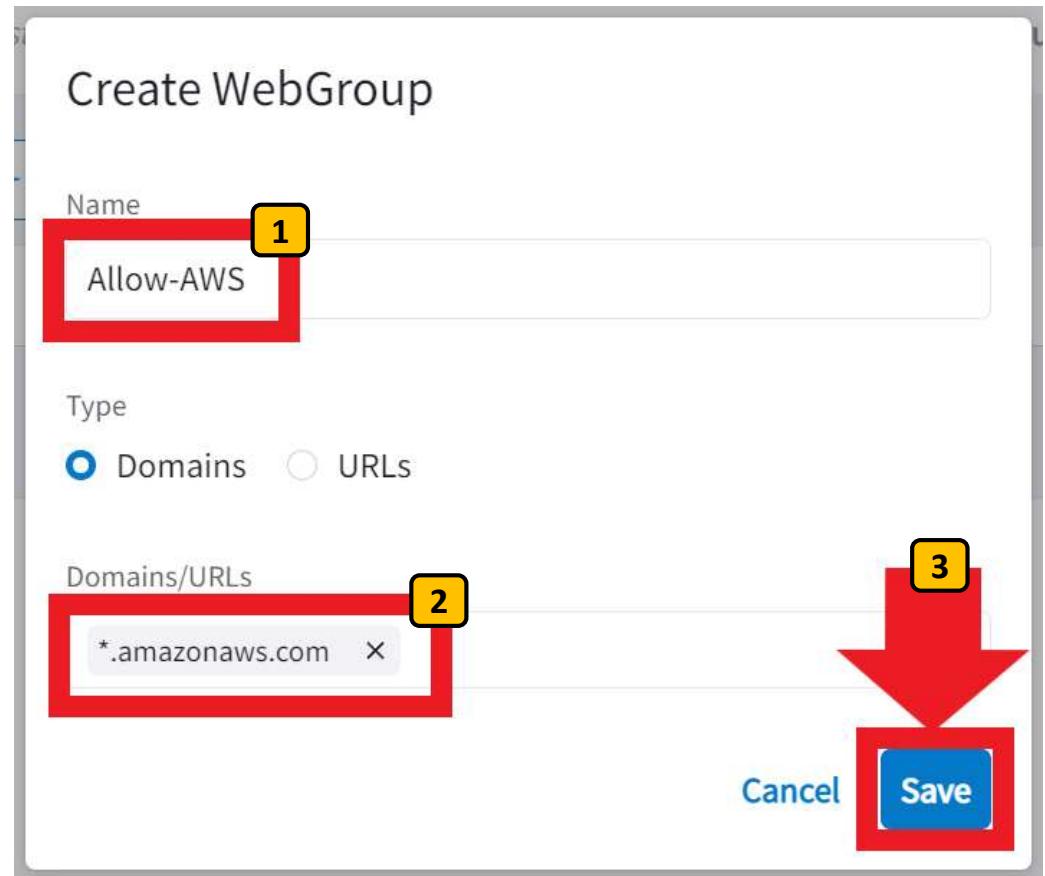
## Lab 2: Step 2.25

Create a WebGroup to use for firewall rules

Name the WebGroup **Allow-AWS** 1

Enter the domain name  
**\*.amazonaws.com** 2

Click **Save** 3





## Lab 2: Step 2.26

Create a Distributed Firewall Rule

The screenshot shows the Aviatrix CoPilot interface. On the left, there's a sidebar with various options like Scaling, UserVPN, Networking, Security (selected), Distributed Cloud Firewall (highlighted with a red box), Egress, ThreatIQ, FireNet, and Anomaly Detection. A search bar at the top has 'firewall' typed into it. The main area is titled 'Distributed Cloud Firewall' and shows the 'Rules' tab selected (highlighted with a yellow box labeled '1'). Below the tab, there's a table header with columns: Priority, Name, Source, Destination, and WebGroup. A single row is visible in the table body, labeled '214748...' with 'Greenfield-Rule' under 'Name', 'Anywhere (0.0.0.0...)' under 'Source', and 'Anywhere (0.0.0.0...)' under 'Destination'. At the top of the main area, there's a toolbar with icons for search, refresh, and save, along with a 'Save View' button. A red box highlights the '+ Rule' button in the toolbar, and a yellow box labeled '2' highlights this button.

On the Distributed Firewall click the **Rules** tab **1**

Click the button **+ Rule** to create a new distributed firewall rule **2**



## Lab 2: Step 2.27

Create a firewall rule

Name the rule **Allow-NTP** 1

Chose the Source SmartGroups **DEV** and **PROD** 2

Choose the Destination SmartGroup **Public Internet** 3

Choose the Protocol **UDP** and enter Port number **123** 4

Enable Enforce and Logging 5

Place Rule at Top and click **Save In Drafts** 6

The screenshot shows the 'Create Rule' dialog box. The rule is named 'Allow-NTP'. The 'Source Groups' field contains 'DEV' and 'PROD'. The 'Destination Groups' field contains 'Public Internet'. Under 'Protocol', 'UDP' is selected and 'Port' is set to '123'. In the 'Rule Behavior' section, both 'Enforcement' and 'Logging' are turned on. Under 'Action', 'Permit' is selected and 'SG Orchestration' is turned on. 'Place Rule' is set to 'Top'. The 'Save In Drafts' button is highlighted with a red box.



## Lab 2: Step 2.28

Create a firewall rule for AWS domains

Create another rule named **Allow-AWS** 1

Chose the Source SmartGroups **DEV** and **PROD** 2

Choose the Destination SmartGroup **Public Internet** 3

Choose the WebGroup **Allow-AWS** 4

Choose the Protocol **TCP** and enter Port number **443** 5

Enable Enforce and Logging 6

Place Rule at Top and click **Save In Drafts** 7

The screenshot shows the 'Create Rule' dialog box. Key steps are highlighted with yellow boxes and numbers:

1. Name: Allow-AWS
2. Source Groups: DEV, PROD
3. Destination Groups: Public Internet
4. WebGroups: Allow-AWS
5. Protocol: TCP, Port: 443
6. Rule Behavior: Enforcement (checked), Logging (checked)
7. Place Rule: Top, Save In Drafts button



## Lab 2: Step 2.29

Delete the default permit any rule

Screenshot of the Aviatrix Distributed Cloud Firewall Rules interface.

The top navigation bar includes tabs for *Distributed Cloud Firewall*, **Rules**, Monitor, Detected Intrusions, and Settings. The **Rules** tab is selected.

Below the tabs is a toolbar with buttons for **+ Rule**, Actions (dropdown), Filter (Y), Sort (triangles), Refresh (refresh), 2 New (green circle), 1 Deleted (red circle), Discard, Commit, Search, Modified View (dropdown), and Save View.

The main table displays firewall rules:

Priority	Name	Source	Destination	WebGroup	Protocol	Port	Actions
0	Allow-AWS	DEV, PROD	Public Internet	Allow-AWS	TCP	44	
1	Allow-NTP	DEV, PROD	Public Internet		UDP	12	
214748...	Greenfiel...	Anywhere (0.0.0.0...)	Anywhere (0.0.0.0...)		Any		

A yellow box labeled **1** highlights the trash bin icon for the last rule (Priority 214748...).

- 1 Find the **Greenfield-Rule** that permits all traffic and delete it by clicking on the Trash Bin icon. You will see the Rule turn red.



## Lab 2: Step 2.30

Commit your firewall rules

The screenshot shows the Aviatrix Distributed Cloud Firewall Rules interface. The top navigation bar includes tabs for 'Distributed Cloud Firewall', 'Rules' (which is selected), 'Monitor', 'Detected Intrusions', and 'Settings'. Below the navigation is a toolbar with buttons for '+ Rule', 'Actions', 'Discard', and 'Commit'. A yellow box with the number '1' highlights the 'Commit' button. To the right of the toolbar are buttons for 'Modified View' and 'Save View'. The main area displays a table of firewall rules:

Priority	Name	Source	Destination	WebGroup	Protocol	Port	Actions
0	Allow-AWS	DEV, PROD	Public Internet	Allow-AWS	TCP	44	
1	Allow-NTP	DEV, PROD	Public Internet		UDP	12	
214748...	Greenfiel...	Anywhere (0.0.0.0...)	Anywhere (0.0.0.0...)		Any		

Commit your distributed firewall rule configurations by clicking **Commit**

Once committed, Aviatrix will write the rules to the Aviatrix Gateway(s)

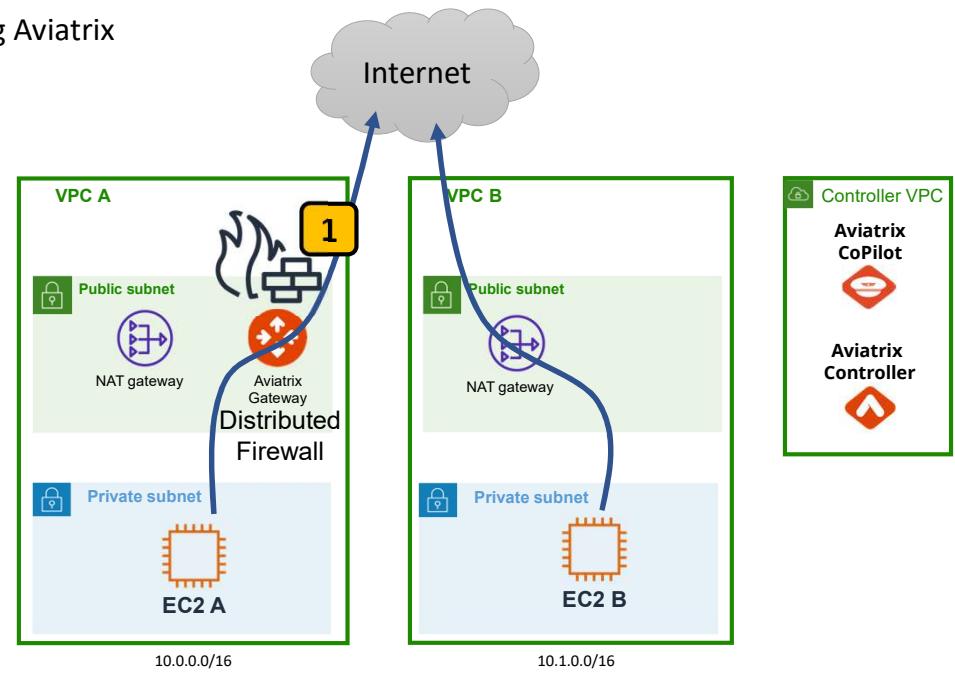
## Lab 2: Checkpoint 3

Simple egress NAT using Aviatrix

At this point you just deployed the Aviatrix Distributed Firewall on your Aviatrix Gateway in VPC A. **1**

Your EC2 A instance will only be able access the internet according to the firewall rules you created.

