Lab 2 – AWS – FGT Active/Passive HA

**Goal** – Learn how to deploy FGT HA within AWS utilizing CloudFormation.

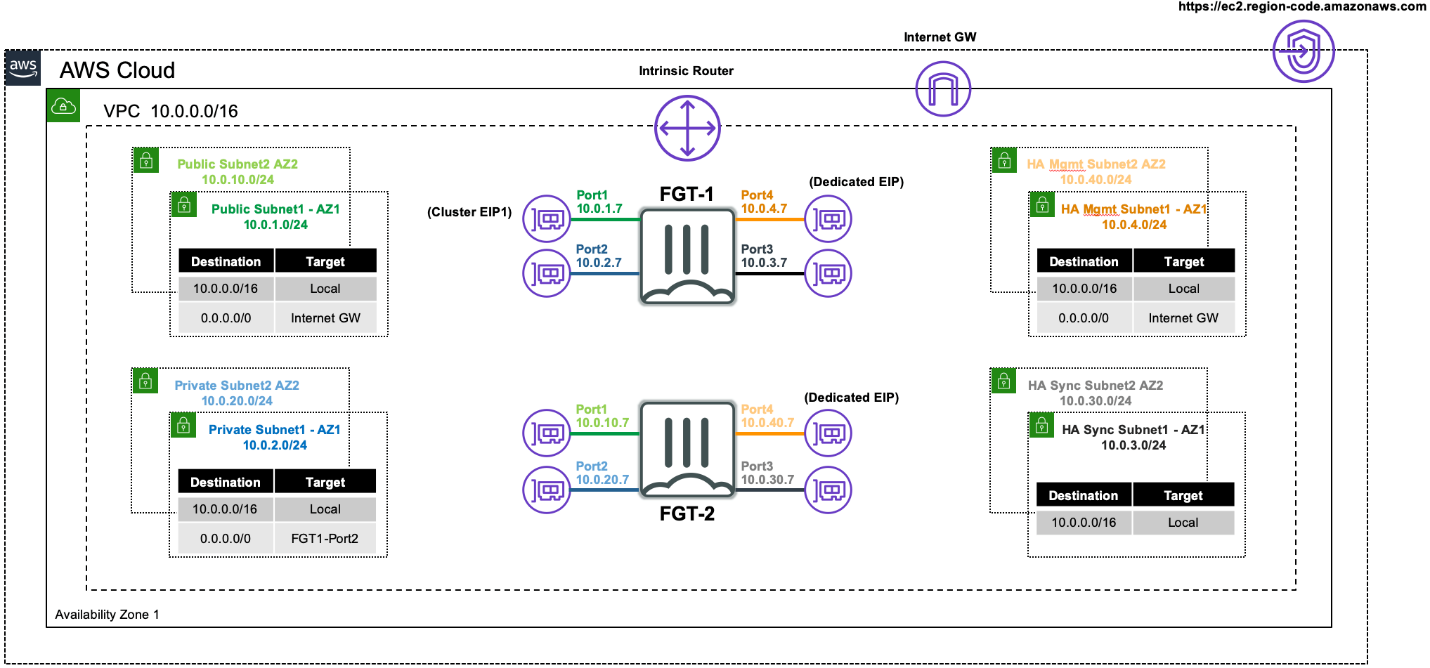
**Task** – You will deploy FGCP based A/P HA utilizing two FortiGate’s in the same AZ.

**Validation** – FortiGate HA will be fully deployed and operational.

**Introduction**

In this task, you will continue to learn the foundation of deploying networks in the cloud. While lab 1 focused on a single FGT deployment configured manually, this lab will introduce you to Infrastructure-as-a-service (IaC) by leveraging CloudFormation templates.

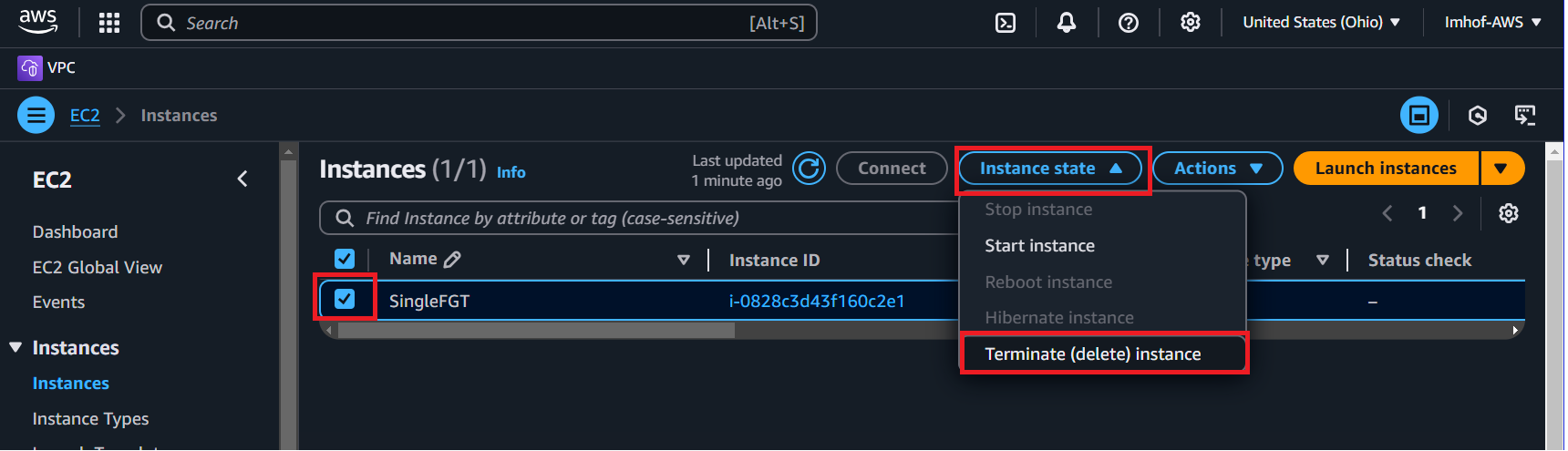
**Topology**



**Pre-Work:**

Delete the SingleFGT EC2 instance prior to starting this lab by navigating to the EC2 console:

