**VPC with servers in private subnets**

This example demonstrates how to create a VPC that you can use for servers in a production environment.

To improve resiliency, you deploy the servers in two Availability Zones, by using an **Auto Scaling group** and an Application Load Balancer. For additional security, you deploy the servers in **private subnets**. The servers receive requests through the load balancer. The servers can connect to the internet by using a **NAT gateway**. To improve resiliency, you deploy the NAT gateway in both Availability Zones.

**Overview**

* The VPC has public subnets and private subnets in two Availability Zones.
* Each public subnet contains a NAT gateway and a load balancer node.
* The servers run in the private subnets, are launched and terminated by using an Auto Scaling group, and receive traffic from the load balancer.
* The servers can connect to the internet by using the NAT gateway.
* The servers can connect to Amazon S3 by using a gateway VPC endpoint.

Step1:

**1. Create the VPC**

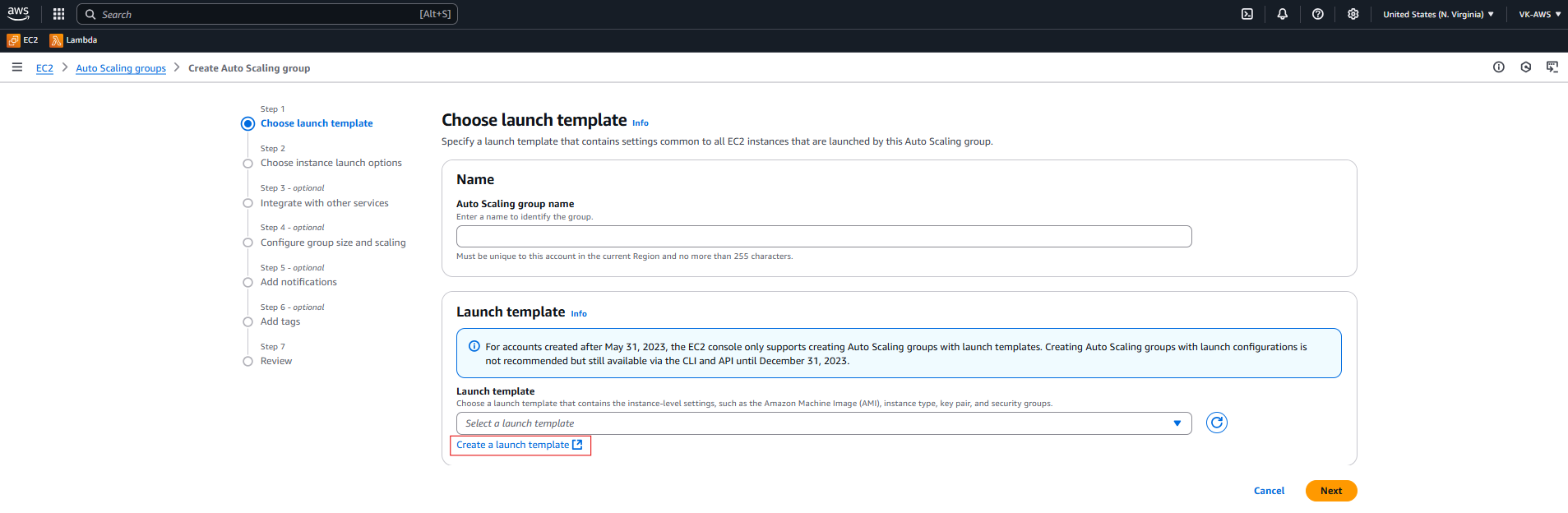
Use the following procedure to create a VPC with a public subnet and a private subnet in two Availability Zones, and a NAT gateway in each Availability Zone.