**Next: Let's test the internet again from the instance EC2 A and see if we have secured our egress traffic using the Aviatrix Distributed Firewall.**



AWS us-east-1

Note: Aviatrix does not charge data processing charges for Firewall and NAT. The Aviatrix cost for Distributed Firewall and NAT is **\$0.23 /hr per gateway** (plus the EC2 gateway instance charges)



## Lab 2: Step 2.31

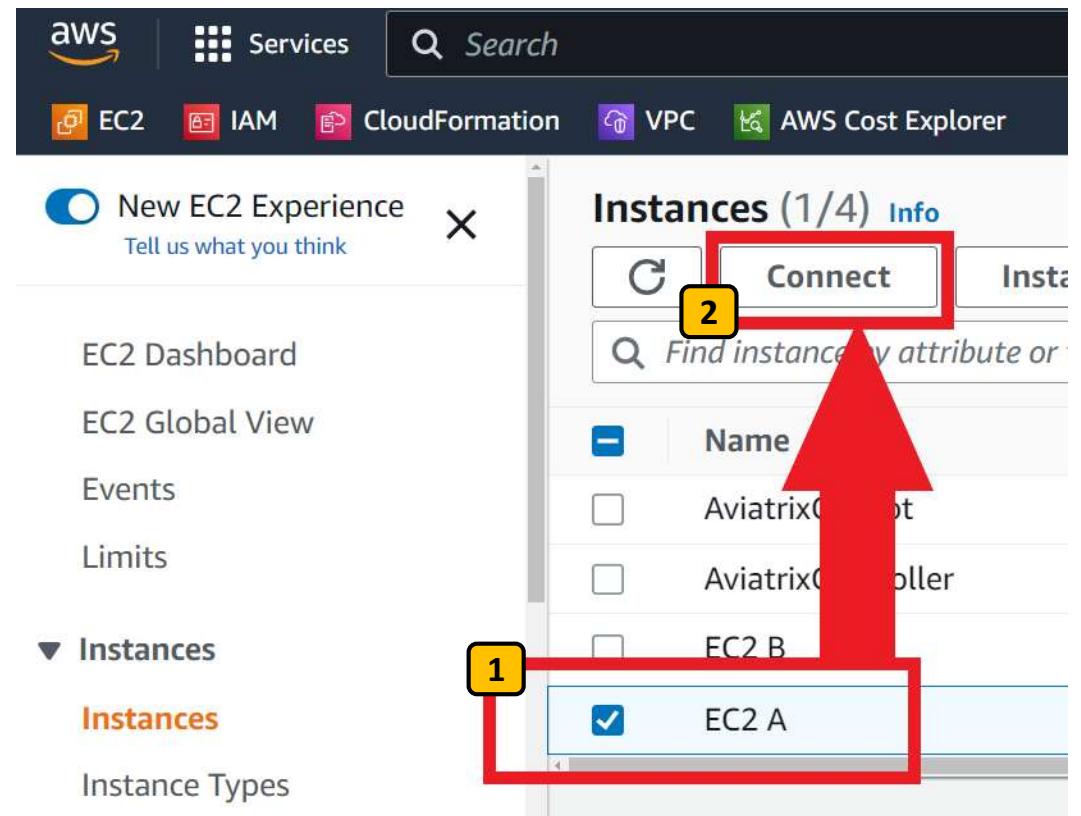
Connect to EC2 A instance console

Go the EC2 section of the AWS Console.

Select Instances

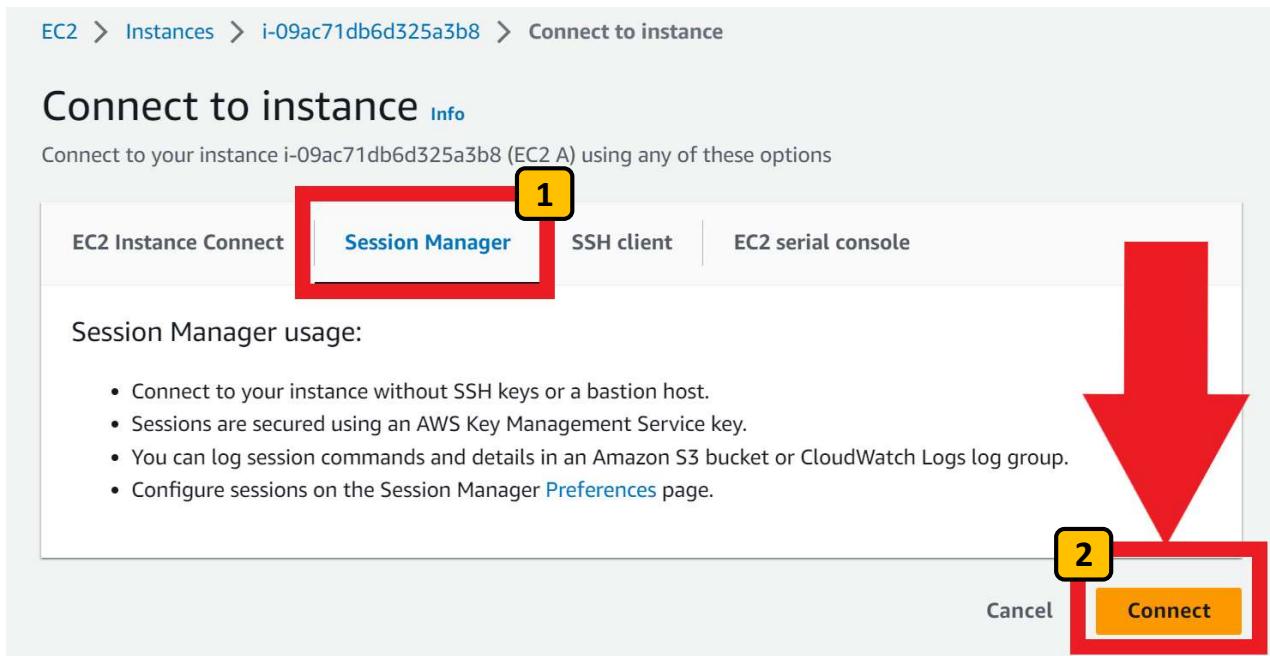
Find the EC2 A instance and select it. **1**

Click the Connect button **2**



## Lab 2: Step 2.32

Connect to EC2 A instance console



Select the Session Manager tab. **1**

Click the Connect button **2**

**Note:** If you get an error here, try rebooting EC2 A to accelerate the process of Session Manager reconnecting through the Aviatrix Gateway



## Lab 2: Step 2.33

Connect to EC2 A instance console

Your browser should open a new tab giving you a CLI session.

Type the command:

**sudo su -l ec2-user** 1

*(dash lower case L)*

You are now logged on as the ec2-user and should see your private IP address in the hostname.

Next, let's test internet egress.

A screenshot of a web-based terminal window titled "Session ID: brad-000841717aa667ae4" and "Instance ID: i-09ac71db6d325a3b8". The URL is "us-east-1.console.aws.amazon.com/systems-manager/session-manager/i-09ac71db6d325a3b8?region=us-east-1". The terminal window has a black background and white text. It shows a command-line interface with several "sh-4.2 \$" prompts. In the middle of the session, the command "sudo su -l ec2-user" is entered, followed by a password confirmation "[ec2-user@ip-10-0-2-10 ~]\$". A red rectangular box highlights this command entry. A yellow box labeled "1" is positioned over the top right corner of the red box. The session continues with several more "[ec2-user@ip-10-0-2-10 ~]\$" prompts at the bottom.



Test that your EC2 A instance has secured internet access that will prevent it from connecting to potentially harmful domains.

Run the commands:

curl https://ransomware.org 1

curl https://malware.net 2

curl https://botnet.com 3

For each command you should see the CLI return an SSL Error, because the Aviatrix Distributed Firewall is blocking the connection 4

## Lab 2: Step 2.34

Test connections to potentially harmful domains

The image shows four screenshots of an AWS Systems Manager session (Session ID: brad-000841717aa667ae4, Instance ID: i-09ac71db6d325a3b8) on an EC2 instance. Each screenshot displays a terminal window with the following command being typed:

- Screenshot 1: curl https://ransomware.org 1
- Screenshot 2: curl https://malware.net 2
- Screenshot 3: curl https://botnet.com 3
- Screenshot 4: curl https://ransomware.org 4

In Screenshot 4, the command is followed by an SSL error message:

```
curl: (35) OpenSSL/1.0.2k-fips: error:14077410:SSL routines:SSL3_GET_SERVER_HELLO
```

A large red 'X' is overlaid on the bottom right of Screenshot 4.



