AWS Immersion Day
LAB 4

SECURITY: NETWORK SEGMENTATION

Brad Hedlund

**Principal Solutions Architect,
Aviatrix Systems**

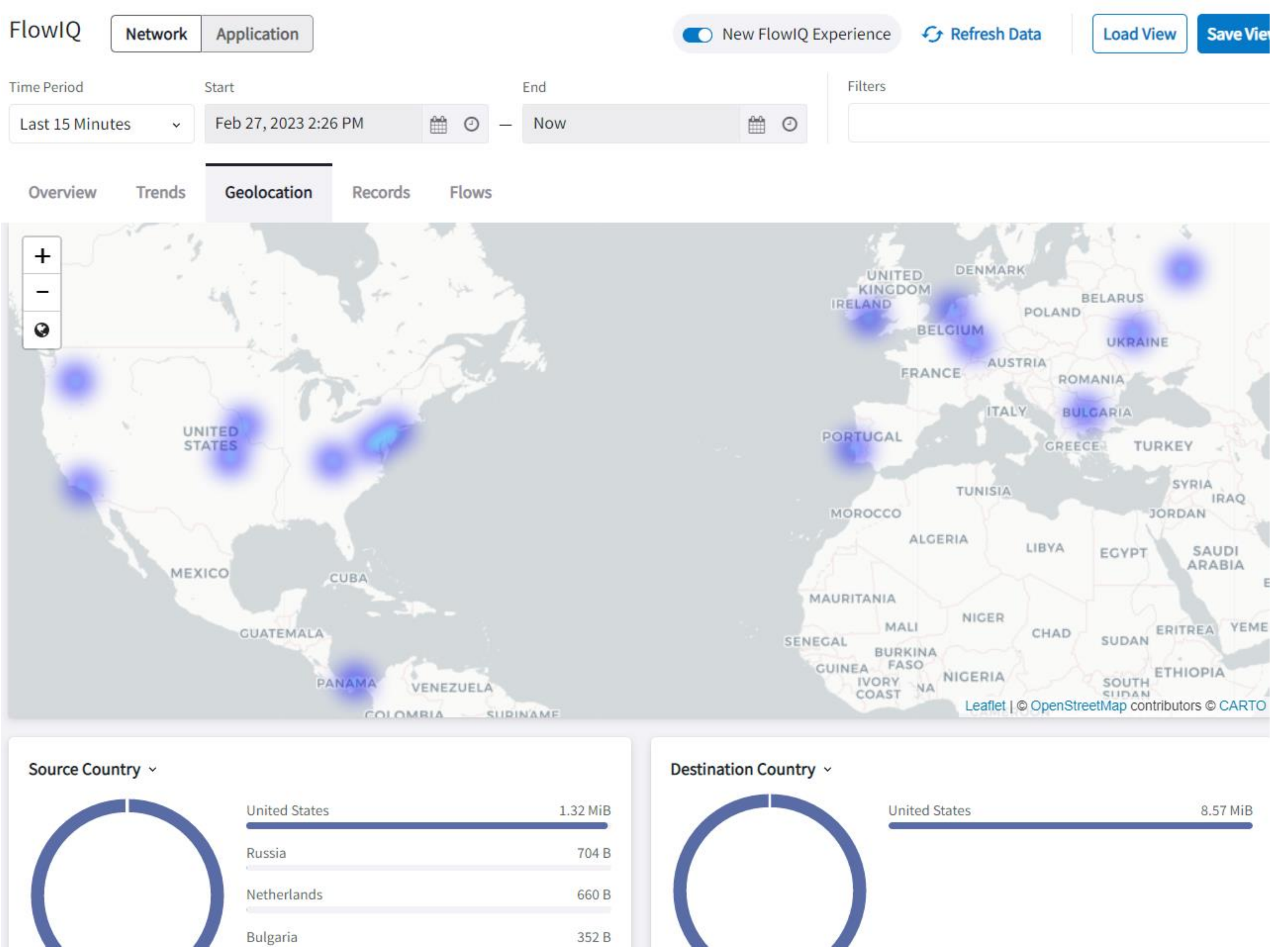
Lab 3 Recap

Deep network visibility with FlowIQ

You used FlowIQ to see all the traffic flows traversing your cloud backbone.

You built filters with simple mouse clicks to drill down on traffic details.

You observed how Aviatrix CoPilot is aware of the native CSP tags on your instances, and how you can build filters based on them.



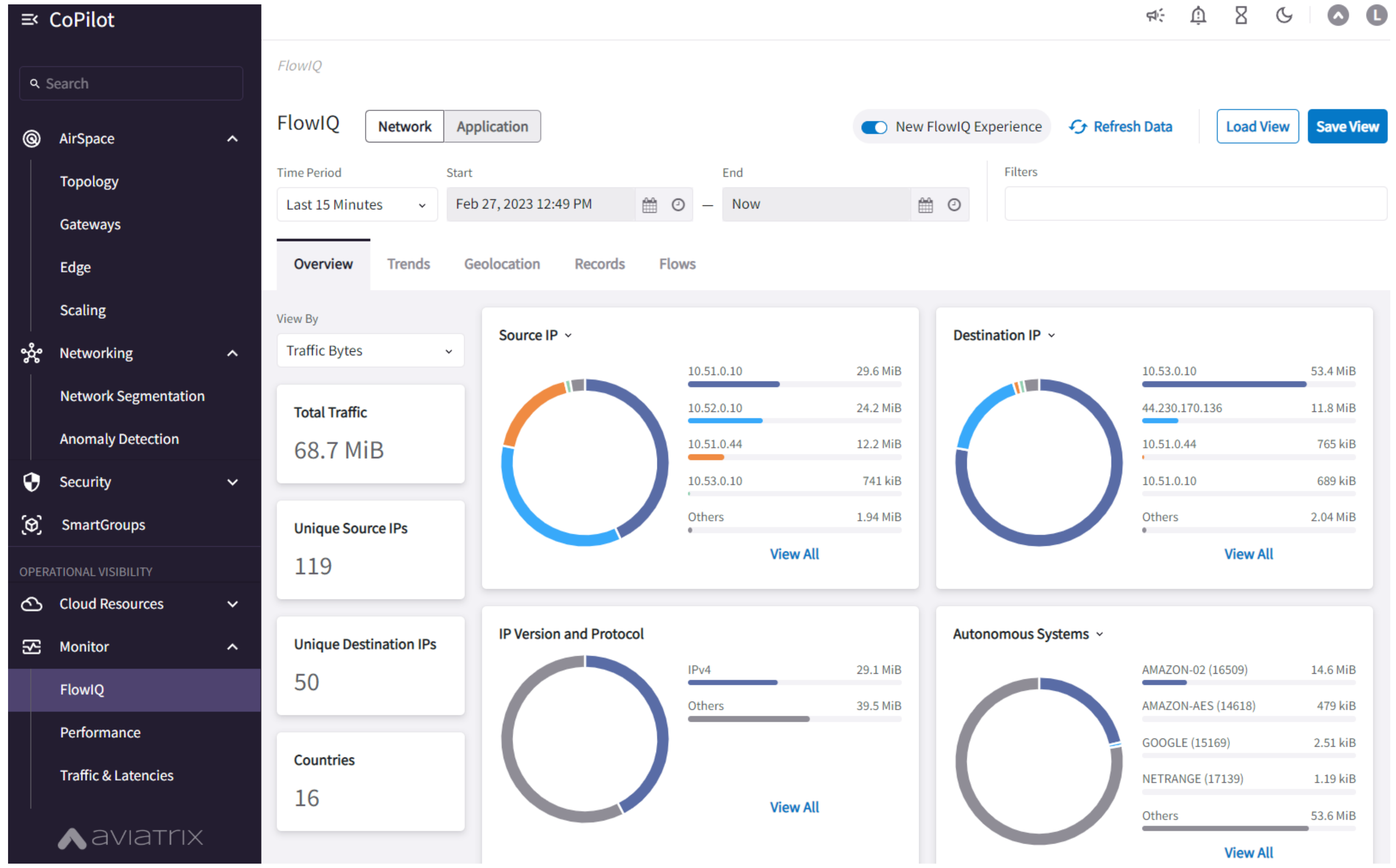
Time Period: Last 15 Minutes
Start: Feb 27, 2023 2:26 PM
End: Now
Filters:

Source Country

United States	1.32 MiB
Russia	704 B
Netherlands	660 B
Bulgaria	352 B

Destination Country

United States	8.57 MiB



Time Period: Last 15 Minutes
Start: Feb 27, 2023 12:49 PM
End: Now
Filters:

View By: Traffic Bytes

Total Traffic
68.7 MiB

Unique Source IPs
119

Unique Destination IPs
50

Countries
16

Source IP

10.51.0.10	29.6 MiB
10.52.0.10	24.2 MiB
10.51.0.44	12.2 MiB
10.53.0.10	741 kiB
Others	1.94 MiB

Destination IP

10.53.0.10	53.4 MiB
44.230.170.136	11.8 MiB
10.51.0.44	765 kiB
10.51.0.10	689 kiB
Others	2.04 MiB

IP Version and Protocol

IPv4	29.1 MiB
Others	39.5 MiB

Autonomous Systems

AMAZON-02 (16509)	14.6 MiB
AMAZON-AES (14618)	479 kiB
GOOGLE (15169)	2.51 kiB
NETRANGE (17139)	1.19 kiB
Others	53.6 MiB

Lab 4 Intro

Network Segmentation

AWS us-west-2
AWS us-east-1
10.51.0.0/16
Spoke 1
Jump Instance
10.51.0.10
SAP1
10.52.0.0/16
Spoke 2
10.52.0.10
SAP2
10.52.0.0/16
Spoke 3
10.53.0.10
SAP3
Transit
10.50.0.0/16
10.0.0.0/16
VPC A
EC2-A
10.1.0.0/16
VPC B
EC2-B
Controller VPC
Aviatrix CoPilot
Aviatrix Controller
10.2.0.0/16
VPC C (Transit)
EC2-C

Here is our cloud network backbone

Can we do isolation and segmentation?

Can we have a Firewall at every hop?

