AWS Immersion Day
LAB 1

AWS NETWORKING FUNDAMENTALS

Aviatrix Systems
Systems Engineering

www.aviatrix.com

aviatrix

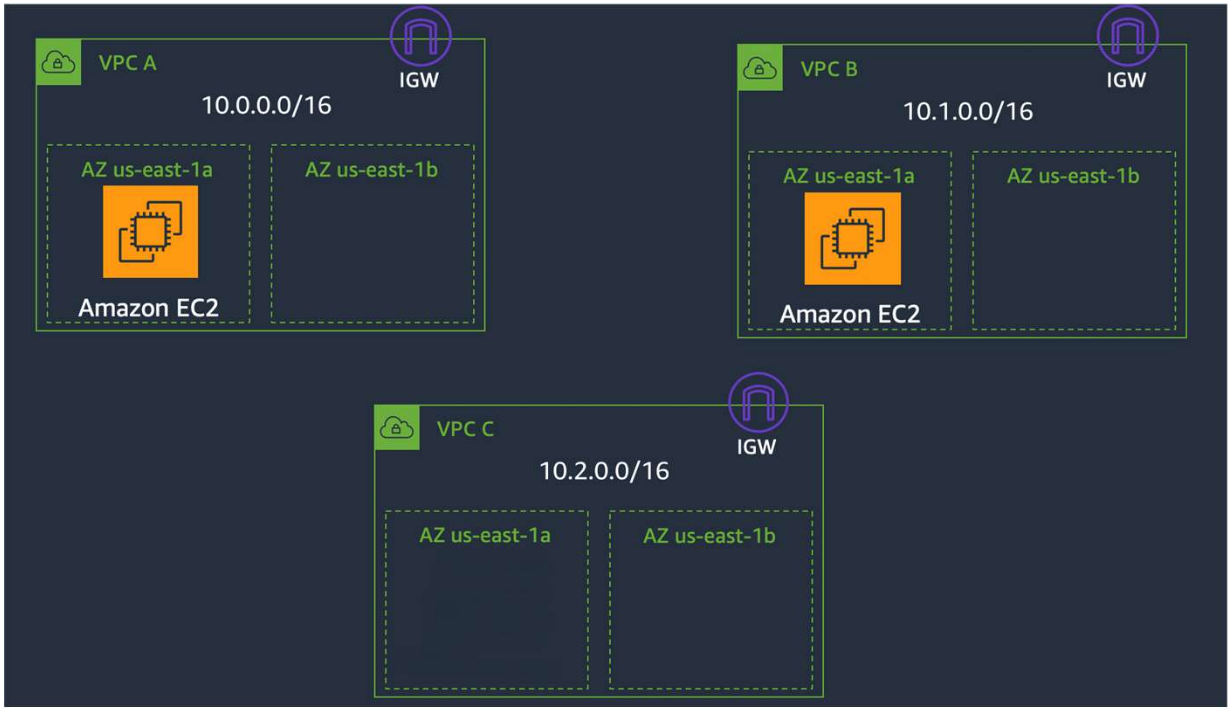
Lab 1: AWS Networking Fundamentals

Create VPCs, IGWs, and EC2 instances

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways. You can use both IPv4 and IPv6 in your VPC for secure and easy access to resources and applications.

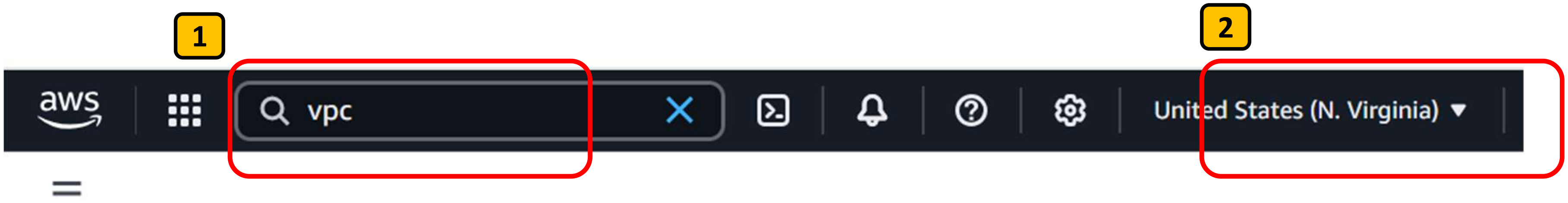
An internet gateway (IGW) is a horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the internet. It therefore imposes no availability risks or bandwidth constraints on your network traffic. In this lab, we will create three VPC's with Internet Gateways.

The lab modules build upon each other. Be sure to follow each step completely, build out in the specified region, and take note of IP addresses and CIDRs to ensure that future lab modules will work correctly.