## Lab 2: Step 2.35

Run a software update

Let's make sure we can still run a software update from the AWS repositories

Run the command:

**sudo yum update -y** 1

This will connect to **amazonaws.com** domains to download software updates.

Great! Software updates are working because of the Allow-AWS rule in the Aviatrix Distributed Firewall.

A screenshot of a web browser window showing an AWS Systems Manager session. The URL is `us-east-1.console.aws.amazon.com/systems-manager/session-manager/i-09ac71db6d325a3b8?region=us-east-1`. The session ID is `brad-000841717aa667ae4` and the instance ID is `i-09ac71db6d325a3b8`. The terminal window shows the command `sudo yum update -y` being typed. A red box highlights the command, and a yellow box with the number **1** is positioned above it.

A screenshot of a terminal window showing the output of the `sudo yum update -y` command. The session ID is `brad-000841717aa667ae4` and the instance ID is `i-09ac71db6d325a3b8`. The output shows the command being run and the system responding that no packages are marked for update. A large red checkmark is overlaid on the bottom right of the terminal window.

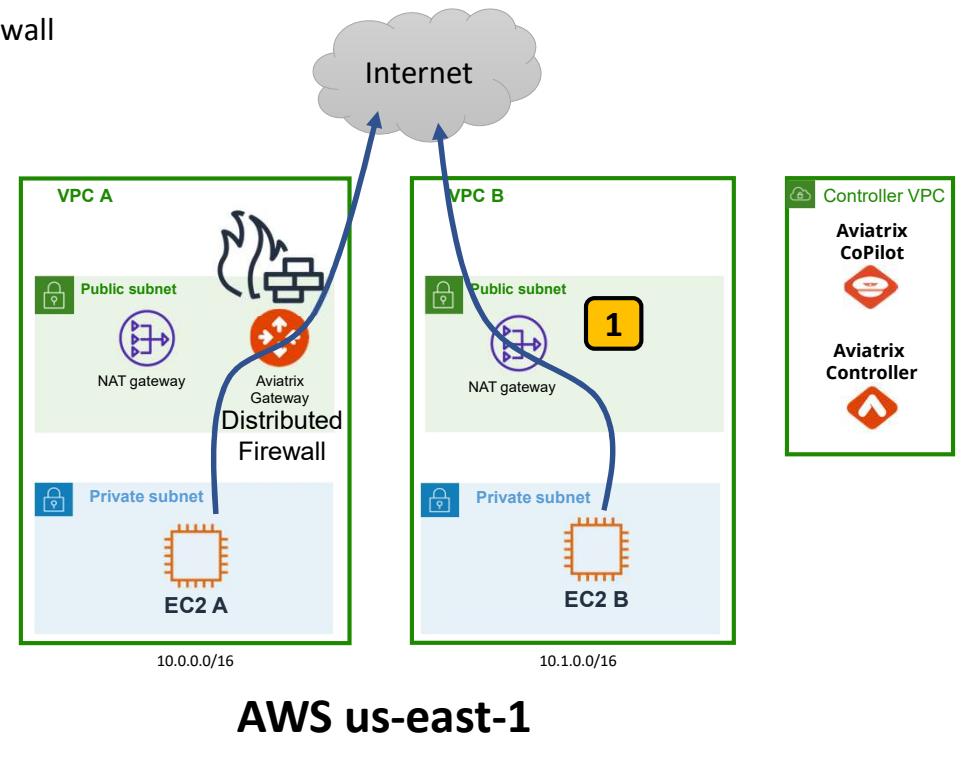
## Lab 2: Checkpoint 4

Aviatrix Distributed Firewall

At this point you've successfully deployed and tested the Aviatrix Distributed Firewall for VPC A.

Your EC2 A instance will only be able access the internet according to the firewall rules you created.

**Next: Let's extend the Distribute Firewall to VPC B, only this time we will use the same EIP that the AWS NAT Gateway is 1 using. Because maybe you have that EIP whitelisted with a partner or customer you work with, and you want to keep it.**





## Lab 2: Step 2.36

Delete your AWS NAT Gateways

NAT gateways (1/2) [Info](#)

Name	NAT gateway ID	Connectivit...
NGW VPC A	nat-061c67d74aa528ced	Public
<b>NGW VPC B</b>	nat-009cc8d3d6a4a0611	Public

Actions ▾ [Create NAT gateway](#)

View details  
Edit secondary IPv4 address associations  
Manage tags **3**  
**Delete NAT gateway**

Primary public IPv4..  
44.215.7.234 **2**  
3.220.133.116

Go to the VPC section of the AWS Console and select **NAT gateways** **1**

Find the NAT Gateway for VPC B and take note of the public IP address **2**

Select **NGW VPC B** and delete it by choosing **Actions** then **Delete NAT gateway** **3**

Once the NAT Gateway is deleted, we can use its EIP for the Aviatrix Gateway



## Lab 2: Step 2.37

Delete your AWS NAT Gateways

You successfully deleted nat-009cc8d3d6a4a0611 (NGW VPC B).

NAT gateways (2) <a href="#">Info</a>						
<input type="text"/> Filter NAT gateways						
	Name	NAT gateway ID	Connectivit...	State	State message	Primary public IPv4...
<input type="radio"/>	NGW VPC A	nat-061c67d74aa528ced	Public	<span>Available</span>	-	44.215.7.234
<input type="radio"/>	NGW VPC B	nat-009cc8d3d6a4a0611	Public	<span>Deleted</span>	1	3.220.133.116

Wait for the NAT Gateway for VPC B to show as **Deleted** 1

**Optional:** While you're waiting, you can also delete the NAT Gateway for VPC A, as it's not longer being used.

## Lab 2: Step 2.38

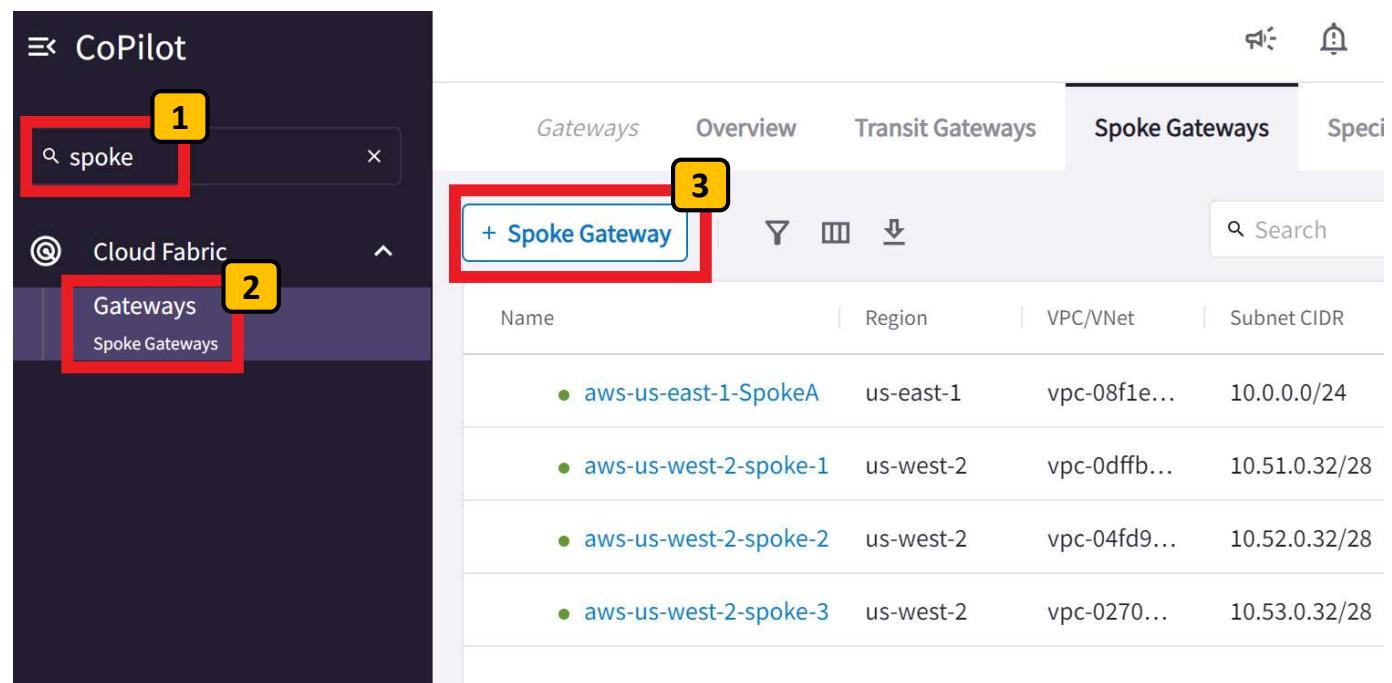
Deploy Aviatrix Spoke Gateway in VPC B Public Subnet

From the left-hand navigation in Aviatrix CoPilot...

Type the word **spoke** in the CoPilot search bar **1**

Select the **Spoke Gateways** search result **2**

Click the **+Spoke Gateway** button to deploy a new Spoke Gateway for VPC B. **3**



The screenshot shows two side-by-side views of the Aviatrix CoPilot interface.

**Left View (Search Results):** A dark-themed search interface. A red box highlights the search bar containing the text "spoke". A yellow box labeled "1" highlights the search result "Spoke Gateways" under the "Cloud Fabric" section. A red box highlights the "Gateways" tab.