Yes! Let's find out how… (next)

Lab 4 Scenario 1

Segmentation

PRODUCTION
SHARED
DEV
PCI
10.51.0.0/16
Spoke 1
Jump Instance
10.51.0.10
SAP1
10.52.0.0/16
Spoke 2
10.52.0.10
SAP2
10.52.0.0/16
Spoke 3
10.53.0.10
SAP3
10.0.0.0/16
VPC A
EC2-A
10.1.0.0/16
VPC B
EC2-B
Controller VPC
Aviatrix CoPilot
Aviatrix Controller
Transit
10.50.0.0/16
10.2.0.0/16
VPC C (Transit)
EC2-C
AWS us-west-2
AWS us-east-1

VPC-A is a Development VPC

We will isolate any Development VPC from our Production VPCs

A Shared Services VPC should be accessible by both Prod and Dev

There is a PCI VPC that must be isolated from Dev, Prod, and Shared

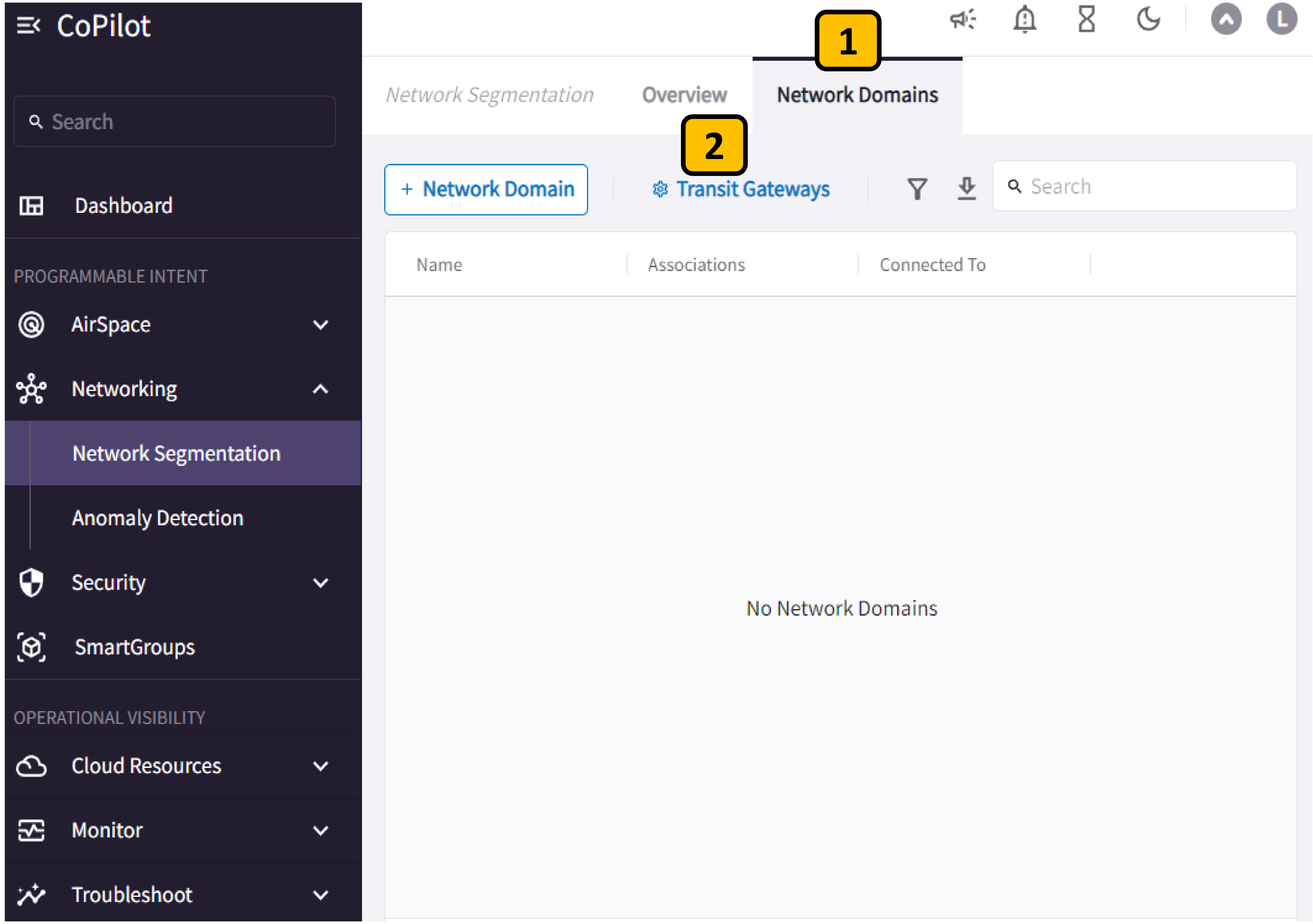
Lab 4: Network Segmentation: Step 4.0

Enable Segmentation

Let’s start by enabling Segmentation on our Aviatrix Transit Gateways

Go to Networking > Network Segmentation > Network Domains 1

Click * **Transit Gateways** 2



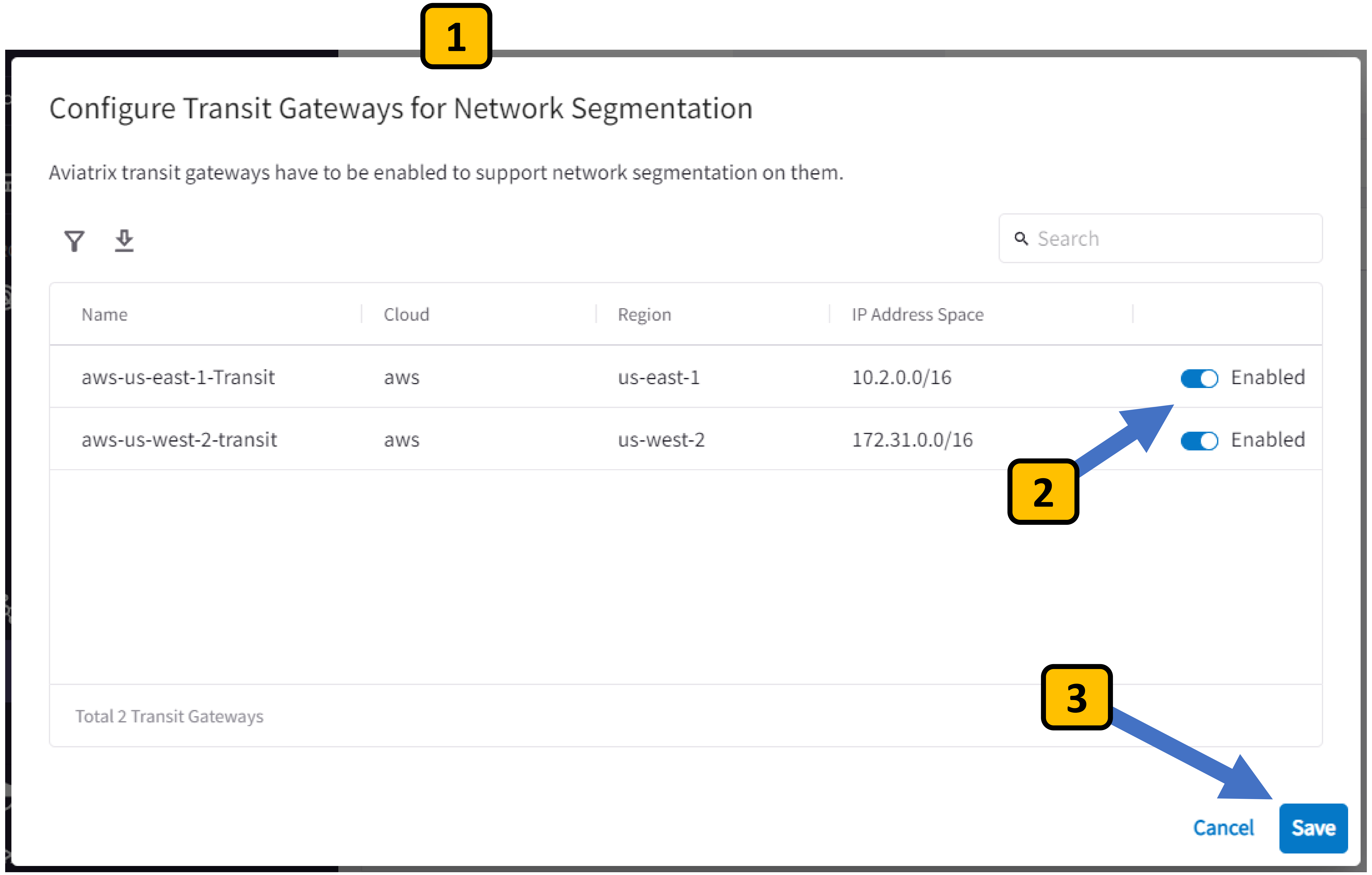
Lab 4: Network Segmentation: Step 4.1

Define the Domains

A pop-up window will appear displaying your Aviatrix Transit Gateways 1

Toggle the switch to **Enabled** for both gateways 2

Click **Save** 3



Configure Transit Gateways for Network Segmentation

Aviatrix transit gateways have to be enabled to support network segmentation on them.

Name	Cloud	Region	IP Address Space	
aws-us-east-1-Transit	aws	us-east-1	10.2.0.0/16	Enabled
aws-us-west-2-transit	aws	us-west-2	172.31.0.0/16	Enabled