Lab 1: Step 1.1

Switch to US-EAST-1 region



vpc

United States (N. Virginia) ▼

1 Switch to **VPC** in the console.

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

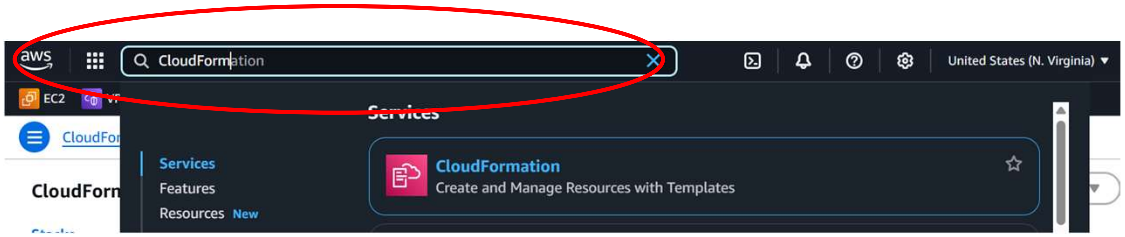
aviatrix aws

Lab 1: Step 1.2

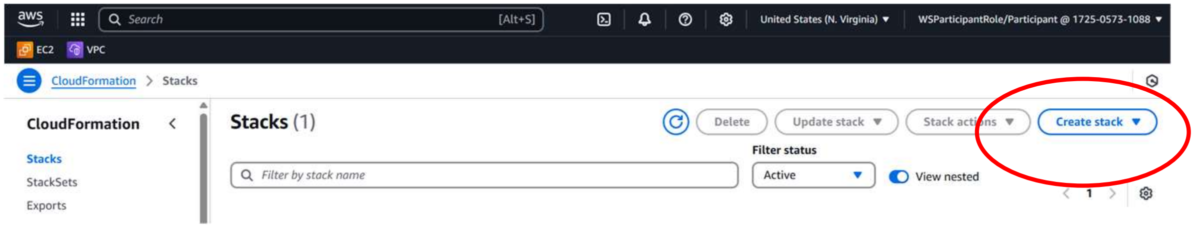
Create the stack

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

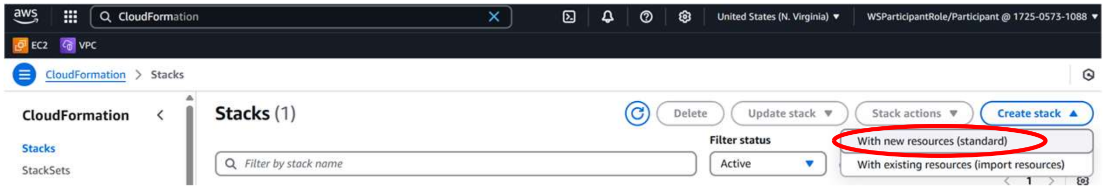
From the AWS Console go to the **CloudFormation** service. 1