**Right View (Spoke Gateways List):** A light-themed list view. The top navigation bar has tabs: Gateways, Overview, Transit Gateways, Spoke Gateways (which is highlighted with a red box and a yellow box labeled "3"), and Specs. Below the tabs is a search bar with the placeholder "Search". A red box highlights the "+ Spoke Gateway" button. The main area lists four existing Spoke Gateways:

Name	Region	VPC/VNet	Subnet CIDR
aws-us-east-1-SpokeA	us-east-1	vpc-08f1e...	10.0.0.0/24
aws-us-west-2-spoke-1	us-west-2	vpc-0dff...	10.51.0.32/28
aws-us-west-2-spoke-2	us-west-2	vpc-04fd9...	10.52.0.32/28
aws-us-west-2-spoke-3	us-west-2	vpc-0270...	10.53.0.32/28



## Lab 2: Step 2.39

Deploy Aviatrix Spoke Gateway in VPC B Public Subnet

Create Spoke Gateway

Name **1**: aws-us-east-1-SpokeB

Cloud **2**: AWS Standard

Account **3**: aws-account

Region **4**: us-east-1 (N. Virginia)

VPC/VNet **5**: VPC B

Instance Size **6**: t3.medium

High Performance Encryption: Off

Attach To Transit Gateway: Optional

Name the gateway **aws-us-east-1-SpokeB** **1**

Select the cloud AWS standard **2**

Select the account **aws-account** **3**

Select the region **us-east-1** **4**

Select **VPC B** **5**

Select the **t3.medium** instance size **6**



## Lab 2: Step 2.40

Deploy Aviatrix Spoke Gateway in VPC B Public Subnet

Instances

+ Instance

	Attach to Subnet	Public IP	
1	10.1.0.0/24~~us-east-1a~~VPC B - AZ1	3.220.133.116	

Cancel 9 Save

Select the 10.1.0.0/24 subnet in **AZ1** 7

Select the Public IP that the NAT Gateway in VPC B was using before it was deleted. 8

Click **Save** 9



## Lab 2: Step 2.41

Monitor the gateway deployment Task

The screenshot shows the CoPilot interface. On the left is a sidebar with the following items:

- CoPilot (selected)
- task (highlighted with a red box and yellow number 1)
- Monitor
- Notifications (highlighted with a red box and yellow number 2)
- Tasks
- Settings
- Resources
- Task

The main area has tabs: Notifications, Alerts, Alerts Configuration, System Messages, Tasks (selected), and Recipients. Below is a sub-tab: Tasks (selected) and Active Gateway Operations. A search bar is at the top right. The main table lists tasks:

Name	Entity	Status	Progress
Create spoke gateway: aws-us-east-1-SpokeB		In Progress	[Progress Bar]
Create spoke gateway: aws-us-east-1-SpokeA		Completed	[Progress Bar]

From the CoPilot search bar type **task** 1

Click the search result **Notifications / Tasks** 2

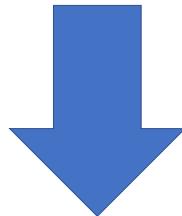
Observe the spoke gateway creation Task and wait for it to complete 3



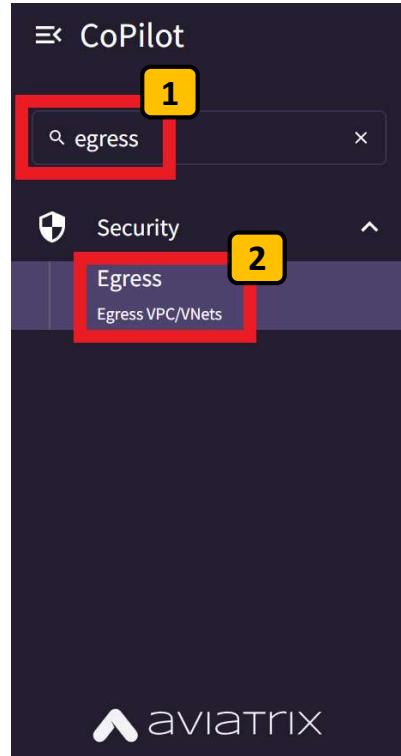
## Lab 2: Step 2.42

Enable Egress on the Aviatrix Spoke Gateway

Now let's configure your new Aviatrix Spoke Gateway to do Local Egress for **VPC B**, so we can use it for egress NAT from our private subnets.



From the CoPilot search bar type **egress** **1**



Click the search result **Egress VPC/VNets** **2**

Name	Point of Egress
aws-us-east-1-SpokeA	Local Egress
aws-us-west-2-spoke-1	Local Egress
aws-us-west-2-spoke-2	Local Egress
aws-us-west-2-spoke-3	Local Egress
aws-us-east-1-SpokeB	Native Cloud Egress

Total 5 VPC/VNets

Click the **+ Local Egress on VPC/VNets** button **3**

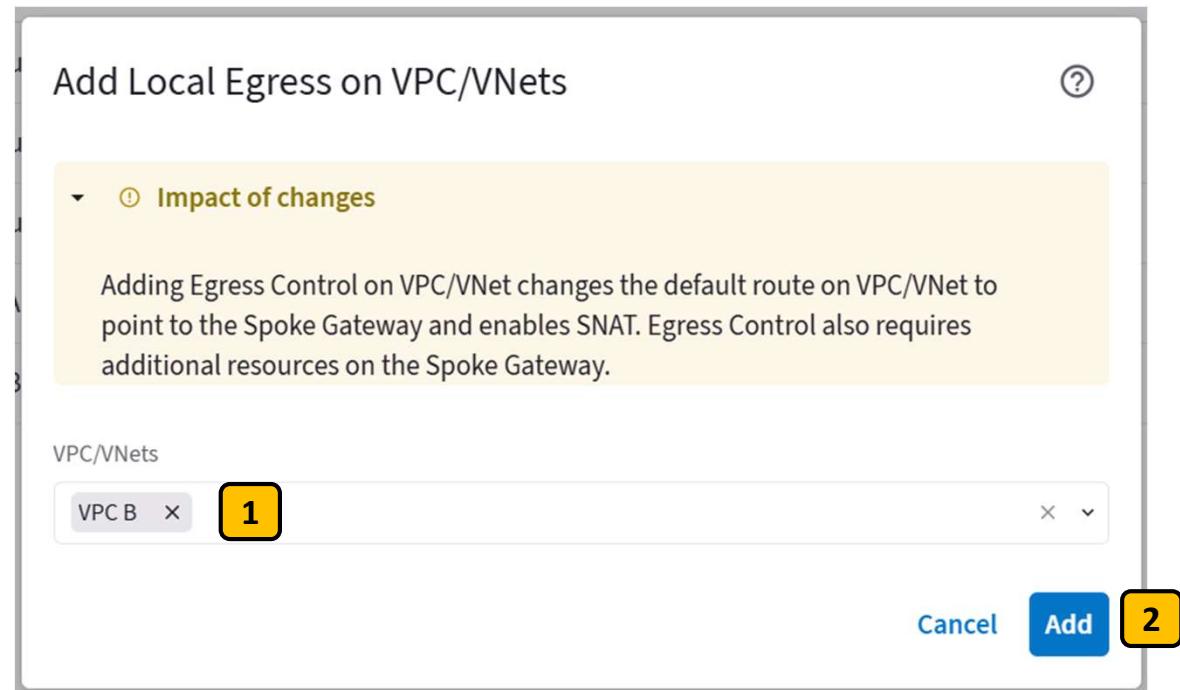
## Lab 2: Step 2.43

Enable Egress on the Aviatrix Spoke Gateway

In the Add Local Egress on VPC/VNets pop-up, select the new **aws-us-east-1-SpokeB** gateway from the VPC/VNets drop down **1**

Click **Add** **2**

After you click Add, Copilot will change the VPC default route associated to all private subnets in VPC B to point to your new Aviatrix Gateway



## Lab 2: Checkpoint 5

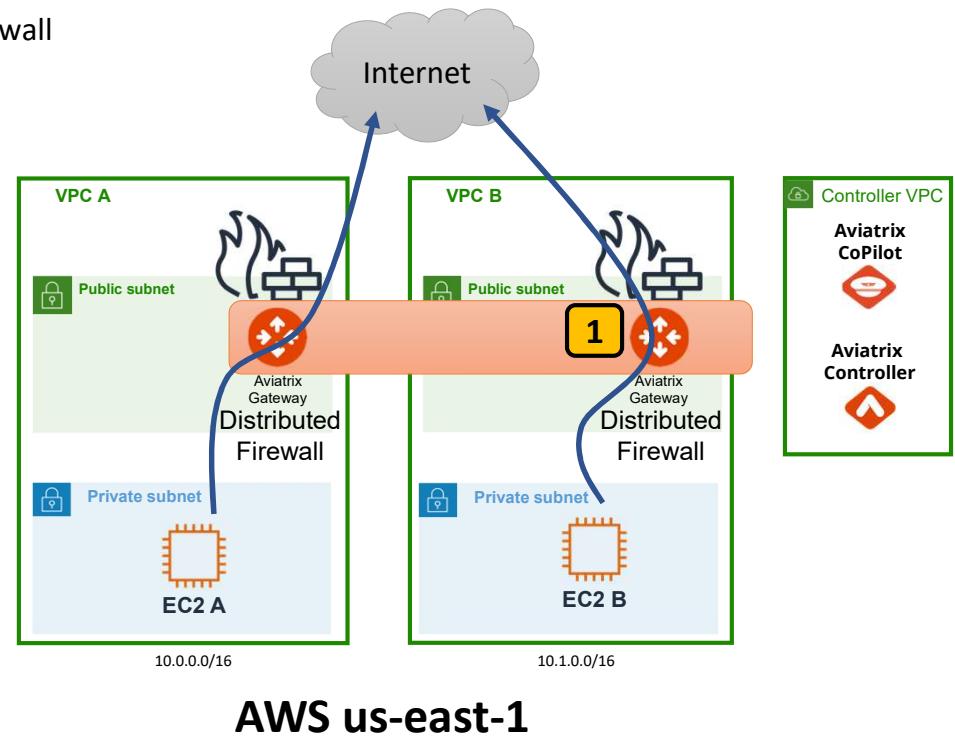
Aviatrix Distributed Firewall

At this point you've just extended your Aviatrix Distributed Firewall to VPC B. **1**

The policies that you've already centrally configured in Aviatrix CoPilot have been extended and applied to VPC B.