Total 2 Transit Gateways

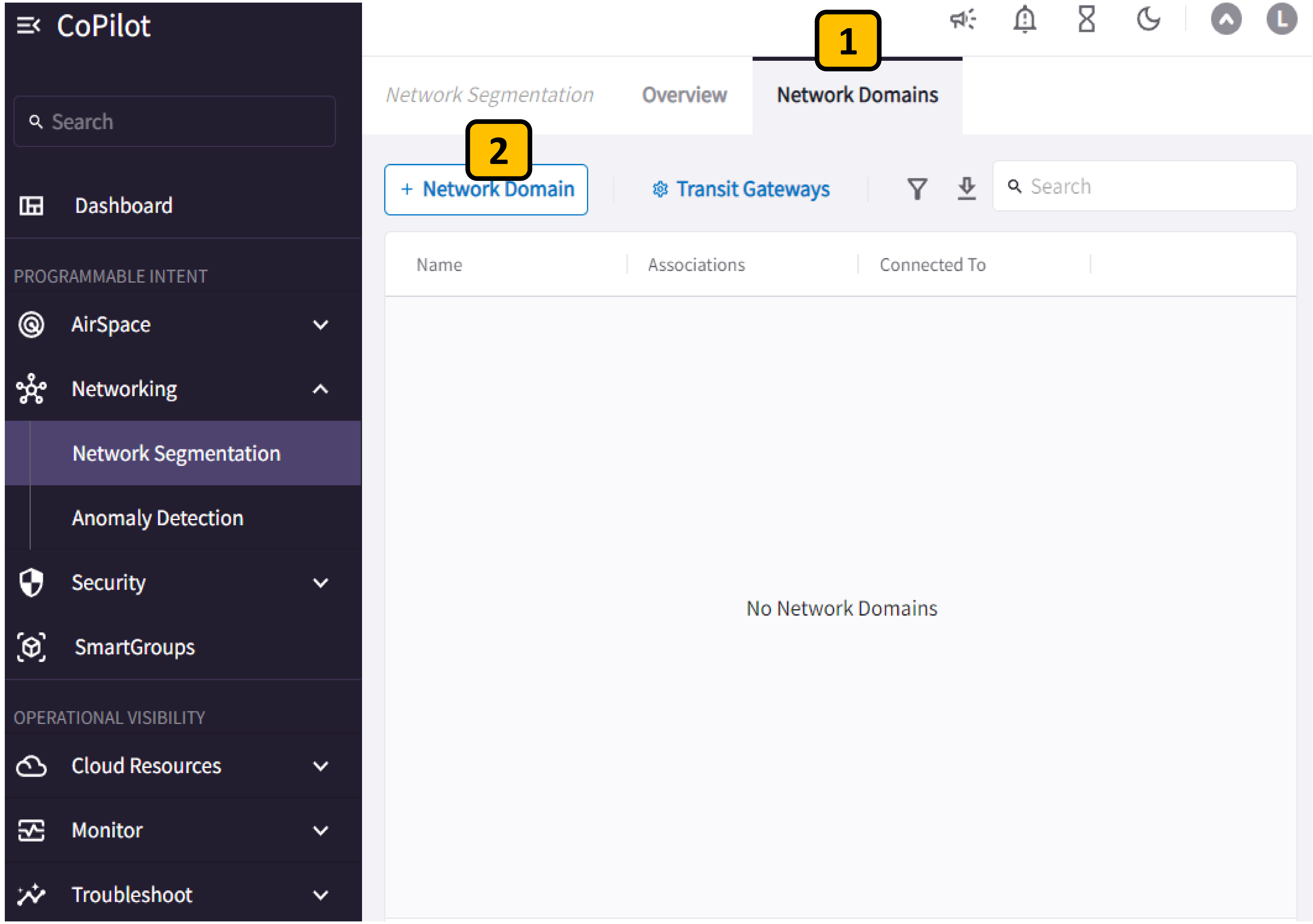
Lab 4: Network Segmentation: Step 4.2

Define the Domains

Now create our four domains: PROD, DEV, SHARED, and PCI

Go to Networking > Network Segmentation > Network Domains 1

Click **+ Network Domain** 2



No Network Domains

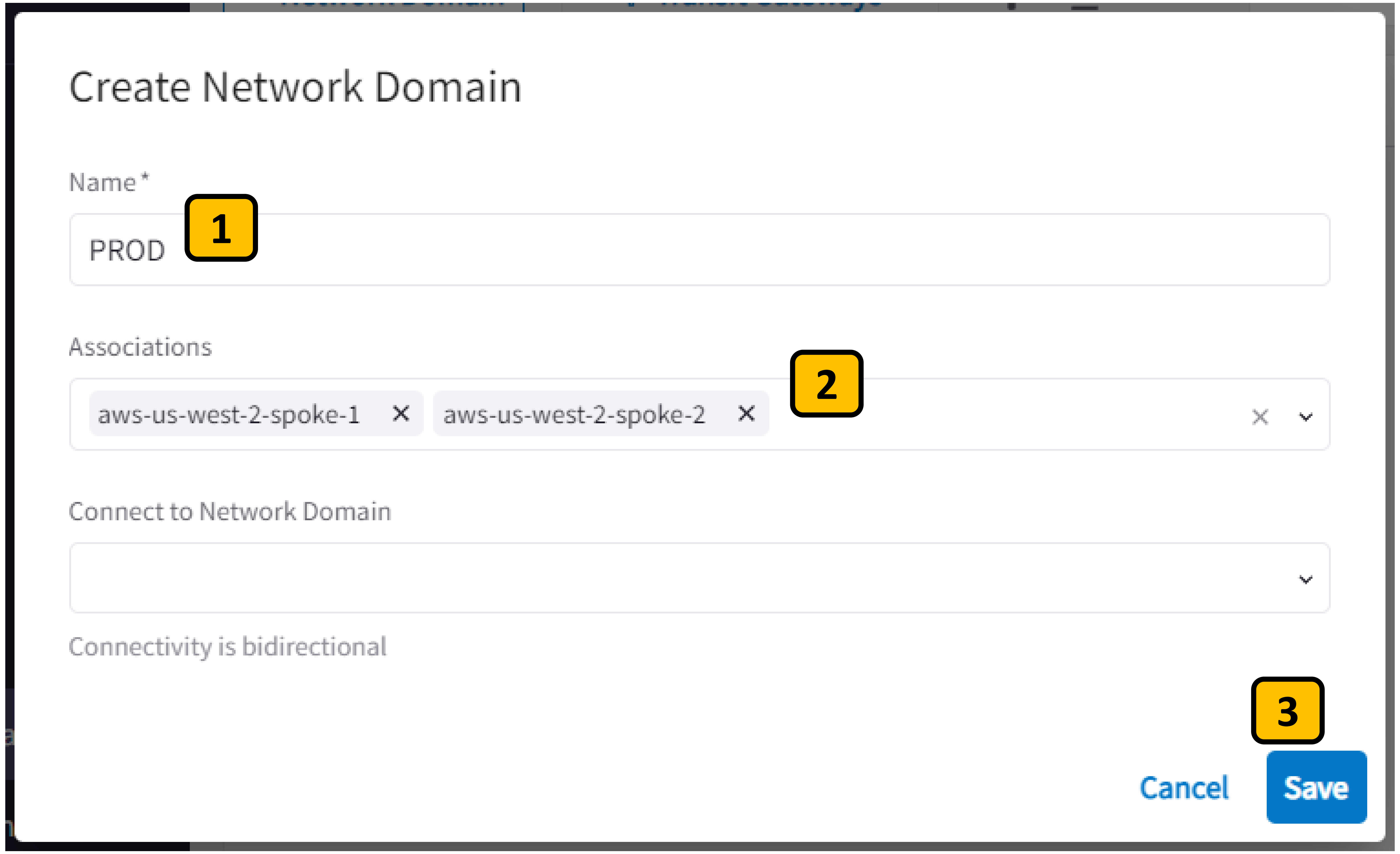
Lab 4: Network Segmentation: Step 4.3

Define the Domains

Name the first domain **PROD** 1

Click the **Associations** drop-down and select the following Spoke gateways to represent this domain:
aws-us-west-2-spoke-1
aws-us-west-2-spoke-2 2

Leave the Connect to Network Domain box alone for now and just click **Save** 3



Create Network Domain

Name *
PROD 1

Associations
aws-us-west-2-spoke-1 ✕ aws-us-west-2-spoke-2 ✕ 2

Connect to Network Domain

Connectivity is bidirectional

3

Lab 4: Network Segmentation: Step 4.5

Define the Domains

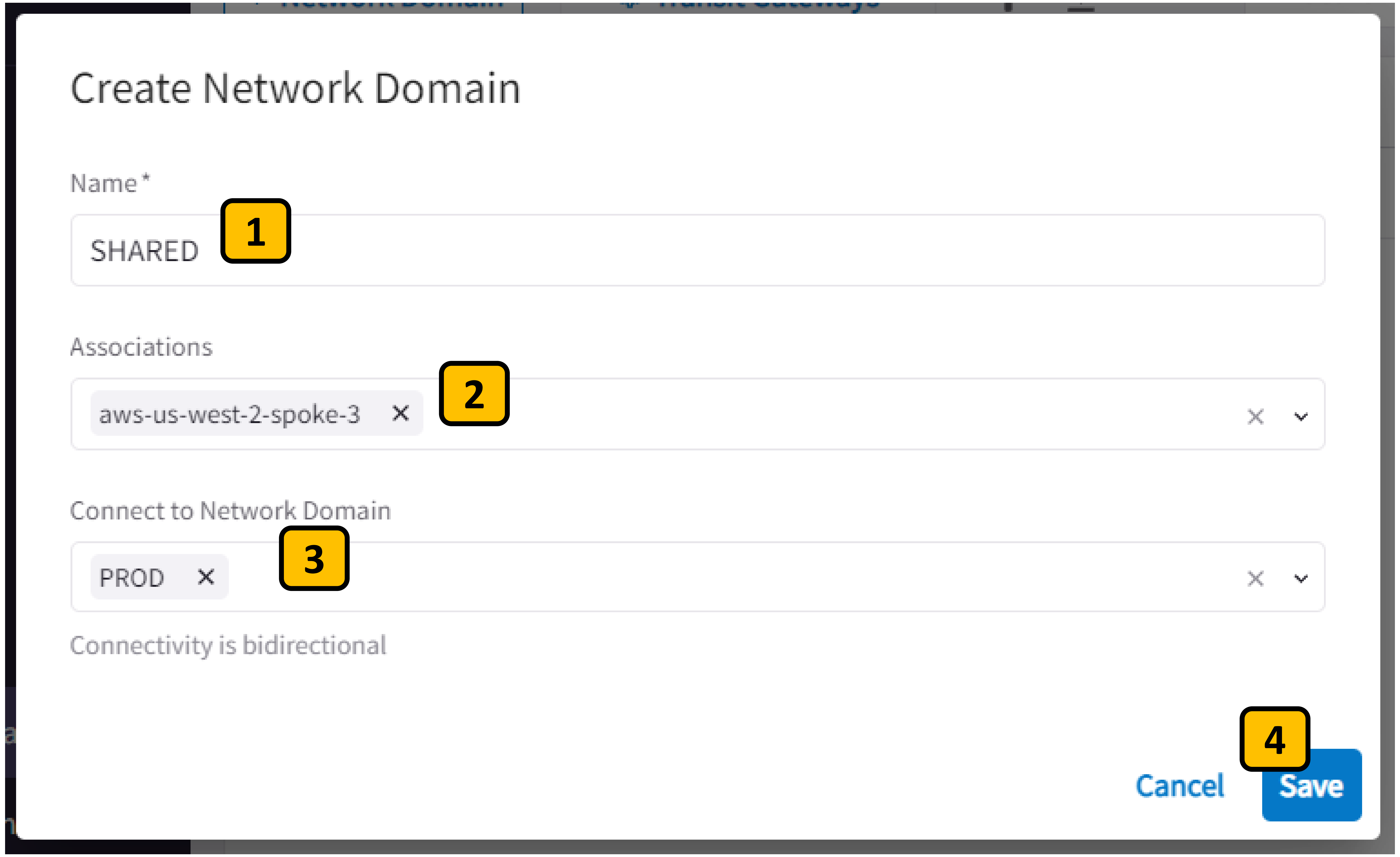
Let's create **SHARED** next 1

Click the **Associations** drop-down and select the following Spoke gateway to represent this domain: **aws-us-west-2-spoke-3** 2