Select **Create stack** 2



Select **With new resources (standard)** 3



aviatrix aws

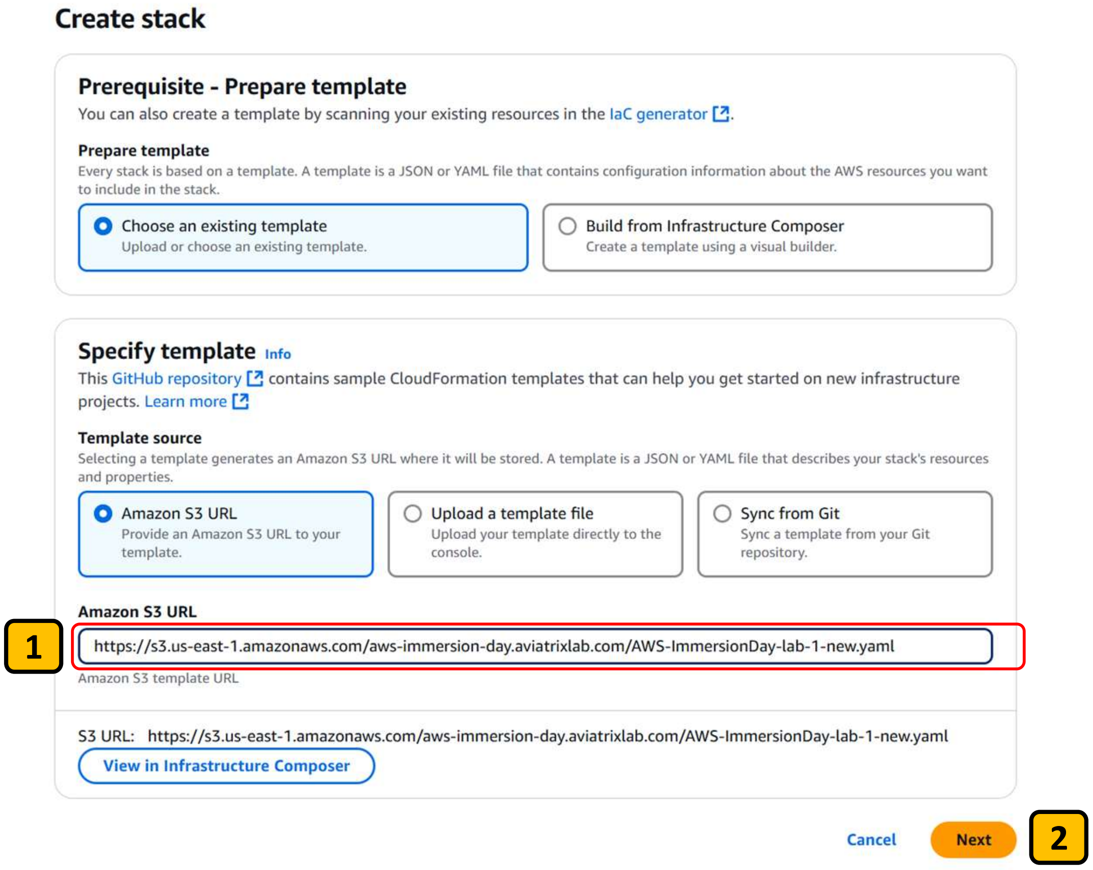
Lab 1: Step 1.3

Provide the stack URL

Keep the default settings of **Template is ready**, and **Amazon S3 URL**.

Enter the URL below in the Amazon S3 URL input field. 1

Select **Next** 2



Create stack

Prerequisite - Prepare template

You can also create a template by scanning your existing resources in the IaC generator.

Prepare template

Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

● Choose an existing template
Upload or choose an existing template.

○ Build from Infrastructure Composer
Create a template using a visual builder.

Specify template Info

This GitHub repository contains sample CloudFormation templates that can help you get started on new infrastructure projects. Learn more

Template source

Selecting a template generates an Amazon S3 URL where it will be stored. A template is a JSON or YAML file that describes your stack's resources and properties.

● Amazon S3 URL
Provide an Amazon S3 URL to your template.

○ Upload a template file
Upload your template directly to the console.

○ Sync from Git
Sync a template from your Git repository.

Amazon S3 URL

1 https://s3.us-east-1.amazonaws.com/aws-immersion-day.aviatrixlab.com/AWS-ImmersionDay-lab-1-new.yaml

Amazon S3 template URL

S3 URL: https://s3.us-east-1.amazonaws.com/aws-immersion-day.aviatrixlab.com/AWS-ImmersionDay-lab-1-new.yaml

1

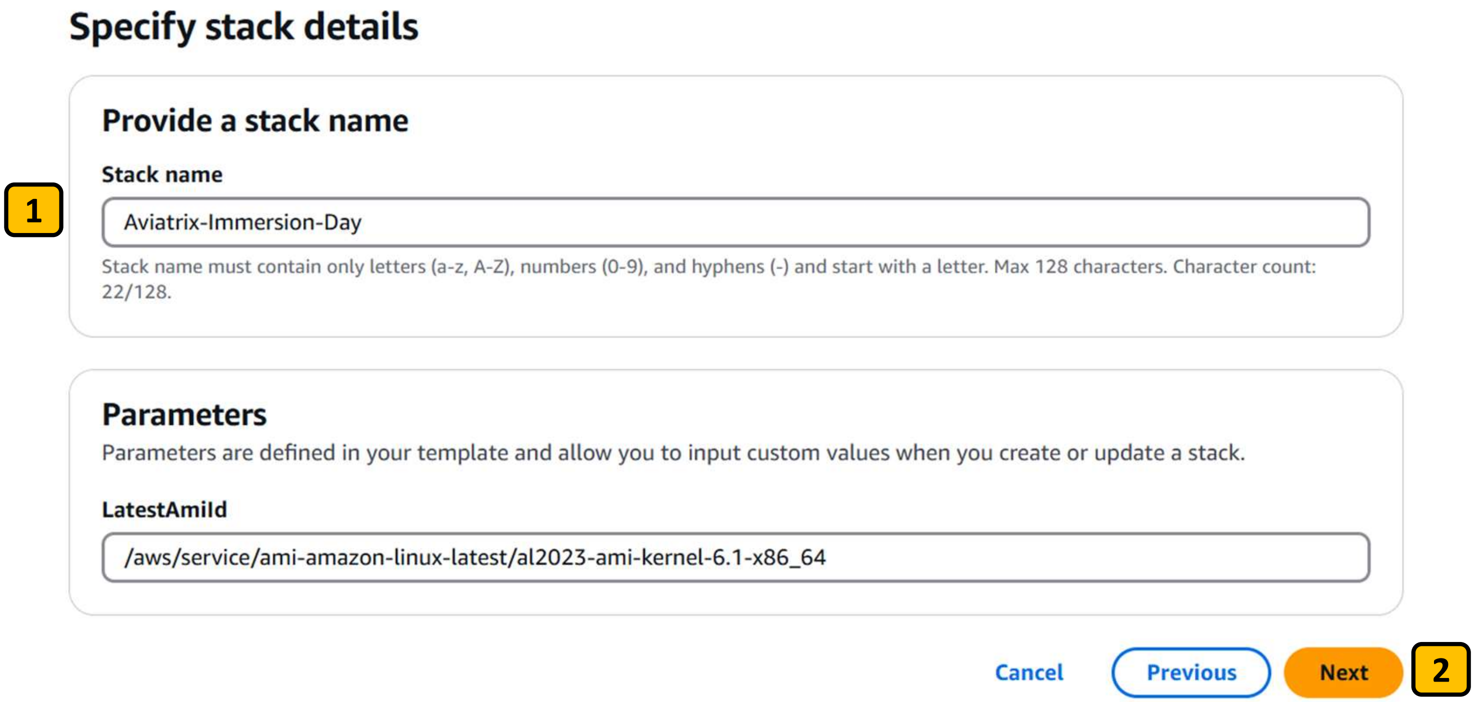
```
https://s3.us-east-1.amazonaws.com/aws-immersion-day.aviatrixlab.com/AWS-ImmersionDay-lab-1-new.yaml
```