**Click on the instance -> Instance state -> Terminate (delete) instance**



Text

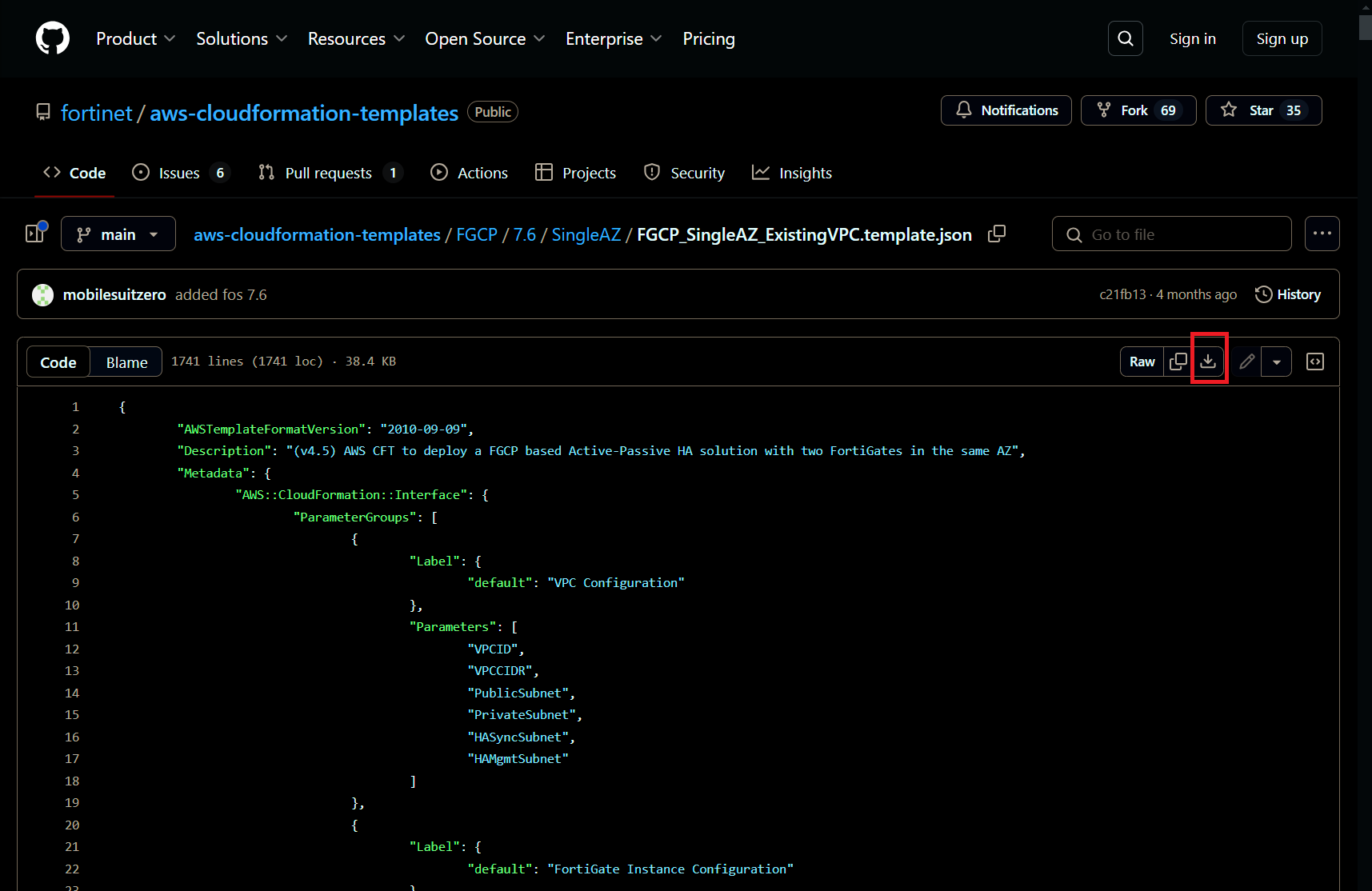
AI-generated content may be incorrect.

**Lab 2:**

1 – This lab will introduce the use of CloudFormation templates which is Infrastructure-as-code (IaC). As we saw in lab 1, there are a lot of different components that go into building cloud infrastructure and it’s easy to miss something. IaC allows you to avoid having to configure everything as the entire technology stack can be written in the template. For this lab we will be deploying FortiGate into an active/passive HA setup from a CloudFormation template written by Fortinet and located within the Fortinet Github page:

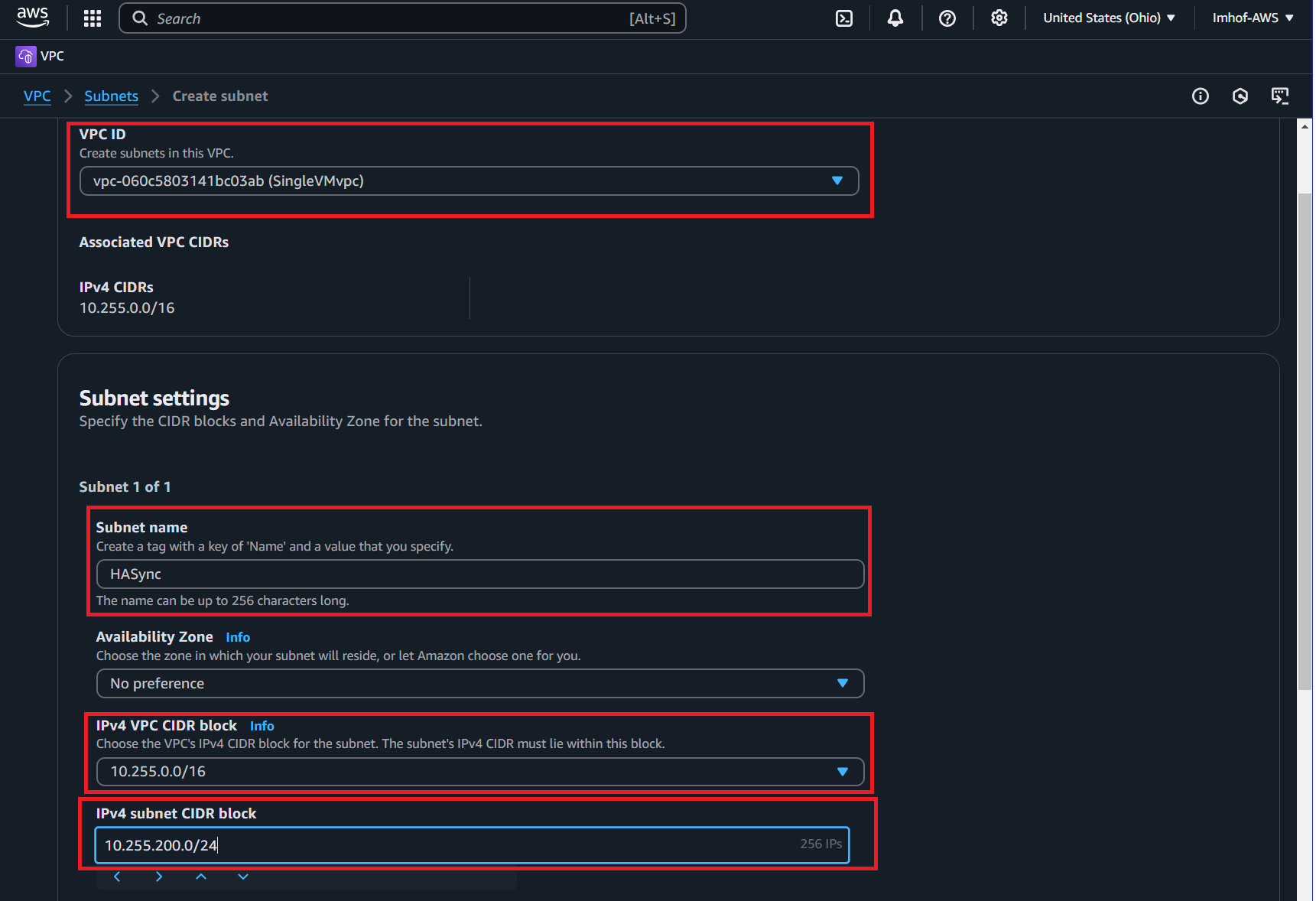
<https://github.com/fortinet/aws-cloudformation-templates/blob/main/FGCP/7.6/SingleAZ/FGCP_SingleAZ_ExistingVPC.template.json>