We want PROD to be able to talk to SHARED, so…

In the Connect to Network Domain box select PROD 3

Click **Save** and let's repeat this process for the next domain.. 4



Create Network Domain

Name *

SHARED

Associations

aws-us-west-2-spoke-3 ×

Connect to Network Domain

PROD ×

Connectivity is bidirectional

Lab 4: Network Segmentation: Step 4.6

Define the Domains

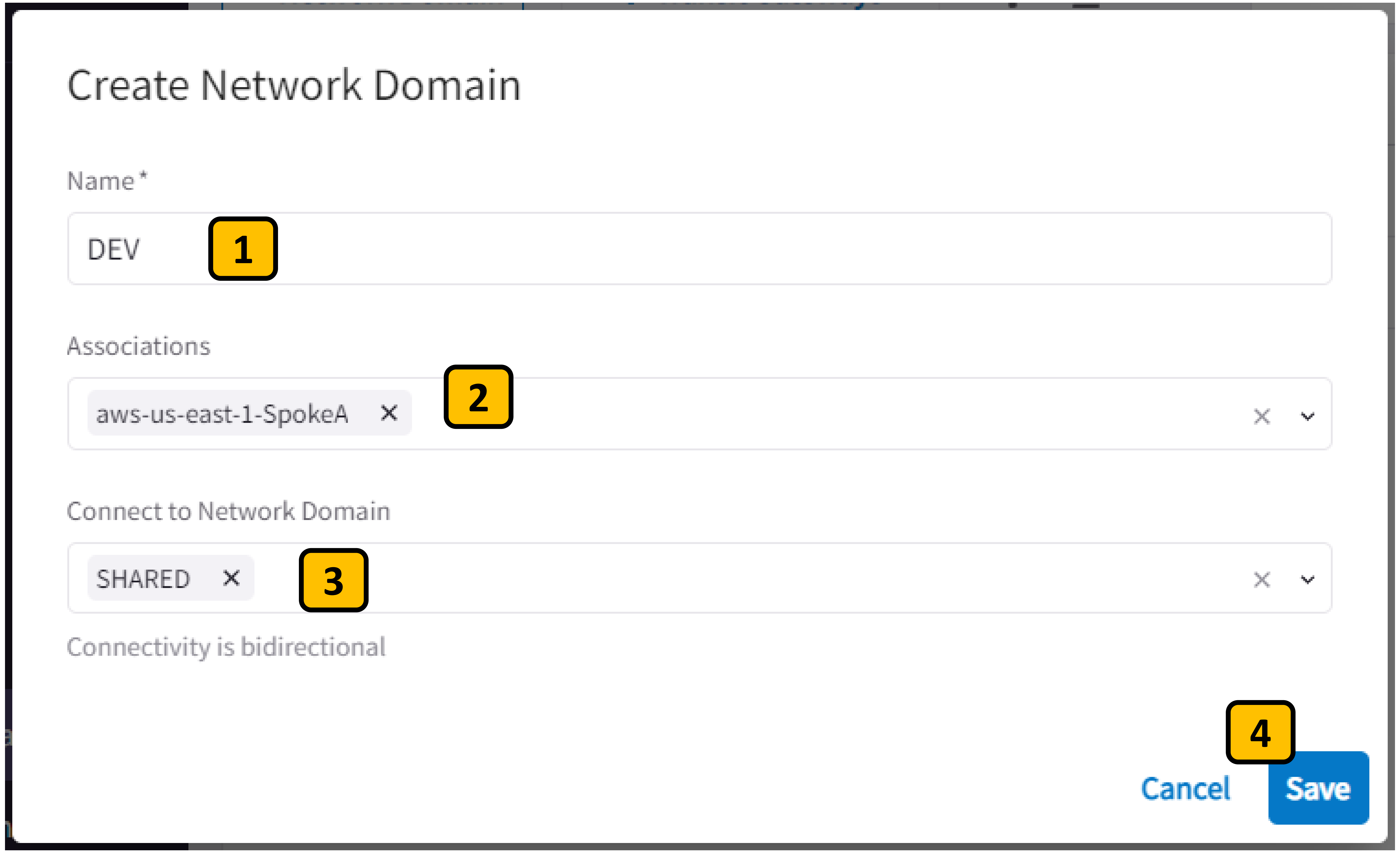
Next let's create **DEV** 1

Click the **Associations** drop-down and select the following Spoke gateway to represent this domain: **aws-us-east-1-SpokeA** 2

We want DEV to be able to talk to SHARED, so…

In the Connect to Network Domain box select SHARED 3

Click **Save** and let's repeat this process for the next domain.. 4



Create Network Domain

Name * : DEV 1

Associations: aws-us-east-1-SpokeA ✕ 2

Connect to Network Domain: SHARED ✕ 3

Connectivity is bidirectional

Lab 4: Network Segmentation: Step 4.7

Define the Domains

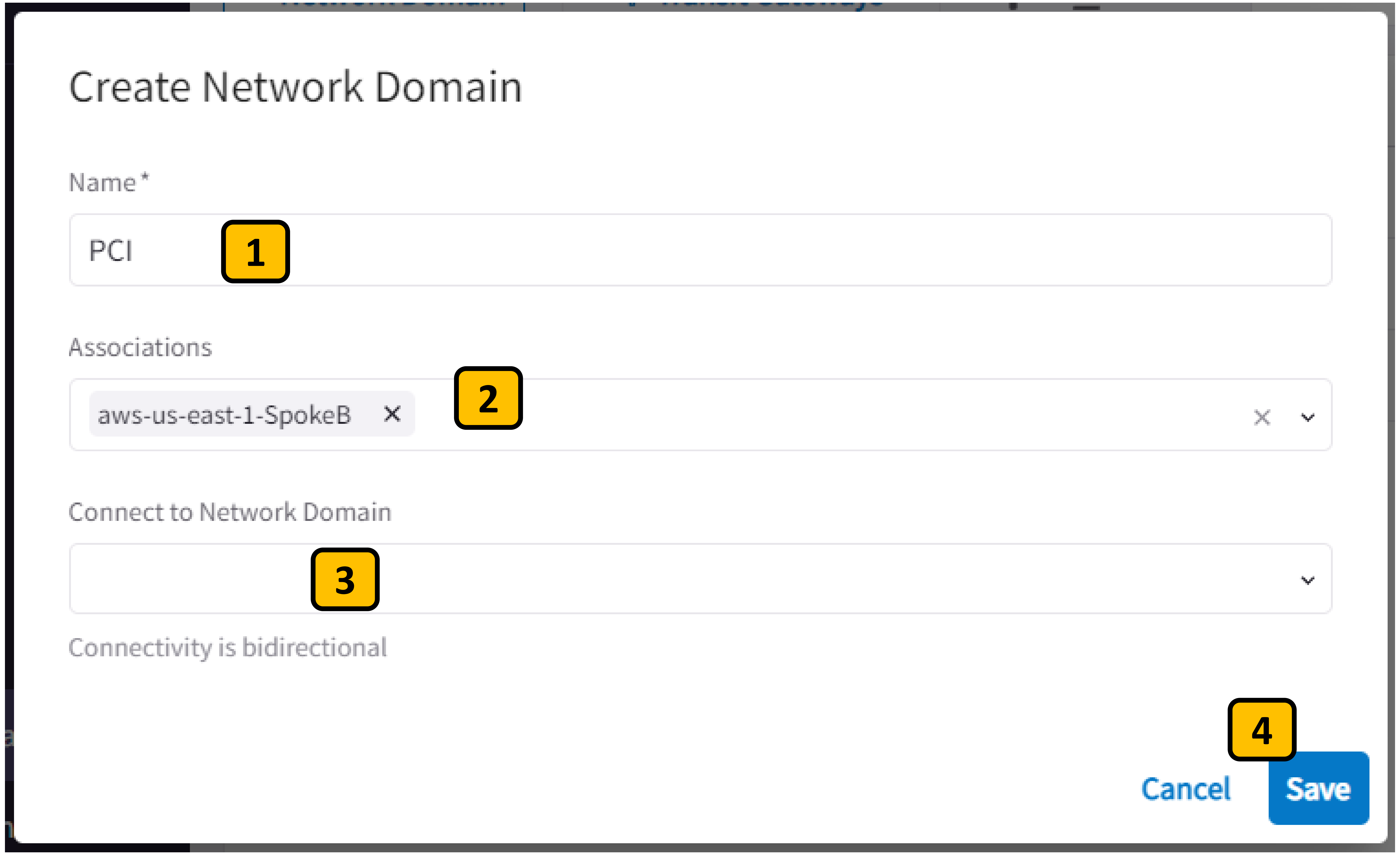
Finally, create **PCI** 1

Click the **Associations** drop-down and select the following Spoke gateway to represent this domain: **aws-us-east-1-SpokeB** 2