Lab 1: Step 1.4

Name the stack

Name the stack
Aviatrix-Immersion-Day 1

Select **Next** 2



Specify stack details

Provide a stack name

1

Stack name

Aviatrix-Immersion-Day

Stack name must contain only letters (a-z, A-Z), numbers (0-9), and hyphens (-) and start with a letter. Max 128 characters. Character count: 22/128.

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

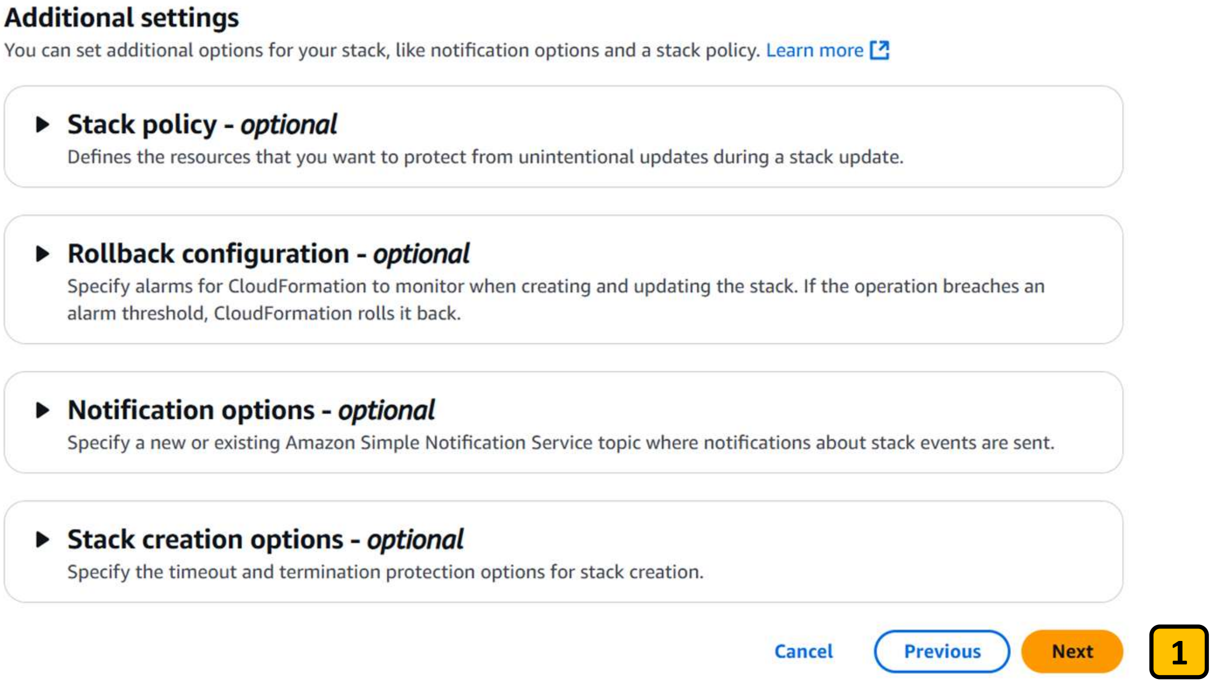
LatestAmiId

/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-6.1-x86_64

Lab 1: Step 1.5

On the next **Configure Stack Options** page just scroll down to the bottom of the page

Then click **Next** 1



Additional settings

You can set additional options for your stack, like notification options and a stack policy. Learn more

▶ **Stack policy - *optional***

Defines the resources that you want to protect from unintentional updates during a stack update.