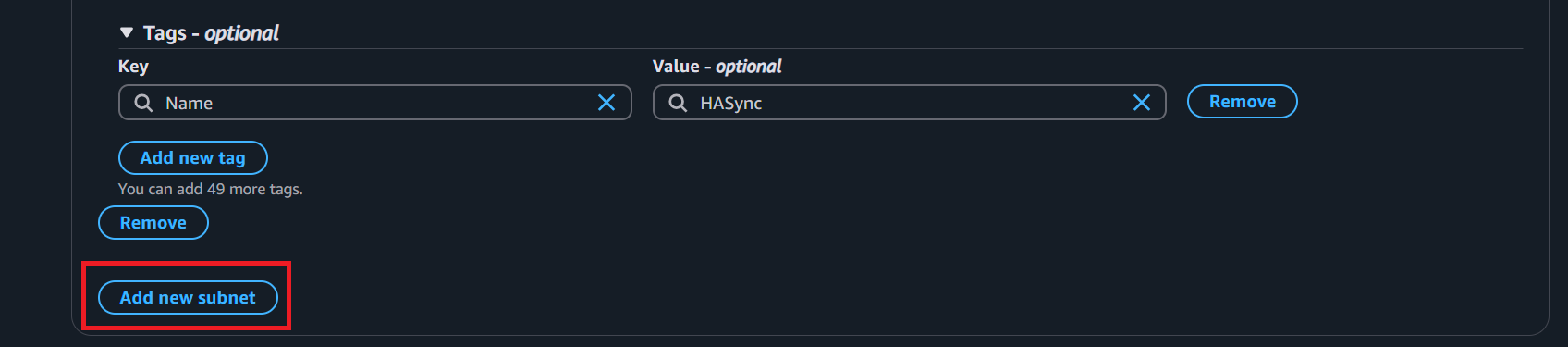
This template will deploy FortiGate into an A/P HA setup within the same VPC that we created in lab1. Download and save the JSON file as shown below:



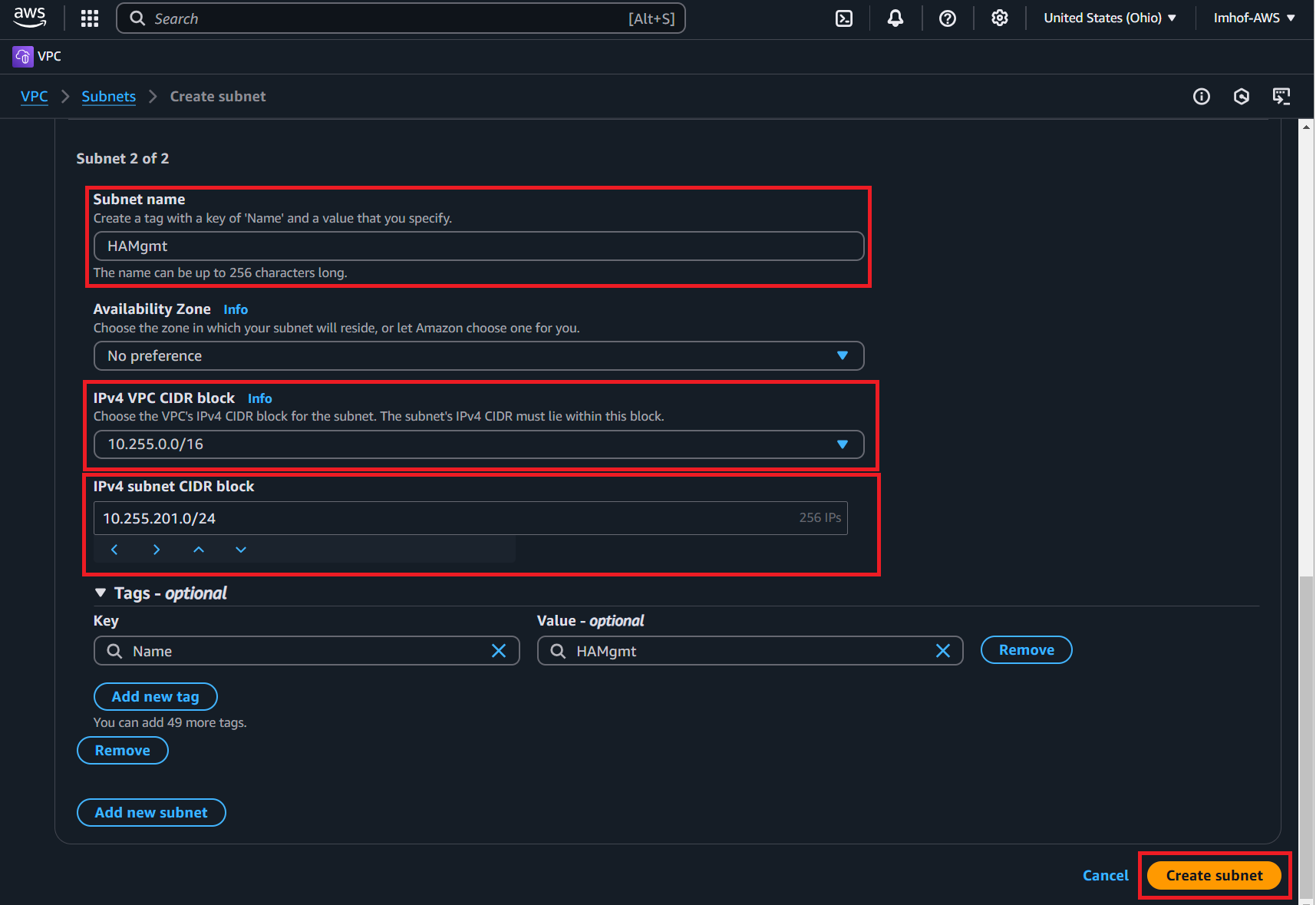
2 – Create the HA Sync and HA management subnets by navigating to the VPC console and clicking on Subnets. Utilize the following parameters:

* VPC ID: **SingleVMvpc**
* Subnet name: **HASync**
* IPv4 VPC CIDR block: **10.255.0.0/16**
* IPv4 subnet CIDR block: **10.255.200.0/24**
* Click on **Add new subnet**





* Subnet name: **HAMgmt**
* IPv4 VPC CIDR block: **10.255.0.0/16**
* IPv4 subnet CIDR block: **10.255.201.0/24**
* Click on **Create subnet**



3 – Search for S3 and create a general purpose storage bucket:

A screenshot of a computer

AI-generated content may be incorrect.

Utilize the following parameters for the S3 bucket:

* Bucket type: **General purpose**
* Bucket name: **fgtbucket<last name>**
* Object ownership: **ACLs disabled**
* Block Public Access settings for this bucket: **Enable**
* Bucket Versioning: **Disable**
* Encryption type: **Server-side encryption with Amazon S3 managed keys (SSE-S3)**
* Bucket Key: **Enable**
* Click on **Create bucket**

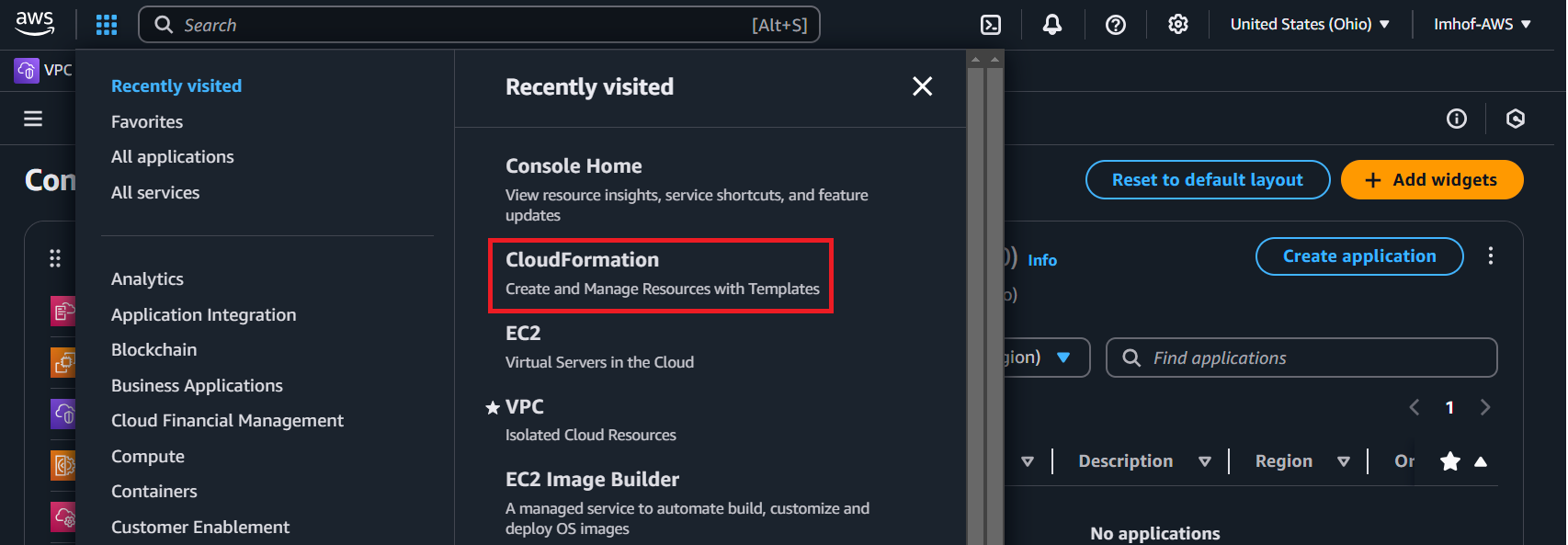




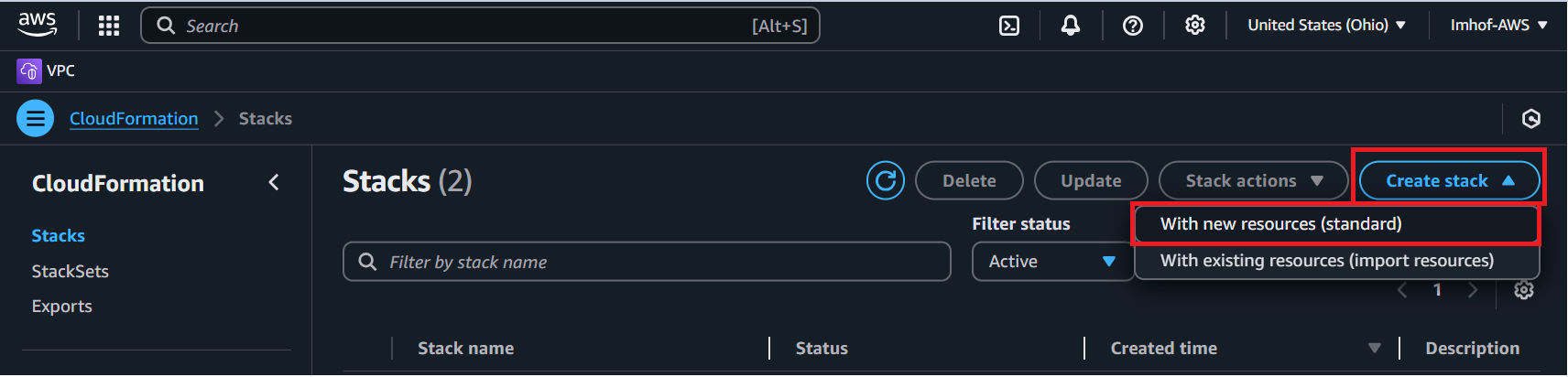
A screenshot of a computer

AI-generated content may be incorrect.