We want PCI to be isolated from the other domains, so…

In the Connect to Network Domain box leave it blank 3

Click **Save** 4



Create Network Domain

Name *

PCI 1

Associations

aws-us-east-1-SpokeB × 2

Connect to Network Domain

3

Connectivity is bidirectional

Lab 4: Network Segmentation: Step 4.8

Define the Domains

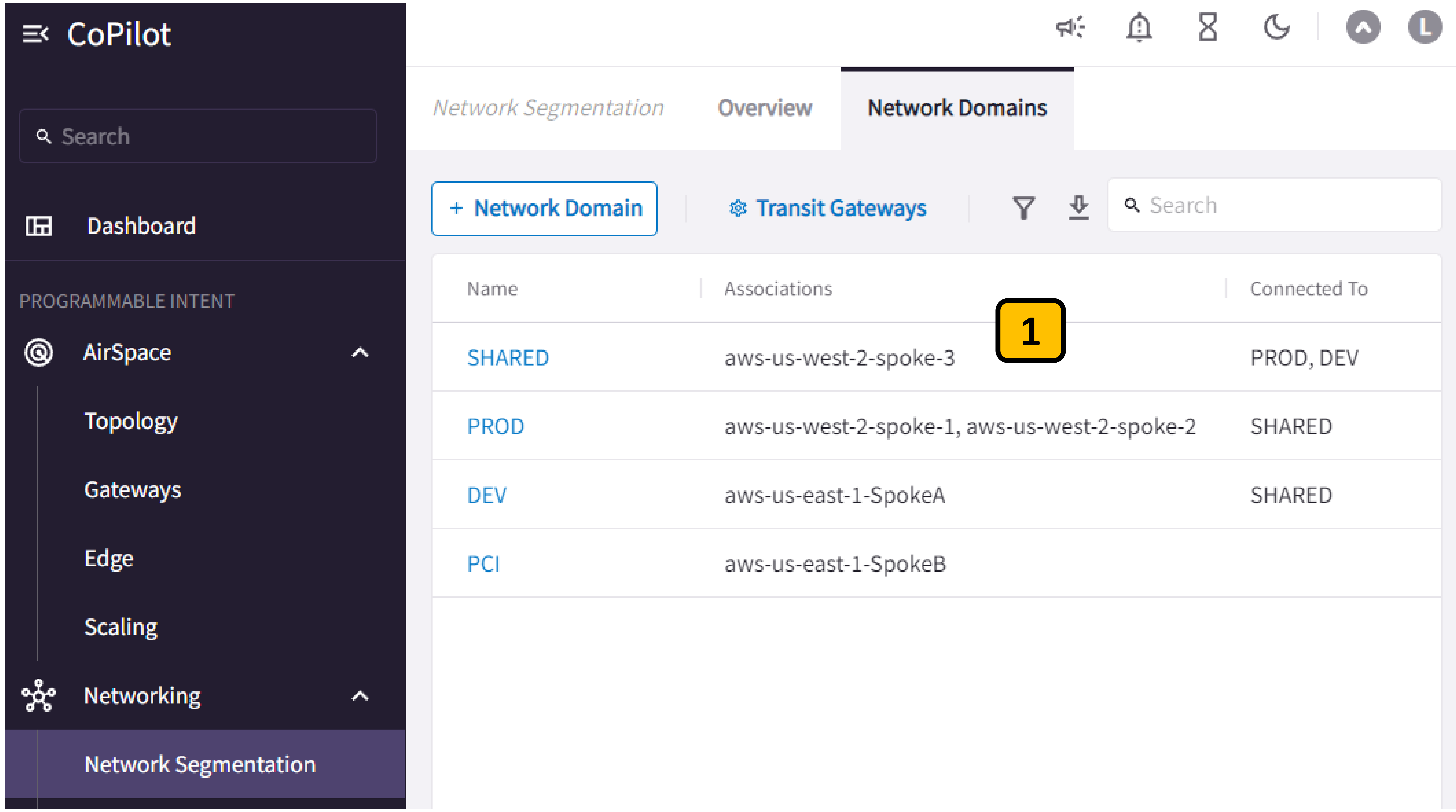
DONE!

You have configured Network Segmentation to meet the requirements of our scenario.

You Network Domains page should look like this… 1

Next, let's see how we can **visualize** our segmentation in CoPilot.

Then we'll test it.



Name	Associations	Connected To
SHARED	aws-us-west-2-spoke-3	PROD, DEV
PROD	aws-us-west-2-spoke-1, aws-us-west-2-spoke-2	SHARED
DEV	aws-us-east-1-SpokeA	SHARED
PCI	aws-us-east-1-SpokeB	

Lab 4: Network Segmentation: Step 4.9

Visualize the Domains

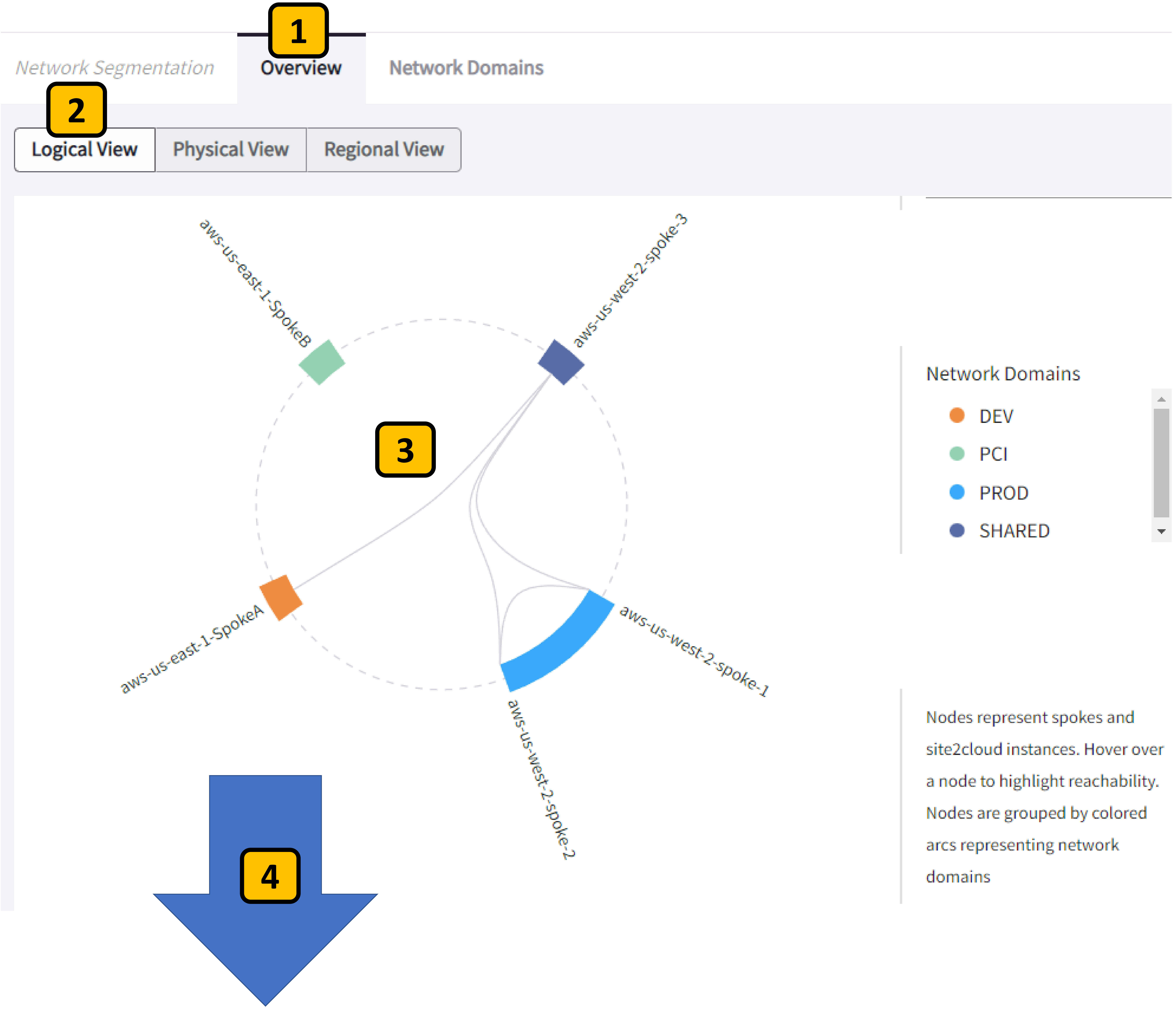
Select the **Overview** tab 1

Select **Logical View** 2

CoPilot has given each domain a color and draws a line connecting two domains if they are allowed to talk. 3

Notice that our PCI domain (colored green here), has no lines to any other domain. That’s what we wanted.

Scroll down to see another view 4



Nodes represent spokes and site2cloud instances. Hover over a node to highlight reachability. Nodes are grouped by colored arcs representing network domains

Lab 4: Network Segmentation: Step 4.11

Visualize the Domains

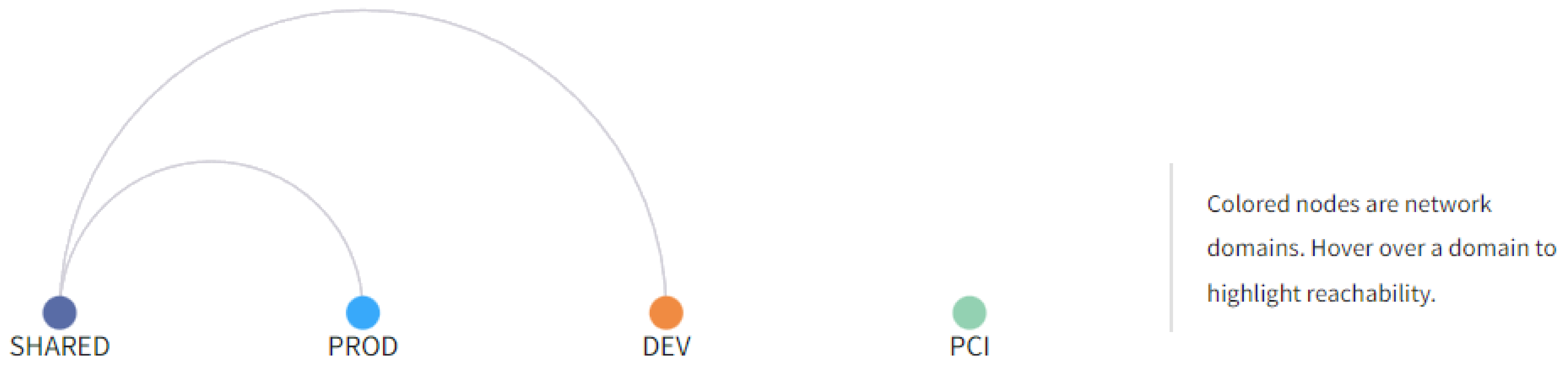
Here is another more simplified view that CoPilot gives us.

We can see that our Shared domain has lines to the DEV and PROD, which means they can talk.

There is no line from PROD to DEV

There is no line from PCI to any other domain.

This can be extremely helpful to show your auditors to prove you are complying to regulations.



Colored nodes are network domains. Hover over a domain to highlight reachability.