Egress traffic for the instance EC2 B is now flowing through the Aviatrix Gateway in VPC B where firewall rules are enforced.

**Next: Let's test the internet again from the instance EC2 B and see if it's already secured.**



AWS us-east-1

Note: Aviatrix does not charge data processing charges for Firewall and NAT. The Aviatrix cost for Distributed Firewall and NAT is **\$0.23 /hr per gateway** (plus the EC2 gateway instance charges)



## Lab 2: Step 2.45

Connect to EC2 B instance console

Go the EC2 section of the AWS Console.

Select Instances 1

Find the EC2 B instance and select it. 2

Click the Connect button 3

The screenshot shows the AWS EC2 Instances page. At the top, there's a navigation bar with the AWS logo, a 'Services' dropdown, and search/filter icons. Below the navigation is a modal for the 'New EC2 Experience' asking for feedback. On the left, a sidebar has 'Instances' selected (highlighted with a red box and a yellow '1') and shows other options like 'Instance Types', 'Launch Templates', and 'Spot Requests'. The main content area is titled 'Instances (1/6)' and shows a list of instances. One instance, 'EC2 B', is selected (highlighted with a red box and a yellow '2') and has a checked checkbox next to its name. A 'Connect' button is also highlighted with a red box and a yellow '3'. A search bar at the bottom allows finding instances by attribute or tag.

Name
AviatrixCopilot
AviatrixController
aviatrix-aws-us-east-1-SpokeB
<input checked="" type="checkbox"/> EC2 B
EC2 A
aviatrix-aws-us-east-1-SpokeA



## Lab 2: Step 2.46

Connect to EC2 B instance console

EC2 > Instances > [i-02898ecd9feafb91a](#) > Connect to instance

### Connect to instance Info

Connect to your instance i-02898ecd9feafb91a (EC2 B) using any of these options

EC2 Instance Connect    **Session Manager** 1    SSH client    EC2 serial console

Session Manager usage:

- Connect to your instance without SSH keys or a bastion host.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager [Preferences](#) page.

Cancel 2 **Connect**

Select the Session Manager tab. 1

Click the Connect button 2

*Note:* If you get an error here, try rebooting EC2 B to accelerate the process of Session Manager reconnecting through the Aviatrix Gateway



## Lab 2: Step 2.47

Connect to EC2 B instance console

Your browser should open a new tab giving you a CLI session.

Type the command:

**sudo su -l ec2-user** 1

*(dash lower case L)*

You are now logged on as the ec2-user and should see your private IP address in the hostname.

Next, let's test internet egress.

```
Session ID: brad-  
0104b73d04-7dfb04  
sh-4.2$  
sh-4.2$  
sh-4.2$ sudo su -l ec2-user1  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$
```



Test that your EC2 B instance has secured internet access that will prevent it from connecting to potentially harmful domains.

Run the commands:

`curl https://ransomware.org` 1

`curl https://malware.net` 2

`curl https://botnet.com` 3

For each command you should see the CLI return an SSL Error, because the Aviatrix Distributed Firewall is blocking the connection 4

## Lab 2: Step 2.48

Test connections to potentially harmful domains

```
Session ID: brad-  
0f07141bd207757a6  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$ curl https://ransomware.org
```

1

```
Session ID: brad-  
0f07141bd207757a6  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$ curl https://malware.net
```

2

```
Session ID: brad-  
0f07141bd207757a6  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$ curl https://botnet.com
```

3

```
Session ID: brad-  
0f07141bd207757a6  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$ curl https://botnet.com  
curl: (35) error:14077410:SSL routines:SSL23_GET_SERVER_HELLO:sslv3 alert
```



4



## Lab 2: Step 2.49

Run a software update

Let's make sure we can still run a software update from the AWS repositories

Run the command:

**sudo yum update -y** 1

This will connect to **amazonaws.com** domains to download software updates.

Great! Software updates are working because of the Allow-AWS rule in the Aviatrix Distributed Firewall.

```
Session ID: brad-  
0f07141bd207757a6  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$  
[ec2-user@ip-10-1-2-10 ~]$ sudo yum update -y1
```

```
xz.x86_64 0:5.2.2-1.amzn2.0.3  
xz-libs.x86_64 0:5.2.2-1.amzn2.0.3  
yum.noarch 0:3.4.3-158.amzn2.0.6  
zlib.x86_64 0:1.2.7-19.amzn2.0.2  
  
Replaced:  
grub2.x86_64 1:2.06-2.amzn2.0.3  
python-colorama.noarch 0:0.3.2-3.amzn2.0.2  
grub2-tools.x86_64 1:2.06-2.amzn2.0.3  
python-six.noarch 0:1.11.0-1.amzn2.0.2  
  
Complete!  
[ec2-user@ip-10-1-2-10 ~]$ 1
```

## Lab 2: Checkpoint 6

Aviatrix Distributed Firewall

At this point you've just validated that your Aviatrix Distributed Firewall extended **1** seamlessly to VPC B.

You also reused the EIP from the AWS NAT Gateway that was there before.

In addition to security, The Aviatrix Distributed Firewall also provides deep traffic visibility, monitoring, alerting, and scaling policies.



Next, let's look at traffic details, set monitoring alerts, and create a scaling policy.



## Lab 2: Step 2.50

See the Policy actions

The screenshot shows the CoPilot interface for the Distributed Cloud Firewall. On the left sidebar, the 'Distributed Cloud Firewall' option is selected and highlighted with a red box. At the top, there are tabs for 'Monitor' (which is highlighted with a yellow box and labeled '1'), 'Rules', 'Detected Intrusions', 'WebGroups', and 'Settings'. Below these tabs is a search bar and a refresh button. The main area displays a table of logs with the following columns: Timestamp, Rule, L4/L7, Source SmartGroup, Action, and Enforced. The 'Action' column is highlighted with a red box and labeled '2'. The table contains six rows of log entries, all showing the 'Allow-AWS' rule, L4 or L7 protocol, PROD source group, and Permit action. A note at the bottom states: 'Monitor logs the last 1,000,000 packets that hit the Policy Rules in Distributed Cloud Firewall. Total 83471 logs.'

Timestamp	Rule	L4/L7	Source SmartGroup	Action	Enforced
Aug 14, 2023 3:42:11 PM	Allow-AWS	L4	PROD	Permit	Yes
Aug 14, 2023 3:42:11 PM	Allow-AWS	L4	PROD	Permit	Yes
Aug 14, 2023 3:42:11 PM	Allow-AWS	L7	PROD	Permit	Yes
Aug 14, 2023 3:42:11 PM	Allow-AWS	L7	PROD	Permit	Yes
Aug 14, 2023 3:42:10 PM	Allow-AWS	L4	PROD	Permit	Yes

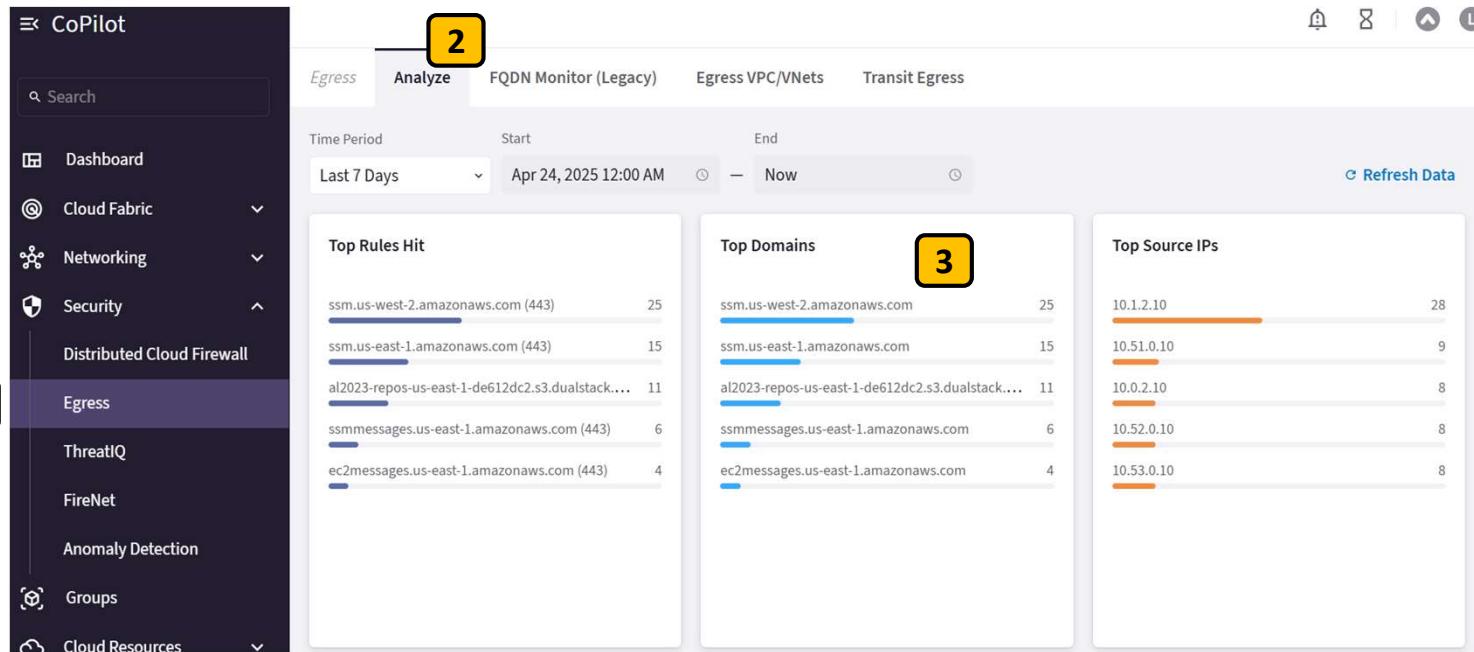
In the Distributed Cloud Firewall section of CoPilot select **Monitor** tab **1**