▶ **Rollback configuration - *optional***

Specify alarms for CloudFormation to monitor when creating and updating the stack. If the operation breaches an alarm threshold, CloudFormation rolls it back.

▶ **Notification options - *optional***

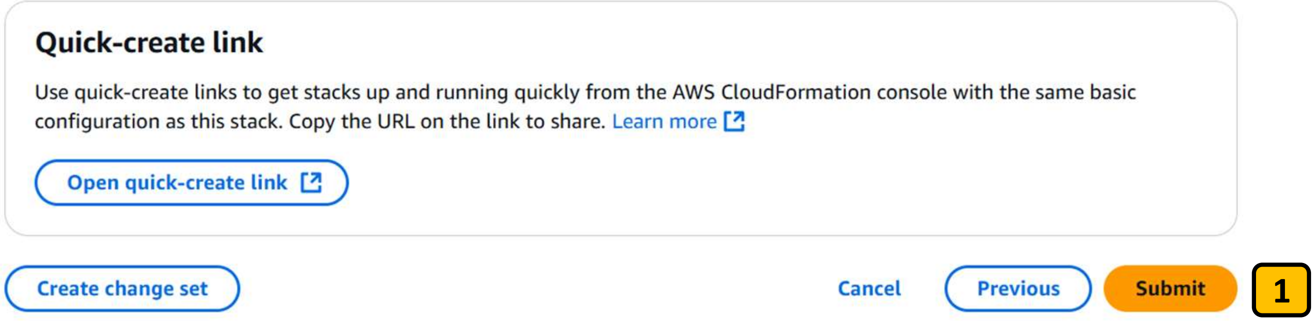
Specify a new or existing Amazon Simple Notification Service topic where notifications about stack events are sent.

▶ **Stack creation options - *optional***

Specify the timeout and termination protection options for stack creation.

Lab 1: Step 1.6

On the **Review and create** page just scroll down to the bottom of the page and click **Submit** 1



Quick-create link

Use quick-create links to get stacks up and running quickly from the AWS CloudFormation console with the same basic configuration as this stack. Copy the URL on the link to share. Learn more