Lab 4: Scenario 1: Recap

Segmentation

PRODUCTION
SHARED
DEV
PCI
10.51.0.0/16
Spoke 1
Jump Instance
10.51.0.10
SAP1
10.52.0.0/16
Spoke 2
10.52.0.10
SAP2
10.52.0.0/16
Spoke 3
10.53.0.10
SAP3
10.0.0.0/16
VPC A
EC2-A
10.1.0.0/16
VPC B
EC2-B
Controller VPC
Aviatrix CoPilot
Aviatrix Controller
Trans
10.50.0.0/16
10.2.0.0/16
VPC C (Transit)
EC2-C
AWS us-west-2
AWS us-east-1

This diagram summarizes our allowed traffic flows.

Let's test it!

Lab 4: Network Segmentation: Step 4.12

Access the console of DEV instance EC2-A in us-east-1

From the AWS Console make sure you're in the correct region 1

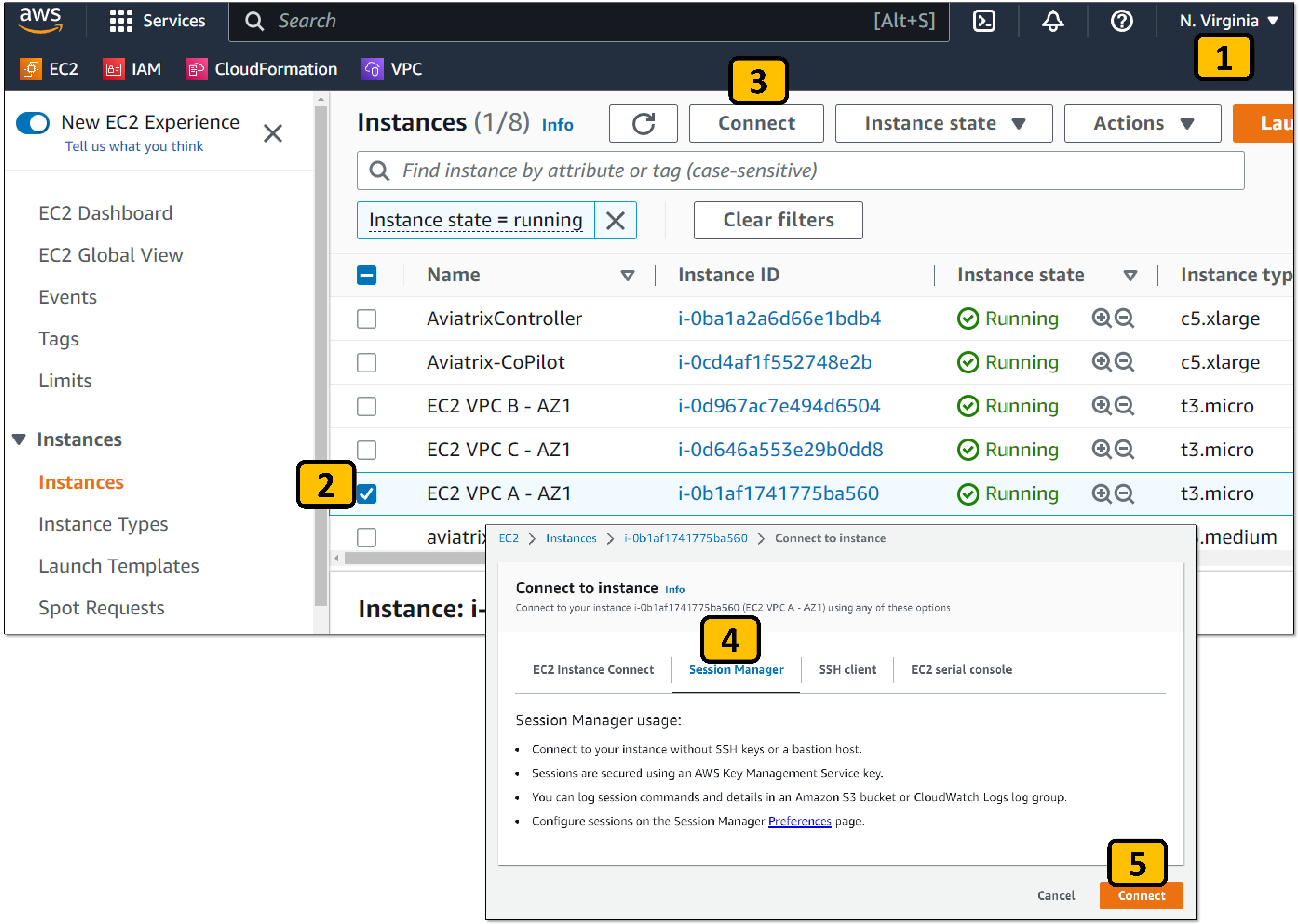
Select the **EC2 VPC A – AZ1** instance 2

Click on **Connect** 3

Select the **Session Manager** tab 4

Click on **Connect** 5

This will open a console on that instance where we can ping… (next)



Instances (1/8) Info

Find instance by attribute or tag (case-sensitive)

Instance state = running ✕

	Name	Instance ID	Instance state	Instance typ
□	AviatrixController	i-0ba1a2a6d66e1bdb4	✓ Running	c5.xlarge
□	Aviatrix-CoPilot	i-0cd4af1f552748e2b	✓ Running	c5.xlarge
□	EC2 VPC B - AZ1	i-0d967ac7e494d6504	✓ Running	t3.micro
□	EC2 VPC C - AZ1	i-0d646a553e29b0dd8	✓ Running	t3.micro
☑	EC2 VPC A - AZ1	i-0b1af1741775ba560	✓ Running	t3.micro
□	aviatrix			.medium

Instance: i-

EC2 > Instances > i-0b1af1741775ba560 > Connect to instance

Connect to instance Info

Connect to your instance i-0b1af1741775ba560 (EC2 VPC A - AZ1) using any of these options

EC2 Instance Connect | Session Manager | 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.

Lab 4: Network Segmentation: Step 4.13

Ping test from DEV instance EC2-A to SHARED instance SAP3

From the instance console type:

sudo su –l ec2-user 1

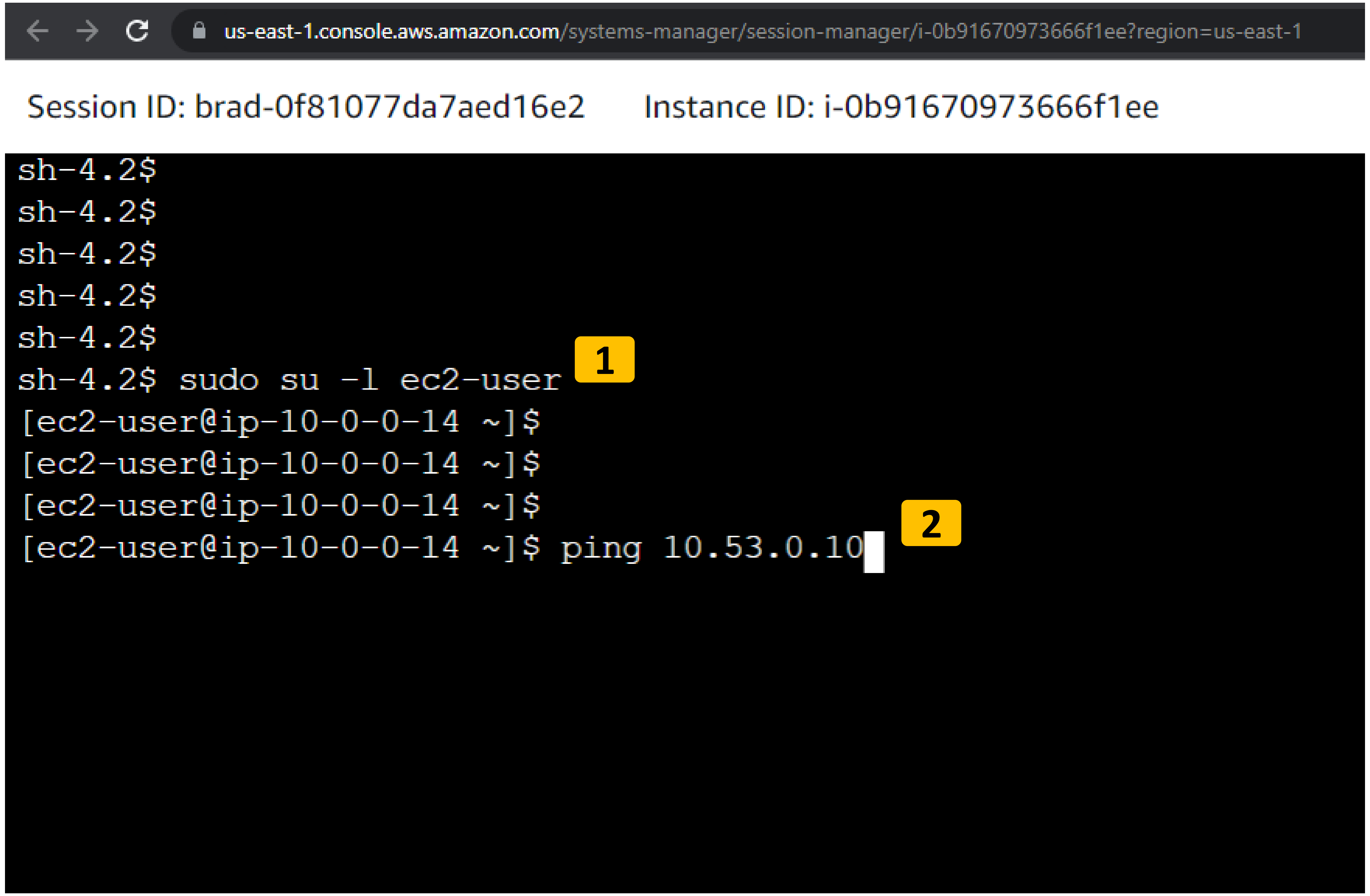
(that's a dash lowercase L)

EC2-A is in the DEV domain, and its going to ping instance SAP3 in the SHARED domain.

We've allowed those domains to talk so we expect this ping to work.