4 – Log into AWS and navigate to the CloudFormation page by using the search bar or by clicking on the 9 squares:

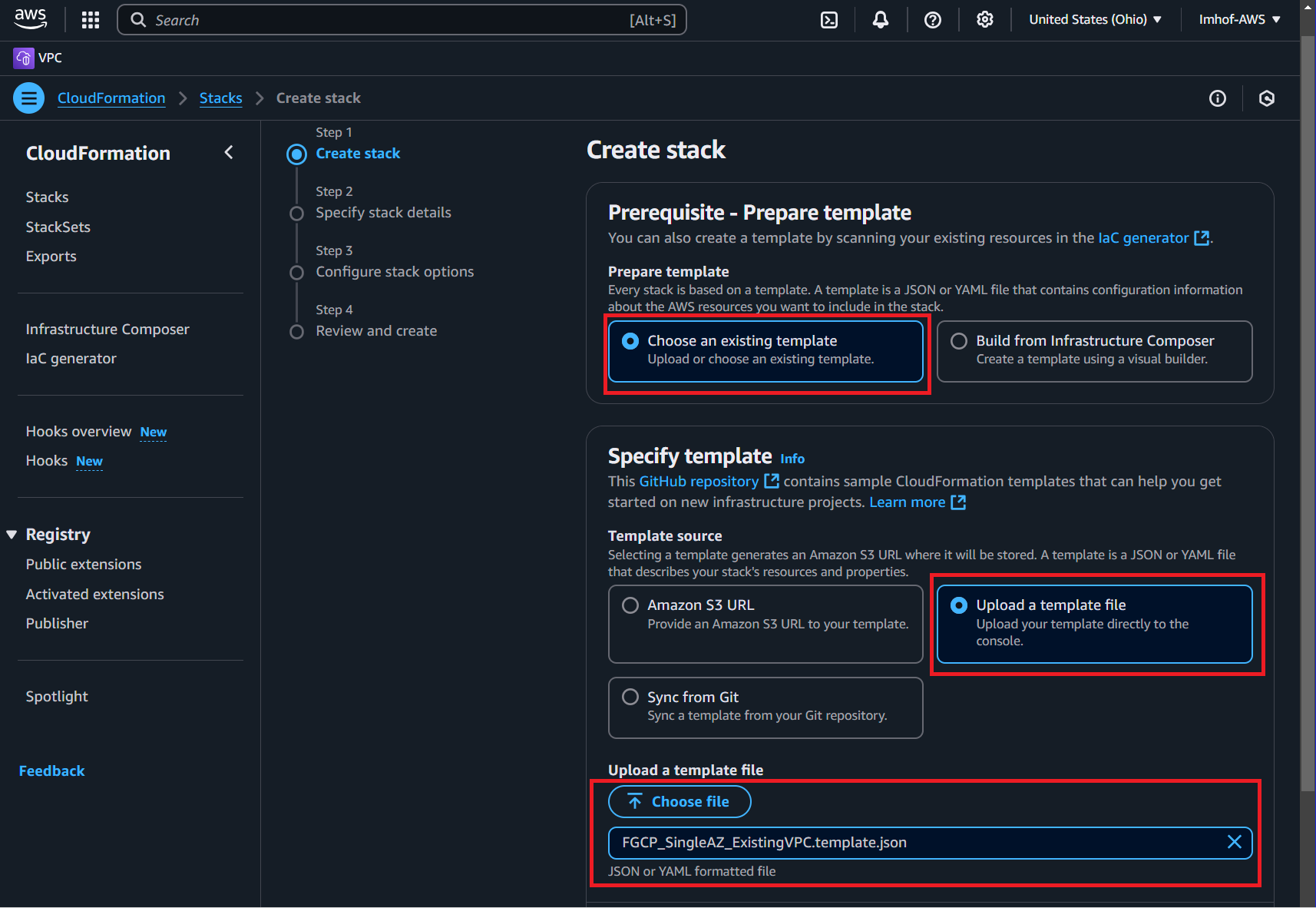


5 – **Create a new stack with new resources**:



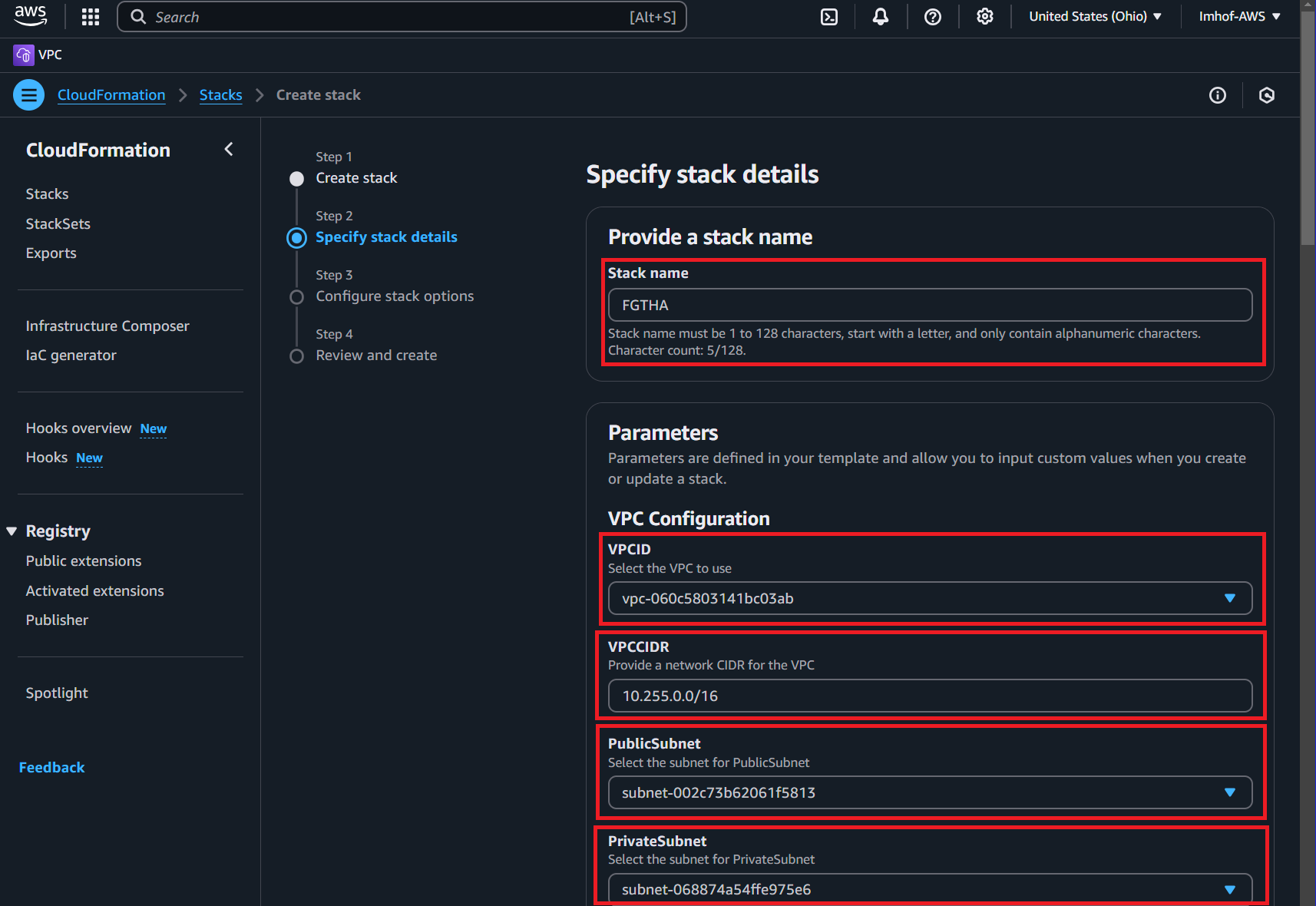
6 – Configure your new stack utilizing the following parameters:

* Prerequisite – Prepare template: **Choose an existing template**
* Specify Template: **Upload a template file**
* Choose file: **FGCP\_SingleAZ\_ExistingVPC.template.json**
* Click on **Next**



7 – Configure the stack using the following parameters:

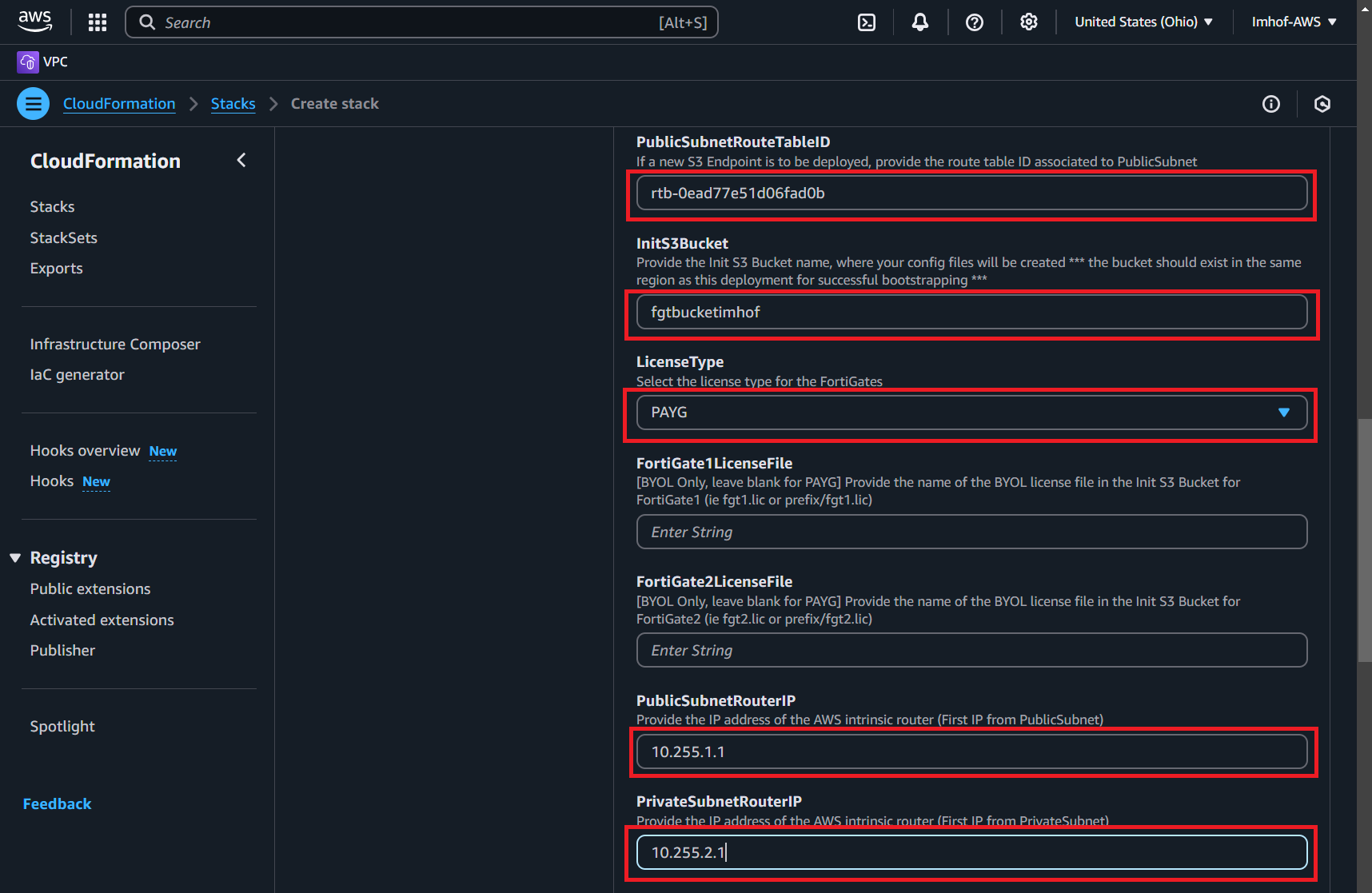
* Stack name: **FGTHA**
* VPCID: Select **SingleVMvpc**
* VPC CIDR: **10.255.0.0/16**
* PublicSubnet: **singleVMpublic**
* PrivateSubnet: **singleVMprivate**



* HASyncSubnet: **HASync**
* HAMgmtSubnet: **HAMgmt**
* InstanceType: **C5.xlarge**
* CIDRForInstanceAccess: **10.255.1.0/24**
* KeyPair: **KeyPair – FGT**
* S3EndpointDeployment: **DeployNew**