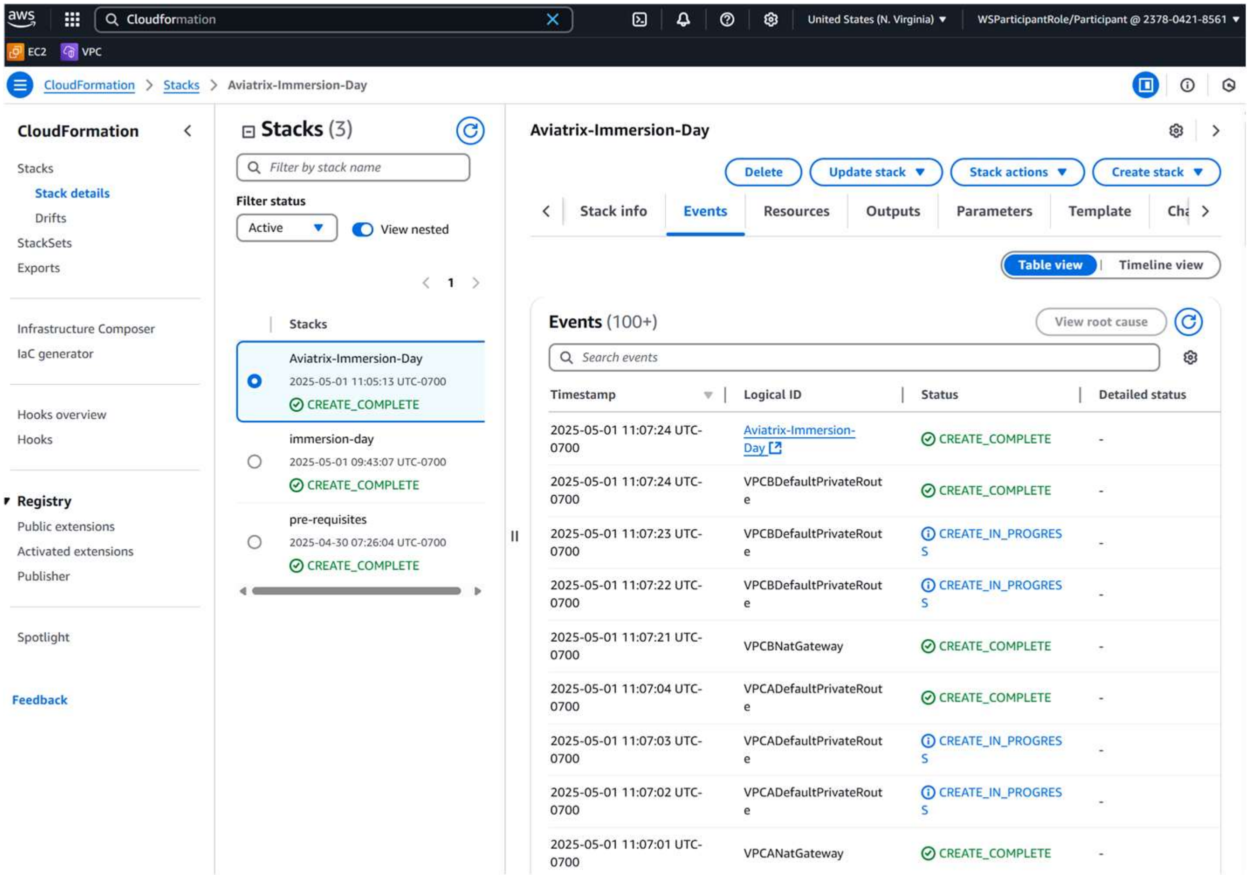
Lab 1: Step 1.7

Observe the stack deployment

On the next page you'll see your stack deploying if you select the **Events** tab

WSDefaultPolicy

The deployment should take <5 minutes to complete.



Stacks
● Aviatrix-Immersion-Day 2025-05-01 11:05:13 UTC-0700 CREATE_COMPLETE
○ immersion-day 2025-05-01 09:43:07 UTC-0700 CREATE_COMPLETE
○ pre-requisites 2025-04-30 07:26:04 UTC-0700 CREATE_COMPLETE

Events (100+)

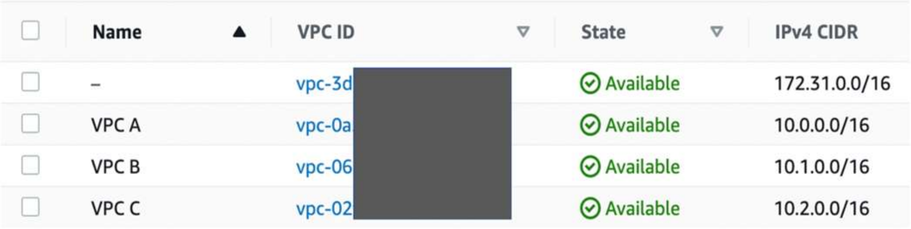
Timestamp	Logical ID	Status	Detailed status
2025-05-01 11:07:24 UTC-0700	Aviatrix-Immersion-Day	CREATE_COMPLETE	-
2025-05-01 11:07:24 UTC-0700	VPCBDefaultPrivateRoute	CREATE_COMPLETE	-
2025-05-01 11:07:23 UTC-0700	VPCBDefaultPrivateRoute	CREATE_IN_PROGRESS	-
2025-05-01 11:07:22 UTC-0700	VPCBDefaultPrivateRoute	CREATE_IN_PROGRESS	-
2025-05-01 11:07:21 UTC-0700	VPCBNatGateway	CREATE_COMPLETE	-
2025-05-01 11:07:04 UTC-0700	VPCADefaultPrivateRoute	CREATE_COMPLETE	-
2025-05-01 11:07:03 UTC-0700	VPCADefaultPrivateRoute	CREATE_IN_PROGRESS	-
2025-05-01 11:07:02 UTC-0700	VPCADefaultPrivateRoute	CREATE_IN_PROGRESS	-
2025-05-01 11:07:01 UTC-0700	VPCANatGateway	CREATE_COMPLETE	-

Lab 1: Step 1.8

Confirm Deployment of VPCs and CIDR Blocks

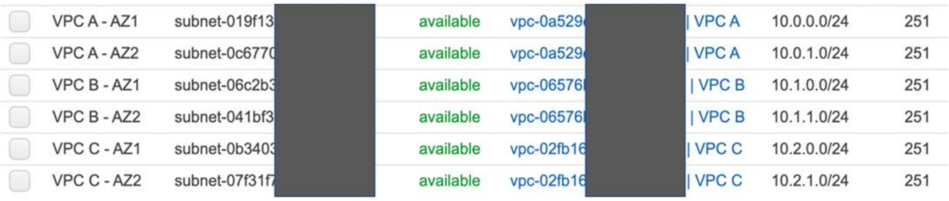
VPC Name	VPC CIDR block	Availability Zone	Availability Zone CIDR block
VPC A	10.0.0.0/16	us-east-1a	10.0.0.0/24
		us-east-1b	10.0.1.0/24
VPC B	10.1.0.0/16	us-east-1a	10.1.0.0/24
		us-east-1b	10.1.1.0/24
VPC C	10.2.0.0/16	us-east-1a	10.2.0.0/24
		us-east-1b	10.2.1.0/24