Enter the command:

ping 10.53.0.10 2


us-east-1.console.aws.amazon.com/systems-manager/session-manager/i-0b91670973666f1ee?region=us-east-1

Session ID: brad-0f81077da7aed16e2   Instance ID: i-0b91670973666f1ee

```
sh-4.2$
sh-4.2$
sh-4.2$
sh-4.2$
sh-4.2$
sh-4.2$ sudo su -l ec2-user
[ec2-user@ip-10-0-0-14 ~]$
[ec2-user@ip-10-0-0-14 ~]$
[ec2-user@ip-10-0-0-14 ~]$
[ec2-user@ip-10-0-0-14 ~]$ ping 10.53.0.10
```

Lab 4: Network Segmentation: Step 4.14

Test connectivity

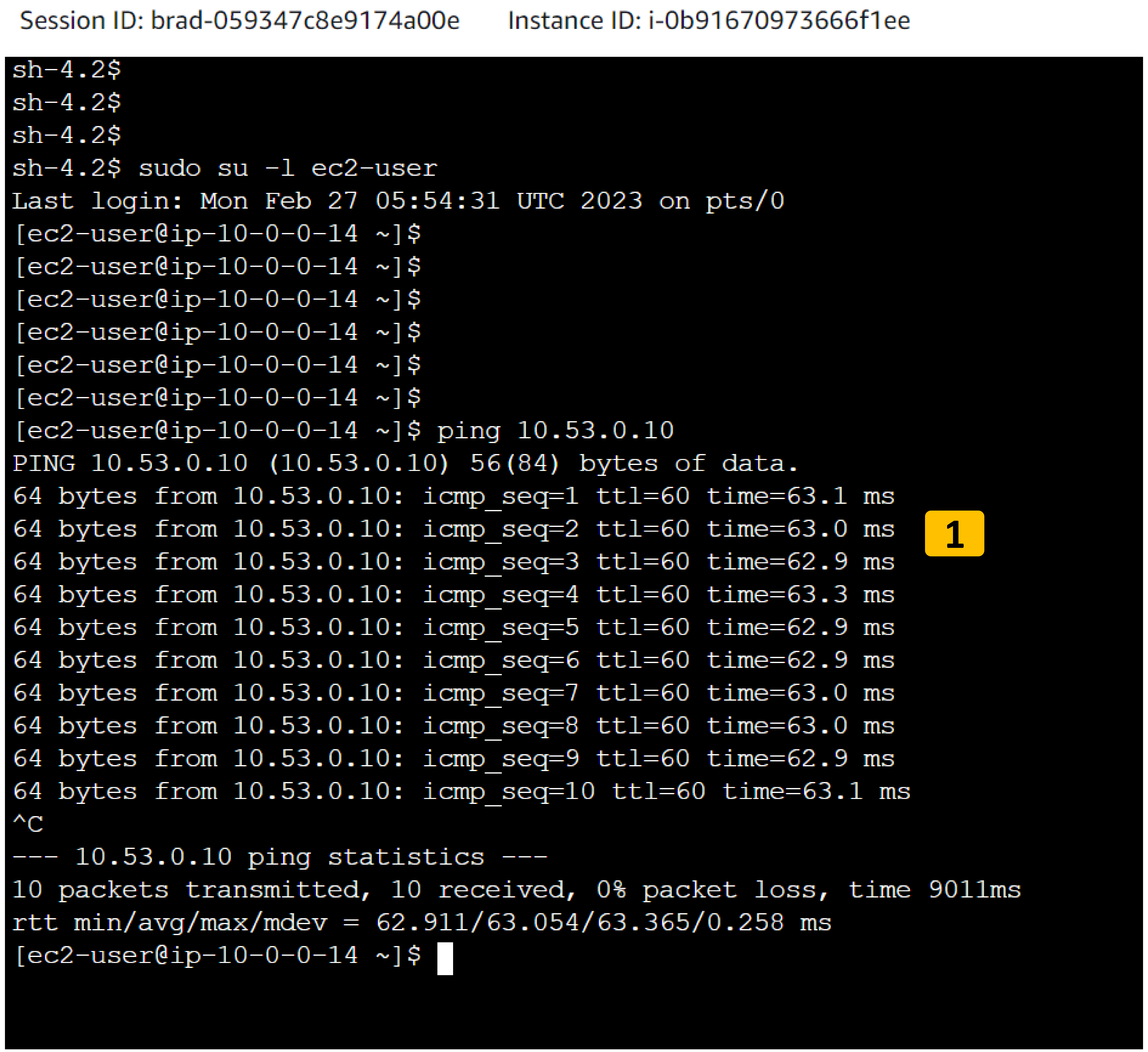
The ping works as we expected 1

So far so good…

Now let’s try to ping from DEV to PROD

We expect that to NOT work…

Let’s see…. (next)



Session ID: brad-059347c8e9174a00e   Instance ID: i-0b91670973666f1ee

```
sh-4.2$
sh-4.2$
sh-4.2$
sh-4.2$ sudo su -l ec2-user
Last login: Mon Feb 27 05:54:31 UTC 2023 on pts/0
[ec2-user@ip-10-0-0-14 ~]$
[ec2-user@ip-10-0-0-14 ~]$
[ec2-user@ip-10-0-0-14 ~]$
[ec2-user@ip-10-0-0-14 ~]$
[ec2-user@ip-10-0-0-14 ~]$
[ec2-user@ip-10-0-0-14 ~]$
[ec2-user@ip-10-0-0-14 ~]$ ping 10.53.0.10
PING 10.53.0.10 (10.53.0.10) 56(84) bytes of data.
64 bytes from 10.53.0.10: icmp_seq=1 ttl=60 time=63.1 ms
64 bytes from 10.53.0.10: icmp_seq=2 ttl=60 time=63.0 ms
64 bytes from 10.53.0.10: icmp_seq=3 ttl=60 time=62.9 ms
64 bytes from 10.53.0.10: icmp_seq=4 ttl=60 time=63.3 ms
64 bytes from 10.53.0.10: icmp_seq=5 ttl=60 time=62.9 ms
64 bytes from 10.53.0.10: icmp_seq=6 ttl=60 time=62.9 ms
64 bytes from 10.53.0.10: icmp_seq=7 ttl=60 time=63.0 ms
64 bytes from 10.53.0.10: icmp_seq=8 ttl=60 time=63.0 ms
64 bytes from 10.53.0.10: icmp_seq=9 ttl=60 time=62.9 ms
64 bytes from 10.53.0.10: icmp_seq=10 ttl=60 time=63.1 ms
^C
--- 10.53.0.10 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9011ms
rtt min/avg/max/mdev = 62.911/63.054/63.365/0.258 ms
[ec2-user@ip-10-0-0-14 ~]$
```

Lab 4: Network Segmentation: Step 4.15

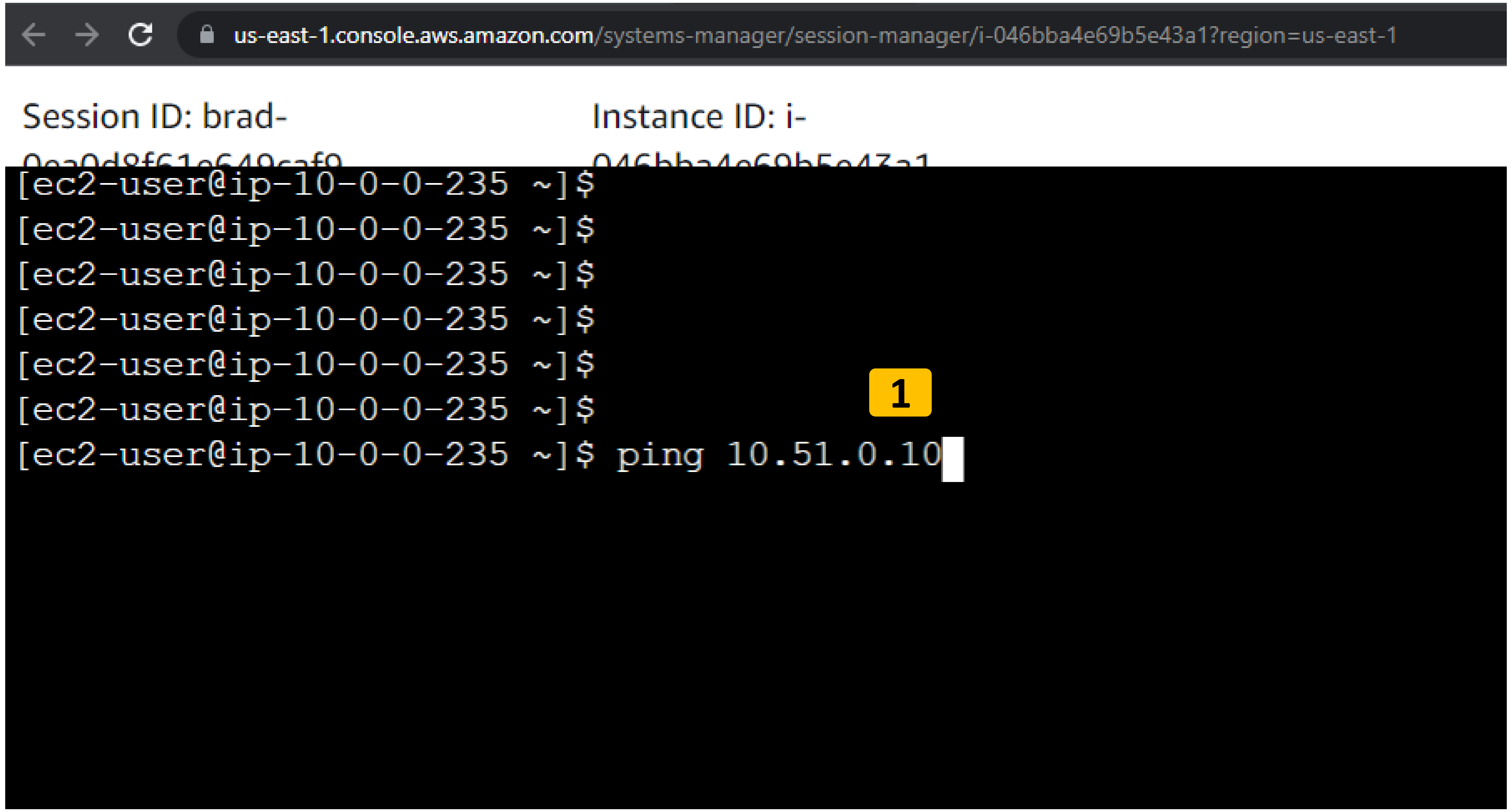
Ping test from DEV instance EC2-A to PROD instance SAP1

From the same EC2-A instance console:

EC2-A is in the DEV domain, and its going to ping instance SAP1 in the PROD domain.

We did NOT allow those domains to talk so we expect this ping will NOT work.