Observe the session details and policy decisions as traffic flows through the Distributed Cloud Firewall. **2**



## Lab 2: Step 2.51

Observe the Domain traffic activity



From the CoPilot navigation select Egress under Security [1]

Select the Analyze tab [2]

Observe domain traffic details including Top Domains and Top Source IPs. [3]

Notice the **amazonaws.com** domains supporting Session Manager and software updates



## Lab 2: Step 2.52

Observe the Domain traffic activity

2

3

Timestamp	Source IP	VPC/VNet	Domain	Port	Rule Match	Action
May 1, 2025 1:50 PM	10.51.0.10	aws-us-wes...	ssm.us-wes...	443	Matched	Allowed
May 1, 2025 1:49 PM	10.51.0.10	aws-us-wes...	ssm.us-wes...	443	Matched	Allowed
May 1, 2025 1:48 PM	10.52.0.10	aws-us-wes...	ssm.us-wes...	443	Matched	Allowed
May 1, 2025 1:47 PM	10.52.0.10	aws-us-wes...	ssm.us-wes...	443	Matched	Allowed
May 1, 2025 1:45 PM	10.51.0.10	aws-us-wes...	ssm.us-wes...	443	Matched	Allowed
May 1, 2025 1:42 PM	10.52.0.10	aws-us-wes...	ssm.us-wes...	443	Matched	Allowed
May 1, 2025 1:40 PM	10.51.0.10	aws-us-wes...	ssm.us-wes...	443	Matched	Allowed
May 1, 2025 1:39 PM	10.51.0.10	aws-us-wes...	ssm.us-wes...	443	Matched	Allowed

From the CoPilot navigation select Egress under Security **1**

Select the FQDN Monitor tab **2**

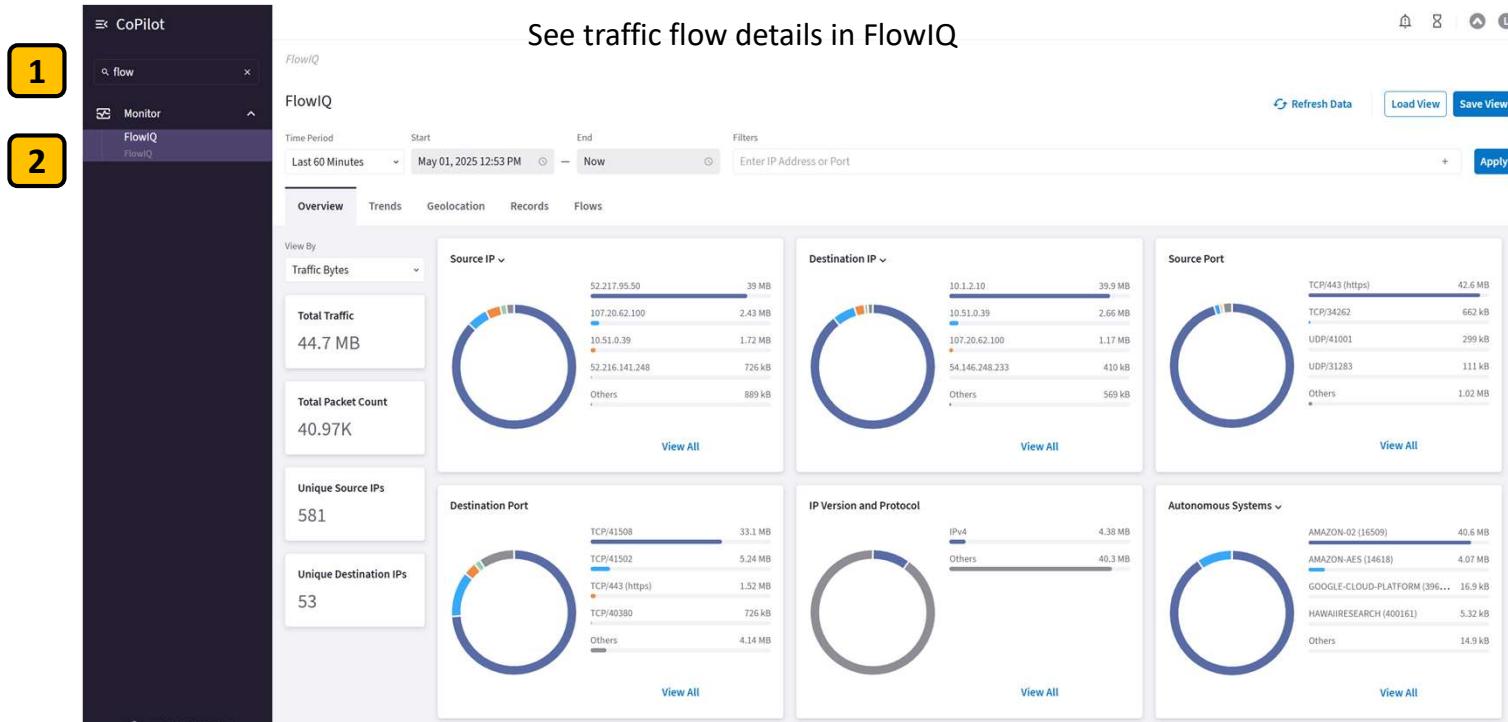
Select Spoke-1 and Spoke-2 gateways in the VPC/VNets drop-down to monitor domain activity flowing through them **3**

**Challenge:** Try to generate domain traffic on EC2 A using curl and come back to this screen to see if you can see the activity.



## Lab 2: Step 2.53

See traffic flow details in FlowIQ



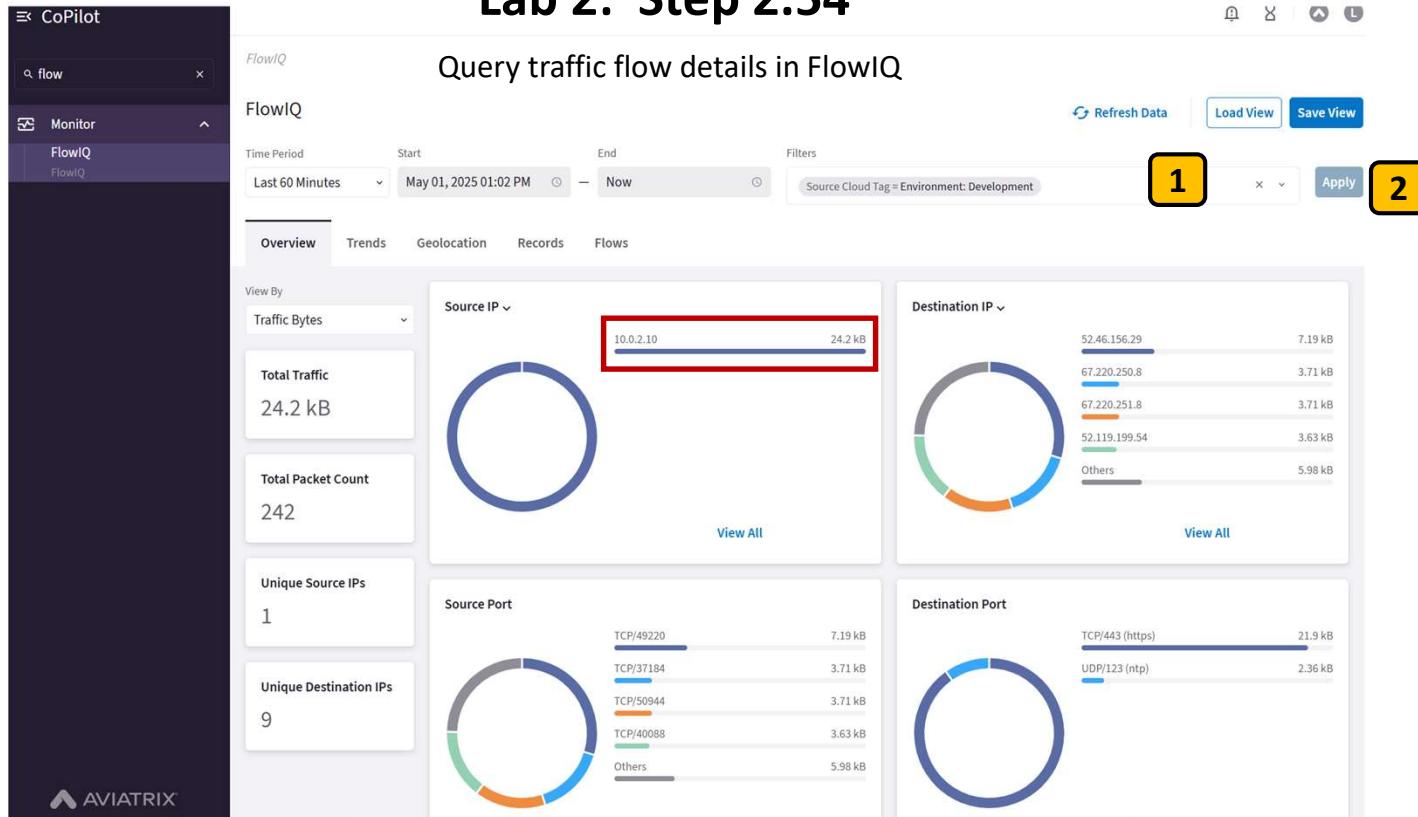
In the CoPilot search bar type **flow** 1

Select the **FlowIQ** search result 2

**FlowIQ** provides an intuitive dashboard to visualize and query traffic details as it flows through your Aviatrix Gateways.