Here is the list of VPCs our **CloudFormation** template created and their assigned CIDR blocks



□	Name ▲	VPC ID ▽	State ▽	IPv4 CIDR
□	–	vpc-3d[redacted]	✓ Available	172.31.0.0/16
□	VPC A	vpc-0a[redacted]	✓ Available	10.0.0.0/16
□	VPC B	vpc-06[redacted]	✓ Available	10.1.0.0/16
□	VPC C	vpc-02[redacted]	✓ Available	10.2.0.0/16

1 Check **Your VPCs** and validate you see VPC A, B, and C and their assigned /16 CIDR Blocks

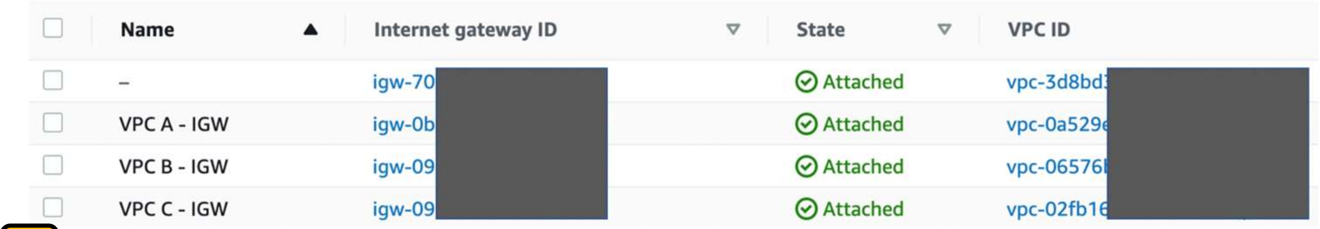


□	VPC A - AZ1	subnet-019f13[redacted]	available	vpc-0a529[redacted]	| VPC A	10.0.0.0/24	251
□	VPC A - AZ2	subnet-0c6770[redacted]	available	vpc-0a529[redacted]	| VPC A	10.0.1.0/24	251
□	VPC B - AZ1	subnet-06c2b3[redacted]	available	vpc-06576[redacted]	| VPC B	10.1.0.0/24	251
□	VPC B - AZ2	subnet-041bf3[redacted]	available	vpc-06576[redacted]	| VPC B	10.1.1.0/24	251
□	VPC C - AZ1	subnet-0b3403[redacted]	available	vpc-02fb16[redacted]	| VPC C	10.2.0.0/24	251
□	VPC C - AZ2	subnet-07f31f7[redacted]	available	vpc-02fb16[redacted]	| VPC C	10.2.1.0/24	251

2 Check **Subnets** and validate you see your VPC AZs with the properly assigned CIDR blocks

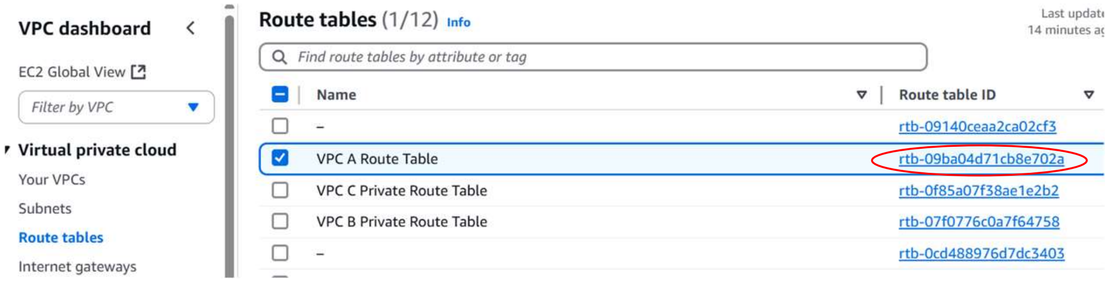
Lab 1: Step 1.9

Confirm Deployment of IGWs and Route Tables



□	Name ▲	Internet gateway ID ▽	State ▽	VPC ID
□	–	igw-70[redacted]	✓ Attached	vpc-3d8bd[redacted]
□	VPC A - IGW	igw-0b[redacted]	✓ Attached	vpc-0a529[redacted]
□	VPC B - IGW	igw-09[redacted]	✓ Attached	vpc-06576[redacted]
□	VPC C - IGW	igw-09[redacted]	✓ Attached	vpc-02fb16[redacted]

1 If you click on **Internet gateways** you should have an **IGW** for the default VPC and three newly created IGWs available and attached to the VPCs