* PublicSubnetRouteTableID: **rtb-0ead77e51d06fad0b**
  + Grab this identifier from the route table page for “SingleVMpublicRT”
* InitS3Bucket: **fgtbucket<last name>**
* LicenseType: **PAYG**
* PublicSubnetRouterIP: **10.255.1.1**
* PrivateSubnetRouterIP: **10.255.2.1**

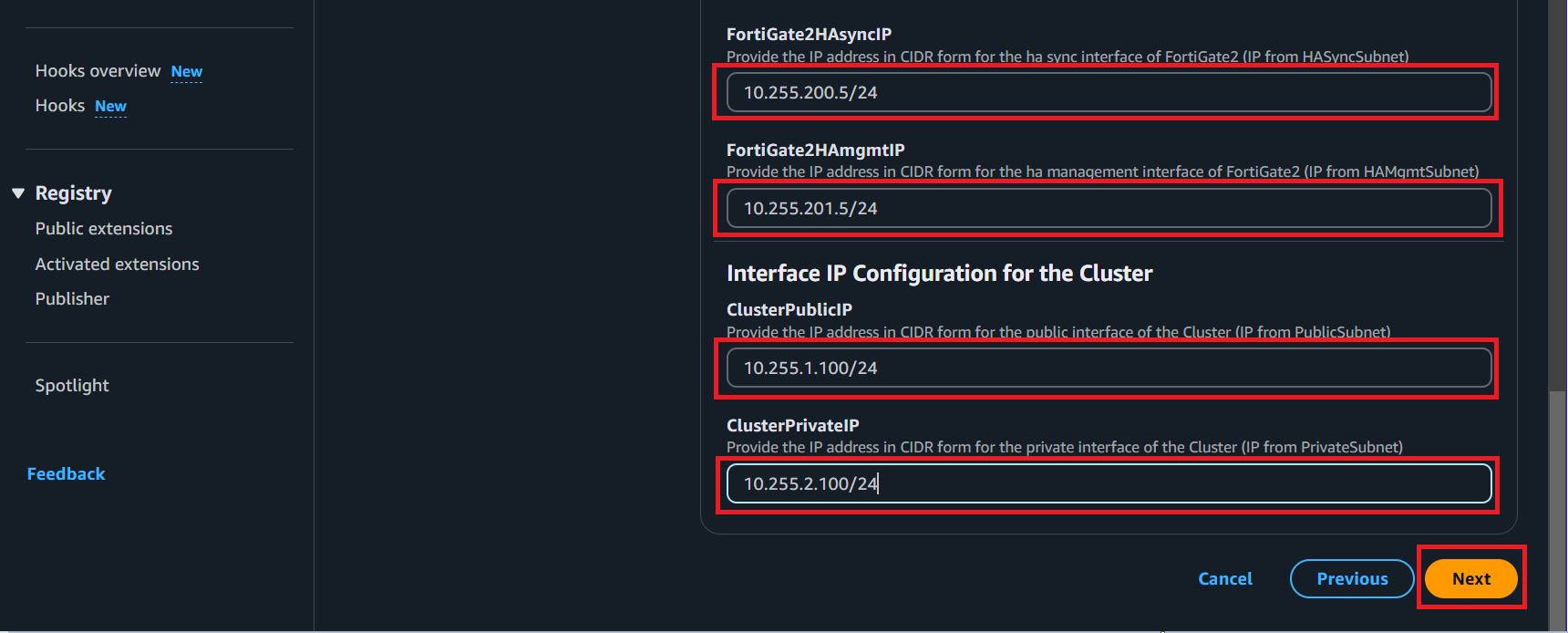


* HAMgmtSubnetRouterIP: **10.255.201.1**
* FortiGate1PublicIP: **10.255.1.4/24**
* FortiGate1PrivateIP: **10.255.2.4/24**
* FortiGate1HAsyncIP: **10.255.200.4/24**
* FortiGate1HAmgmtIP: **10.255.201.4/24**
* FortiGate2PublicIP: **10.255.1.5/24**
* FortiGate2PrivateIP: **10.255.2.5/24**

A screenshot of a computer

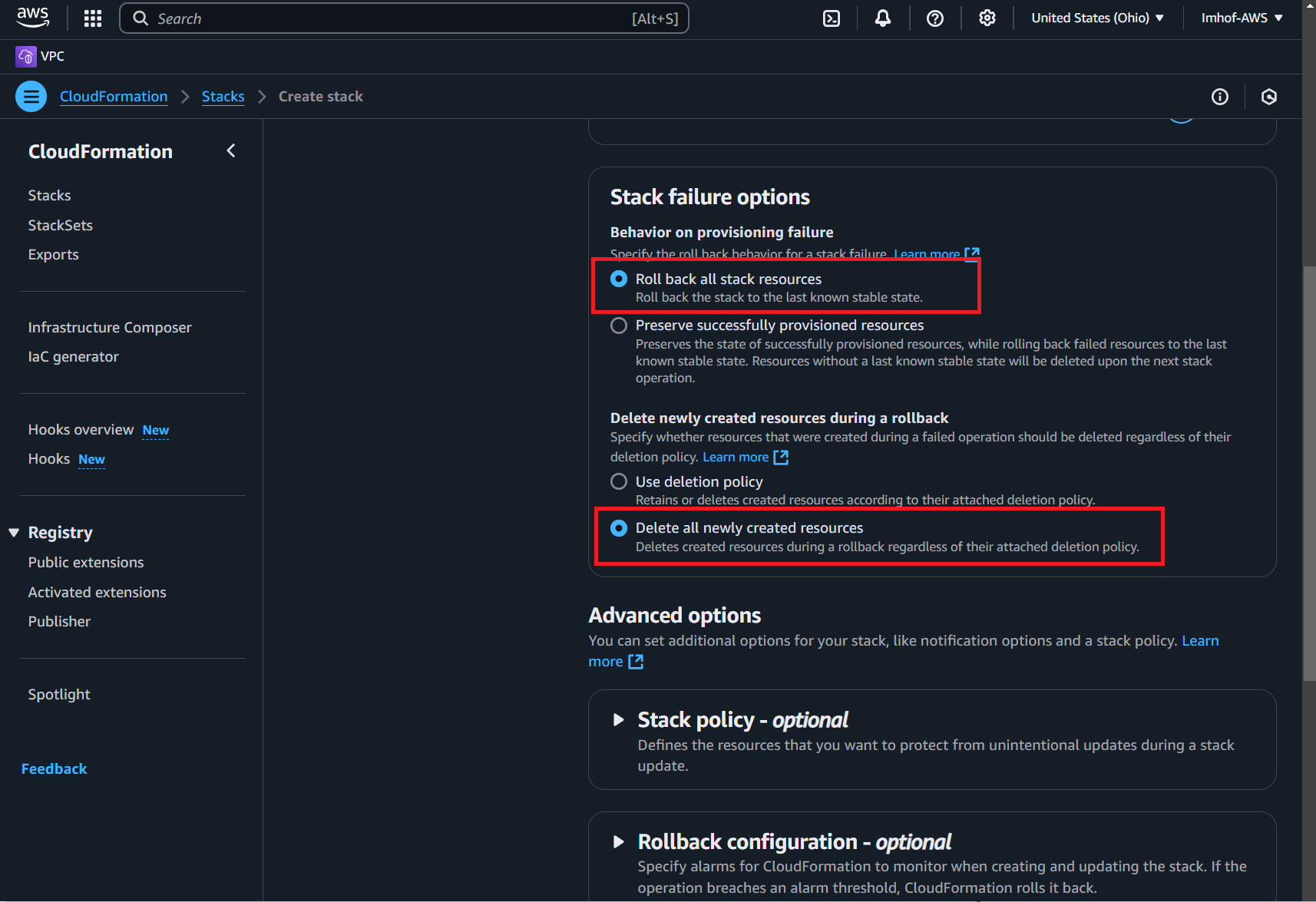
AI-generated content may be incorrect.