## Lab 2: Step 2.54



**1** Select the Filters box and filter on traffic sourced from only EC2 instances of a **Source Cloud Tag** with the variables **Environment = Development**

**2** Click **Apply** to apply the filter

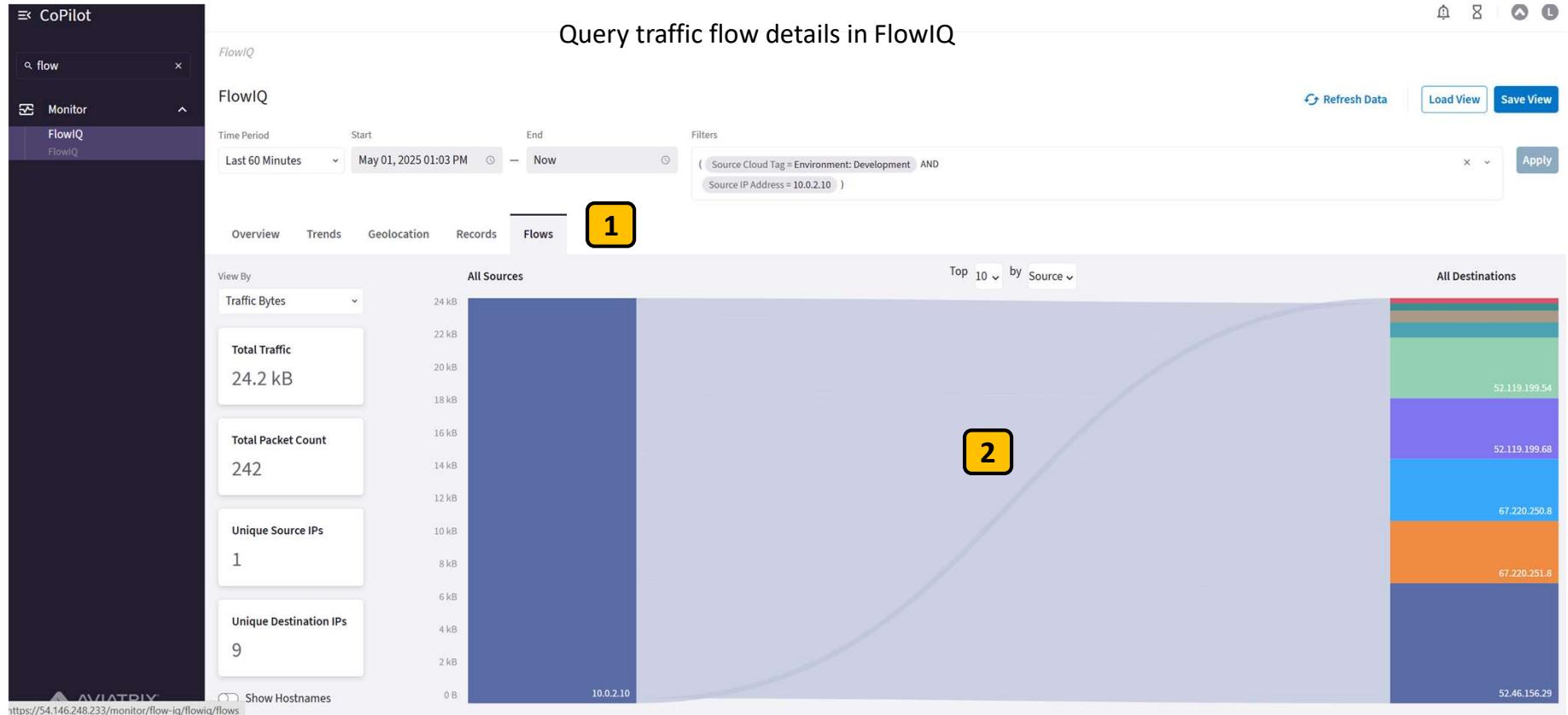
You immediately found the top talker in the **Development** environment without even thinking about IP addresses!

Click the top Source IP address narrow in on this talker and click **Apply again**.



## Lab 2: Step 2.55

Query traffic flow details in FlowIQ



1 Select the Flows tab to see a flow summary from this top talker in Development

2 Hover your mouse over each flow to get more details.



## Lab 2: Step 2.56

Query traffic flow details in FlowIQ

The screenshot shows the CoPilot interface with the FlowIQ tab selected. The 'Records' tab is highlighted with a red box. A yellow box labeled '1' points to the 'Records' tab in the navigation bar. The table below shows three rows of network flow data:

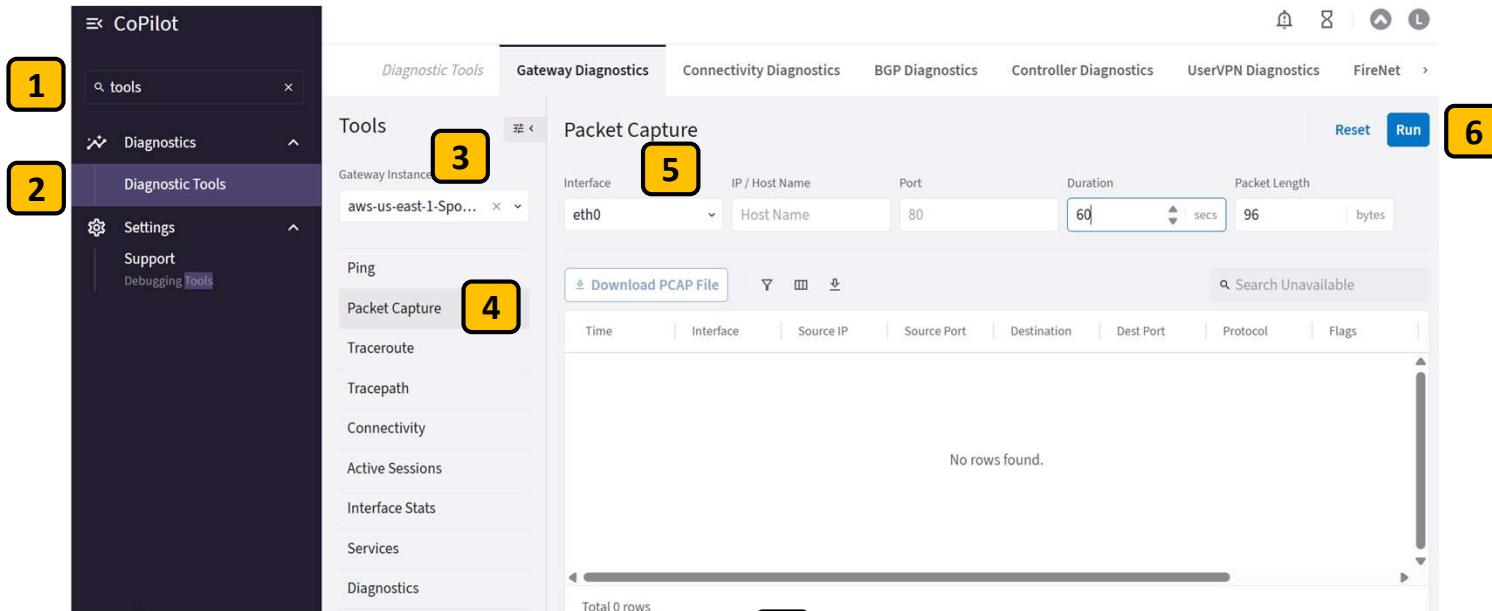
Timestamp	Host	Destination Address	Source Address	Bytes	Direction	Packets	Duration	Throughput
May 1, 2025 1:58:08 PM	52.207.76.185	52.46.156.29	10.0.2.10	1.38 kB	forward	29	7.772 s	
May 1, 2025 1:58:08 PM	52.207.76.185	44.201.148.133	10.0.2.10	76 B	forward	1	0 s	
May 1, 2025 1:58:08 PM	52.207.76.185	54.210.225.137	10.0.2.10	76 B	forward	1	0 s	

Select the **Records** tab to see details of each traffic flow record from this top talker in Development **1**

Aviatrix Gateways export full Netflow information to CoPilot and you're seeing that raw data here.