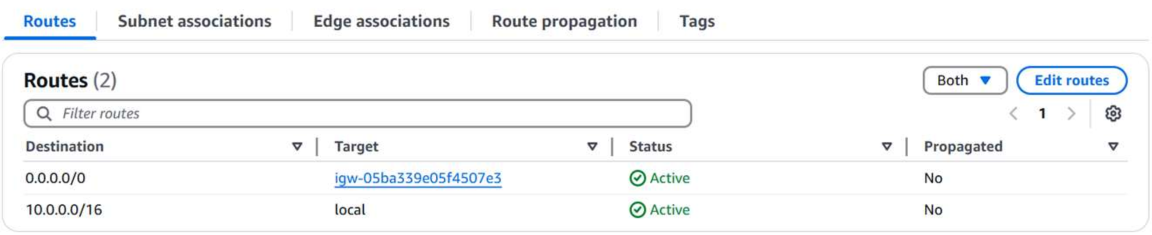


Route tables (1/12)

Find route tables by attribute or tag

▬	Name ▽	Route table ID ▽
□	–	rtb-09140ceaa2ca02cf3
☑	VPC A Route Table	rtb-09ba04d71cb8e702a
□	VPC C Private Route Table	rtb-0f85a07f38ae1e2b2
□	VPC B Private Route Table	rtb-07f0776c0a7f64758
□	–	rtb-0cd488976d7dc3403

If you click on **Route tables** you should see route tables for the VPC have been created. Click on the **Route table ID** for each VPC to see the route table **2**



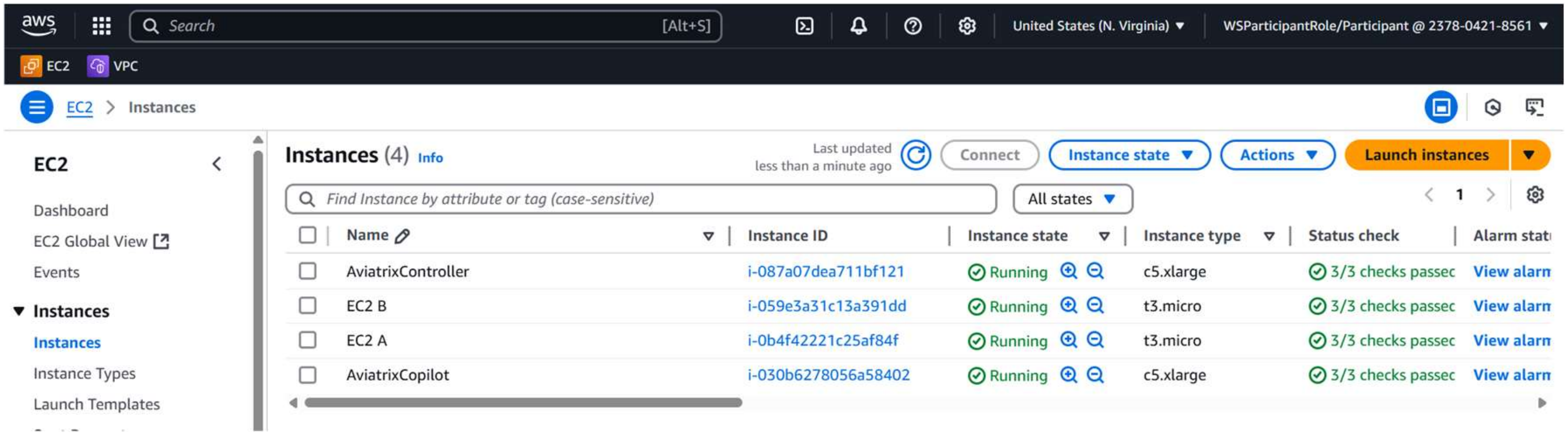
Routes (2)

Destination ▽	Target ▽	Status ▽	Propagated ▽
0.0.0.0/0	igw-05ba339e05f4507e3	✓ Active	No
10.0.0.0/16	local	✓ Active	No

Validate that the route tables for the VPC have been created, and updated to direct Internet-bound traffic to the IGW for each VPC. **3**

Lab 1: Step 1.10

Switch to **ec2** and click on **Instances**. You should see instances for the **AviatrixController**, **AviatrixCopilot**, **EC2 A**, and **EC2 B**. Verify that they are in the **Running** state.



Instances (4) Info

☐	Name	Instance ID	Instance state	Instance type	Status check	Alarm stat
☐	AviatrixController	i-087a07dea711bf121	✓ Running	c5.xlarge	✓ 3/3 checks passec	View alarn
☐	EC2 B	i-059e3a31c13a391dd	✓ Running	t3.micro	✓ 3/3 checks passec	View alarn
☐	EC2 A	i-0b4f42221c25af84f	✓ Running	t3.micro	✓ 3/3 checks passec	View alarn
☐	AviatrixCopilot	i-030b6278056a58402	✓ Running	c5.xlarge	✓ 3/3 checks passec	View alarn

This completes **Lab 1**