Enter the command:
ping 10.51.0.10 1



us-east-1.console.aws.amazon.com/systems-manager/session-manager/i-046bba4e69b5e43a1?region=us-east-1

Session ID: brad-0ea0d8f61e649caf9

Instance ID: i-046bba4e69b5e43a1

```
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$ ping 10.51.0.10
```

Lab 4: Network Segmentation: Step 4.16

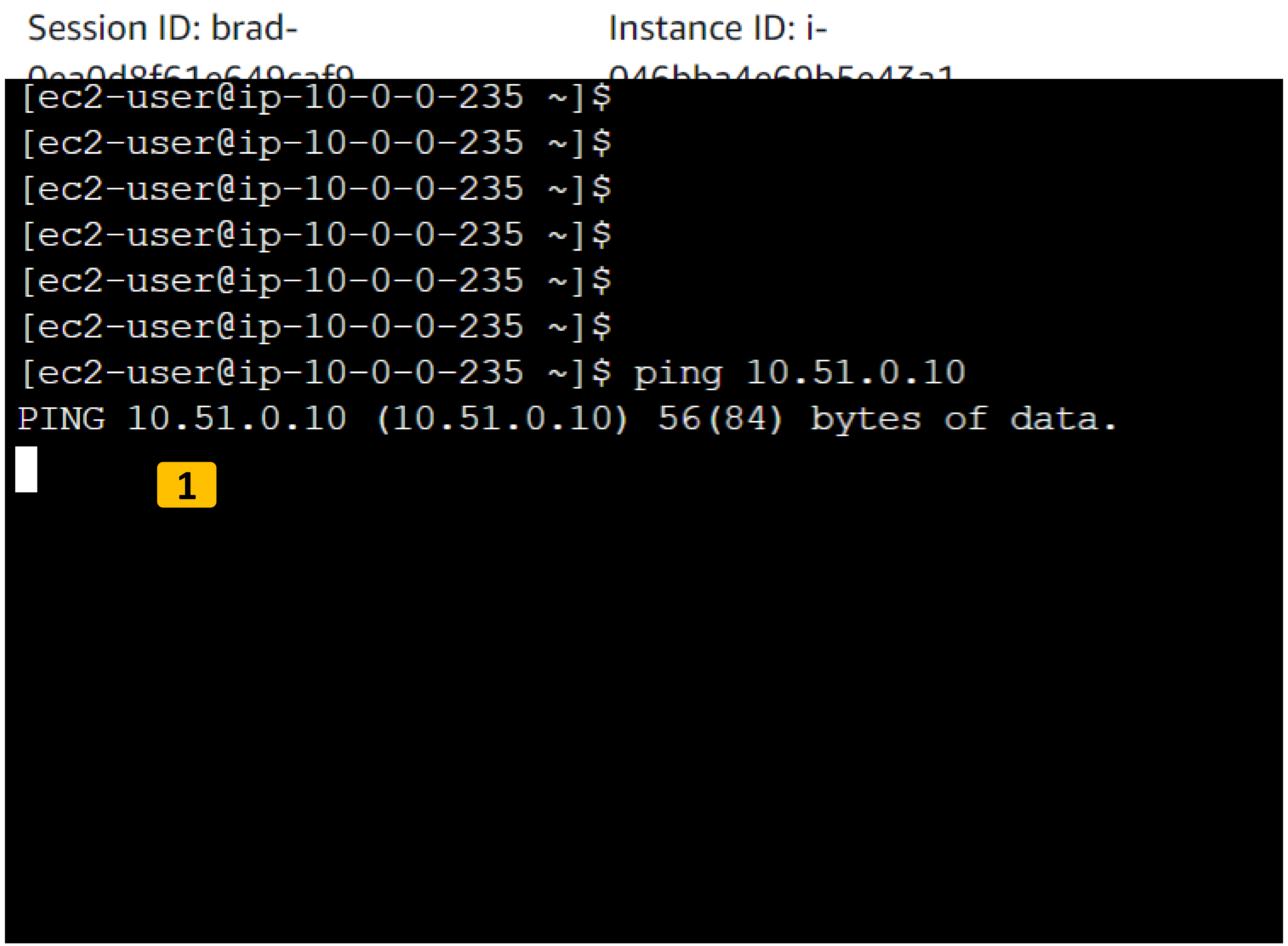
Ping test from DEV instance EC2-A to PROD instance SAP1

SUCCESS! The ping did **NOT** work. This is what we wanted to happen. 1

We have isolated our DEV from PROD with Network Segmentation.

Optional:

If you don’t believe me, go back and remove your Network Segmentation config and see if the ping works…



Session ID: brad-
0ea0d8f61e649caf9

Instance ID: i-
046bba4e69b5e43a1

```
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$
[ec2-user@ip-10-0-0-235 ~]$ ping 10.51.0.10
PING 10.51.0.10 (10.51.0.10) 56(84) bytes of data.
```

Lab 4: Scenario 1: Recap

Segmentation

PRODUCTION
SHARED
DEV
PCI
10.51.0.0/16
Spoke 1
Jump Instance
10.51.0.10
SAP1
10.52.0.0/16
Spoke 2
10.52.0.10
SAP2
10.52.0.0/16
Spoke 3
10.53.0.10
SAP3
10.0.0.0/16
VPC A
EC2-A
10.1.0.0/16
VPC B
EC2-B
Controller VPC
Aviatrix CoPilot
Aviatrix Controller
Trans
10.50.0.0/16
10.2.0.0/16
VPC C (Transit)
EC2-C
AWS us-west-2
AWS us-east-1

Now that you know how to access an instance console and ping. Feel free to test out this scenario more on your own…

Lab 4: Network Segmentation: EXTRA CREDIT

Can you figure out how it works?

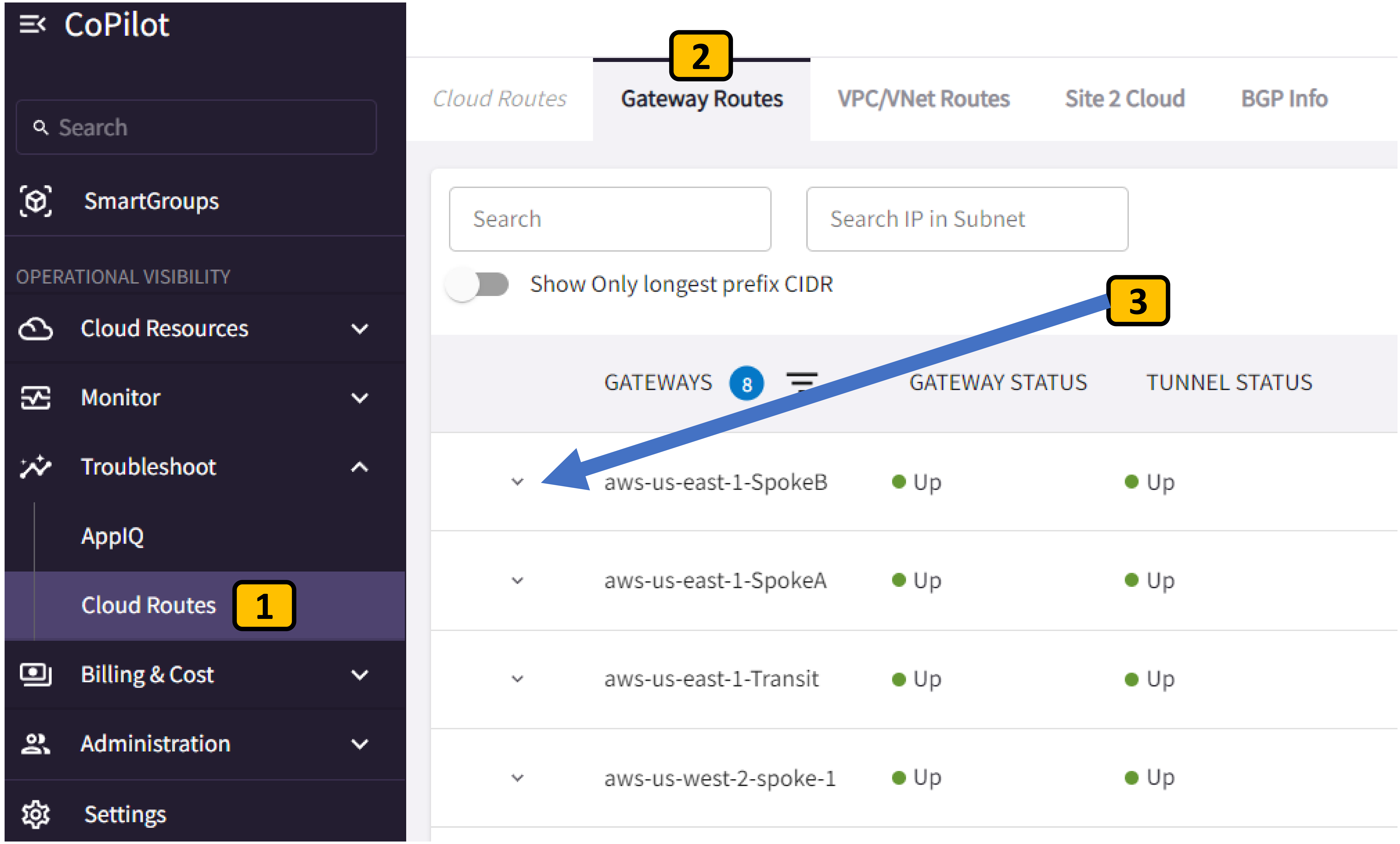
If you really want to know how Segmentation works..

Go to Troubleshoot > Cloud Routes 1

Then Gateway Routes 2

Examine the route tables of the Aviatrix Spoke Gateways 3

See if you can figure out how Segmentation is being enforced :D



GATEWAYS 8	GATEWAY STATUS	TUNNEL STATUS
aws-us-east-1-SpokeB	● Up	● Up
aws-us-east-1-SpokeA	● Up	● Up
aws-us-east-1-Transit	● Up	● Up
aws-us-west-2-spoke-1	● Up	● Up

Lab 4: Scenario 1: Completed

Segmentation

PRODUCTION
SHARED
DEV
PCI
10.51.0.0/16
Spoke 1
Jump Instance
10.51.0.10
SAP1
10.52.0.0/16
Spoke 2
10.52.0.10
SAP2
10.52.0.0/16
Spoke 3
10.53.0.10
SAP3
10.0.0.0/16
VPC A
EC2-A
10.1.0.0/16
VPC B
EC2-B
Controller VPC
Aviatrix CoPilot
Aviatrix Controller
Trans
10.50.0.0/16
10.2.0.0/16
VPC C (Transit)
EC2-C
AWS us-west-2
AWS us-east-1

This concludes Lab 4
Super easy, right?