## Lab 2: Step 2.57

Take a packet capture



**1** In the CoPilot search bar type **tools**

**4** Select **Packet Capture**

**2** Select the **Diagnostic Tools** search result

**5** Select the **eth0** interface

**3** Select **aws-us-east-1-SpokeA** as the gateway to capture from

**6** Select **Run** and **Download PCAP file** when the capture completes.



## Lab 2: Step 2.58

Configure monitoring alerts

The screenshot shows the CoPilot interface for managing monitoring alerts. On the left, there's a sidebar with 'CoPilot' at the top, followed by a search bar containing 'alerts' with a red box around it. Below the search bar is a 'Monitor' section with 'Notifications' and 'Alerts' options, and 'Notifications' is currently selected, with its sub-option 'Alerts Configuration' also highlighted with a red box. The main content area has tabs at the top: 'Notifications', 'Alerts' (which is active), 'Alerts Configuration' (highlighted with a red box), 'System Messages', 'Tasks', and 'Recipients'. Below the tabs is a table with columns: Name, Condition, Monitored Entities, and Recipients. There are two rows in the table:

Name	Condition	Monitored Entities	Recipients
Global Control Plane Health	CPU Used (%) more than 90% Memory Used (%) more tha... Disk Free (%) less than 5%	Controller, + 1 more	<span style="color: blue;">edit</span> <span style="color: green;">view</span> <span style="color: red;">delete</span>
Global Network Health	Gateway Status changed Limit Exceeded Rate (PPS) ...	All Gateways: All Interfaces	<span style="color: blue;">edit</span> <span style="color: green;">view</span> <span style="color: red;">delete</span>

In the CoPilot search bar type **alerts** 1

Select the **Alerts Configuration** search result 2

Click the **+ Alert Configuration** box 3

## Lab 2: Step 2.59

Configure monitoring alerts

Name the Alert **High CPU** 1

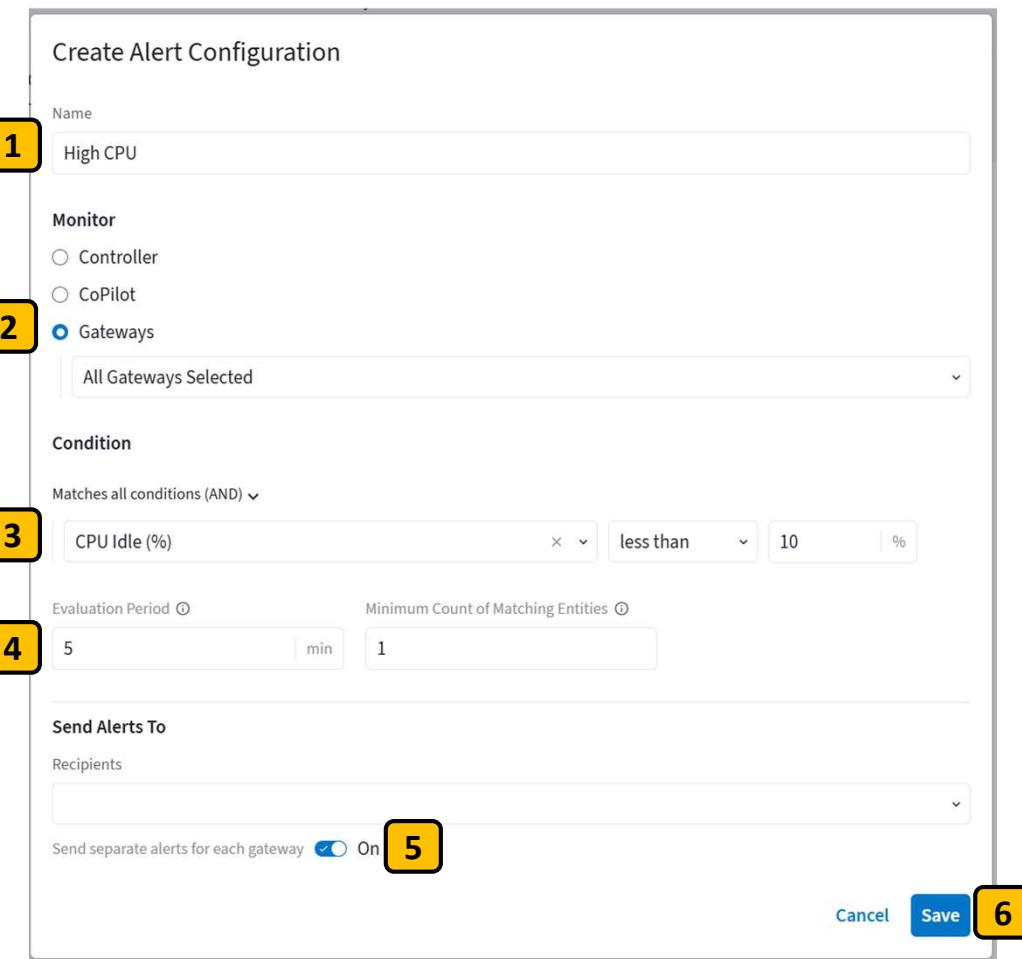
Select to monitor **Gateways** and **All Gateways Selected** 2

Match on the condition of **CPU Idle% less than 10%** 3

Set the **Evaluation Period** to 5 min and **Minimum Count** to 1 4

Toggle the **Send separate alerts for each gateway** button 5

Click **Save** 6



Create Alert Configuration

**1** Name  
High CPU

**2** Monitor  
 Controller  
 CoPilot  
 Gateways  
 All Gateways Selected

**3** Condition  
Matches all conditions (AND) ▾  
CPU Idle (%) less than 10 %

**4** Evaluation Period 5 min Minimum Count of Matching Entities 1

**5** Send Alerts To  
Recipients  
Send separate alerts for each gateway  On

**6** Save



## Lab 2: Step 2.60

### Auto-Scaling

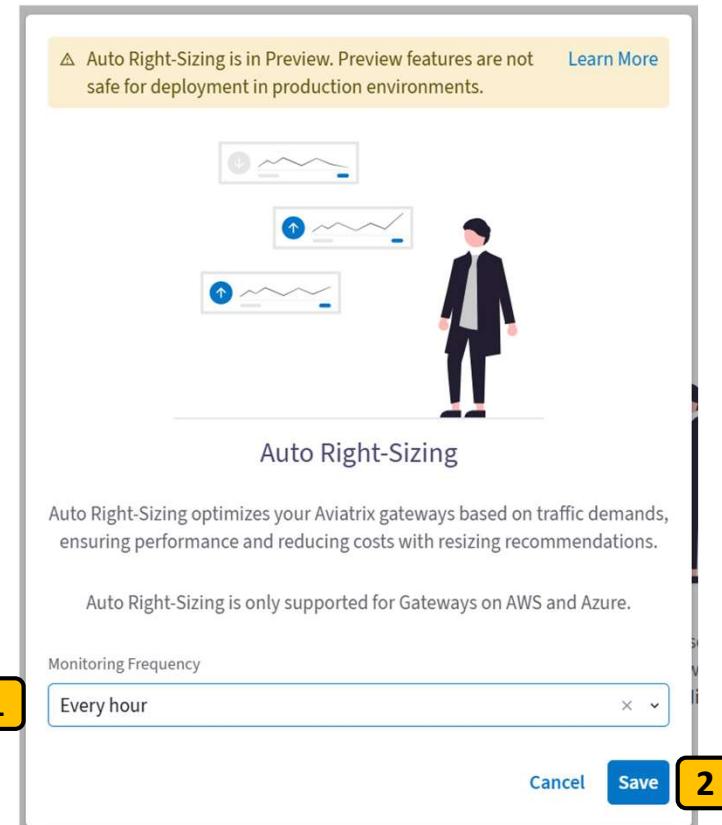
- 1 In the CoPilot search bar type **scaling**
- 2 Select the **Scaling** search result
- 3 Click the **Enable Auto Right-Sizing** button

The screenshot shows the Aviatrix CoPilot interface. On the left, a sidebar has a search bar with 'scaling' typed in, and a list below it with 'Cloud Fabric' and 'Scaling' (which is highlighted). A yellow box labeled '1' is over the search bar, and another yellow box labeled '2' is over the 'Scaling' result. On the right, the main window has tabs for 'Scaling', 'Auto Right-Sizing', and 'Events'. The 'Scaling' tab is active. A message at the top says 'Auto Right-Sizing is in Preview. Preview features are not safe for deployment in production environments.' Below this are sections for 'Actions', 'Settings', and 'Aviatrix Standard Sizes'. A large illustration shows three wavy lines representing traffic demand and a person standing next to them. At the bottom, a button labeled 'Enable Auto Right-Sizing' is highlighted with a red border, and a yellow box labeled '3' is over it.

## Lab 2: Step 2.61

Auto-Scaling

- 1 Under **Monitoring Frequency** choose **Every hour**
- 2 Select **Save**





## Lab 2: Step 2.62

### Auto-Scaling

A screenshot of the Aviatrix CoPilot web interface. The left sidebar has a dark theme with categories like Dashboard, Cloud Fabric (Topology, Gateways, Hybrid Cloud), Scaling (selected), UserVPN, Networking, Security, Groups, Cloud Resources, Monitor (FlowIQ, Performance, Traffic & Latencies). The main area has tabs for Scaling, Auto Right-Sizing (selected), and Events. A yellow banner at the top says "Auto Right-Sizing is in Preview. Preview features are not safe for deployment in production environments." with a "Learn More" link. Below is a search bar and a table header with columns: Gateway Instance, Gateway Type, VPC/VNet, Current Size, Recommended S, Urgency. The table body is empty with the message "No Active Recommendations". It also says "Based on the latest scan, all monitored gateways are right sized." and "The next scan is scheduled for May 1, 2025 3:00 PM". At the bottom, it says "Total 0 Recommendations".

CoPilot

Scaling Auto Right-Sizing Events

Auto Right-Sizing is in Preview. Preview features are not safe for deployment in production environments. [Learn More](#)

Actions [Settings](#) [Aviatrix Standard Sizes](#)

Gateway Instance | Gateway Type | VPC/VNet | Current Size | Recommended S | Urgency

No Active Recommendations

Based on the latest scan, all monitored gateways are right sized.

The next scan is scheduled for May 1, 2025 3:00 PM

Total 0 Recommendations

Congratulations, all your monitored gateways will not automatically right size themselves

## Lab 2: Complete

Aviatrix Distributed Firewall for Secure Egress

**Congratulations!** You've just deployed the Aviatrix Distributed Cloud Firewall for Secure Egress.

You seamlessly switched your traffic from AWS NAT Gateways, created firewall rules that filtered on domain names, and easily extended the firewall to a second VPC.

You also observed traffic details, set monitoring alerts and created a scaling policy.