* FortiGate2HAsyncIP: **10.255.200.5/24**
* FortiGate2HAmgmtIP: **10.255.201.5/24**
* ClusterPublicIP: **10.255.1.100/24**
* ClusterPrivateIP: **10.255.2.100/24**
* Click on **Next**



8 – Configure the template to roll back the created resources if there are issues during deployment:

* Enable **Roll back all stack resources**
* Enable **Delete all newly created resources**



* **Acknowledge the capabilities disclaimer**
* Click **Next**

A screenshot of a computer

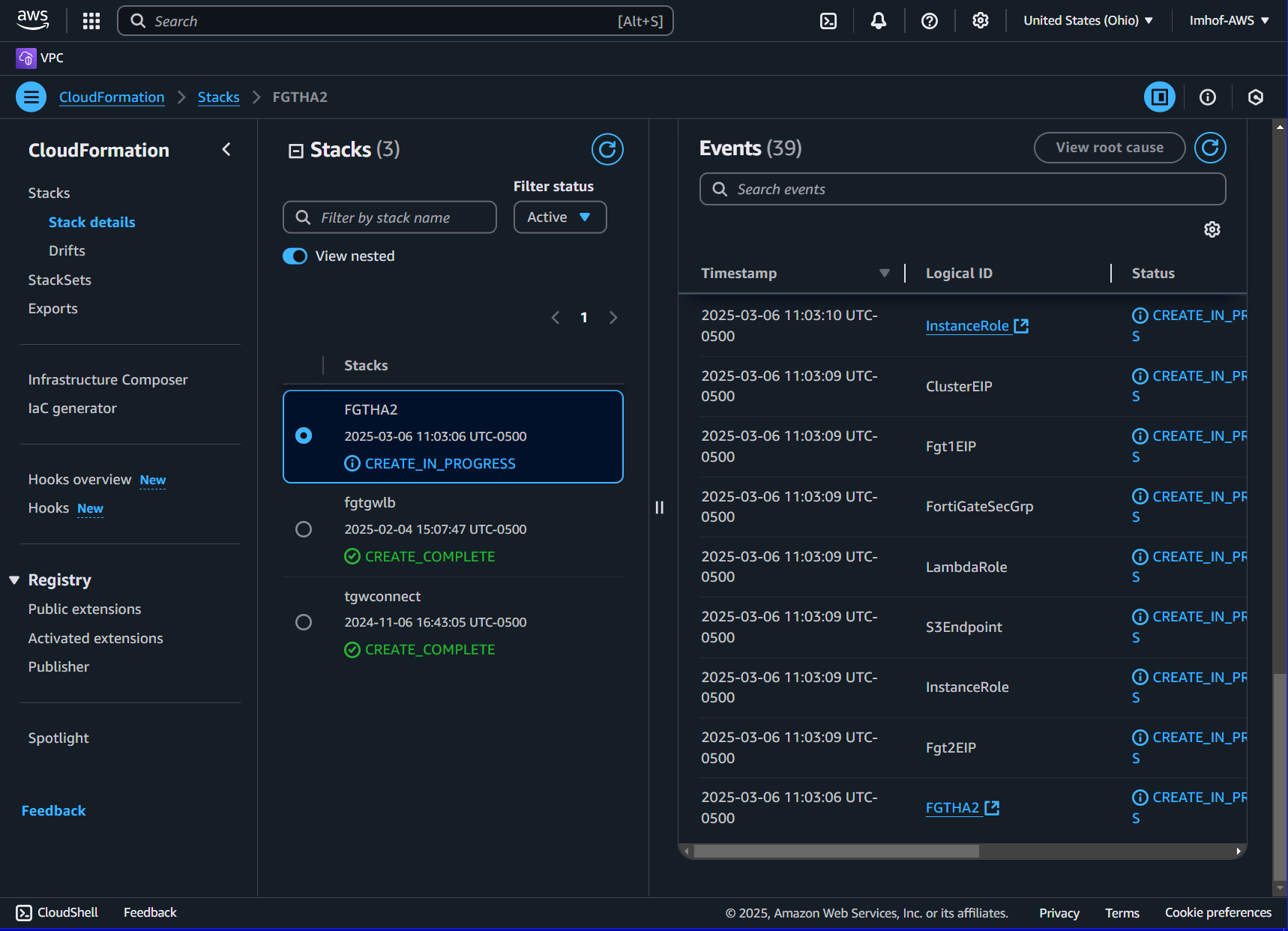
AI-generated content may be incorrect.

9 – Review the template parameters then click on **Submit**

A screenshot of a computer screen

AI-generated content may be incorrect.

10 – This brings you to the CloudFormation stacks page where you can check on its status and address any errors that arise:



**Note – This screenshot shows the stack named as FGTHA2 but yours will be named FGTHA**

Once the stack shows its status as CREATE\_COMPLETE, and once FortiGate fully boots, you can access your AWS resources

A screenshot of a computer

AI-generated content may be incorrect.