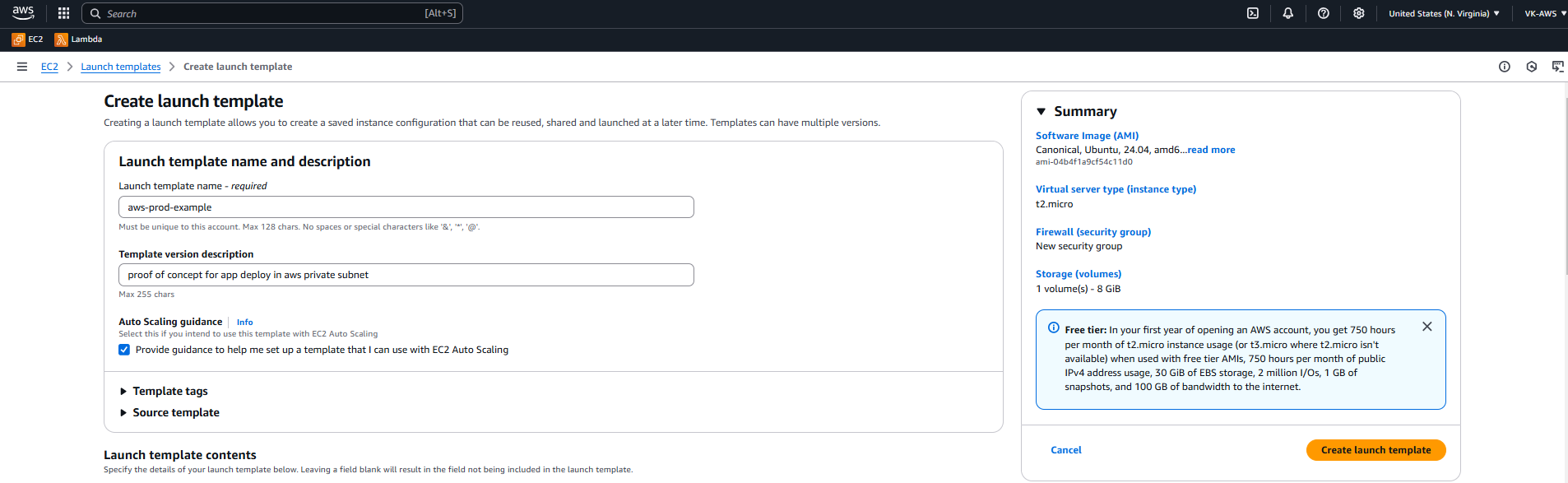
**To create the VPC**

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. On the dashboard, choose **Create VPC**.
3. For **Resources to create**, choose **VPC and more**.
4. **Configure the VPC**
   1. For **Name tag auto-generation**, enter a name for the VPC.
   2. For **IPv4 CIDR block**, you can keep the default suggestion, or alternatively you can enter the CIDR block required by your application or network.
   3. If your application communicates by using IPv6 addresses, choose **IPv6 CIDR block**, **Amazon-provided IPv6 CIDR block**.
5. **Configure the subnets**
   1. For **Number of Availability Zones**, choose **2**, so that you can launch instances in multiple Availability Zones to improve resiliency.
   2. For **Number of public subnets**, choose **2**.
   3. For **Number of private subnets**, choose **2**.
   4. You can keep the default CIDR block for the public subnet, or alternatively you can expand **Customize subnet CIDR blocks** and enter a CIDR block. For more information, see [Subnet CIDR blocks](https://docs.aws.amazon.com/vpc/latest/userguide/subnet-sizing.html).
6. For **NAT gateways**, choose **1 per AZ** to improve resiliency.
7. If your application communicates by using IPv6 addresses, for **Egress only internet gateway**, choose **Yes**.
8. For **VPC endpoints**, if your instances must access an S3 bucket, keep the **S3 Gateway** default. Otherwise, instances in your private subnet can't access Amazon S3. There is no cost for this option, so you can keep the default if you might use an S3 bucket in the future. If you choose **None**, you can always add a gateway VPC endpoint later on.
9. For **DNS options**, clear **Enable DNS hostnames**.
10. Choose **Create VPC**.

A screenshot of a computer

Description automatically generated



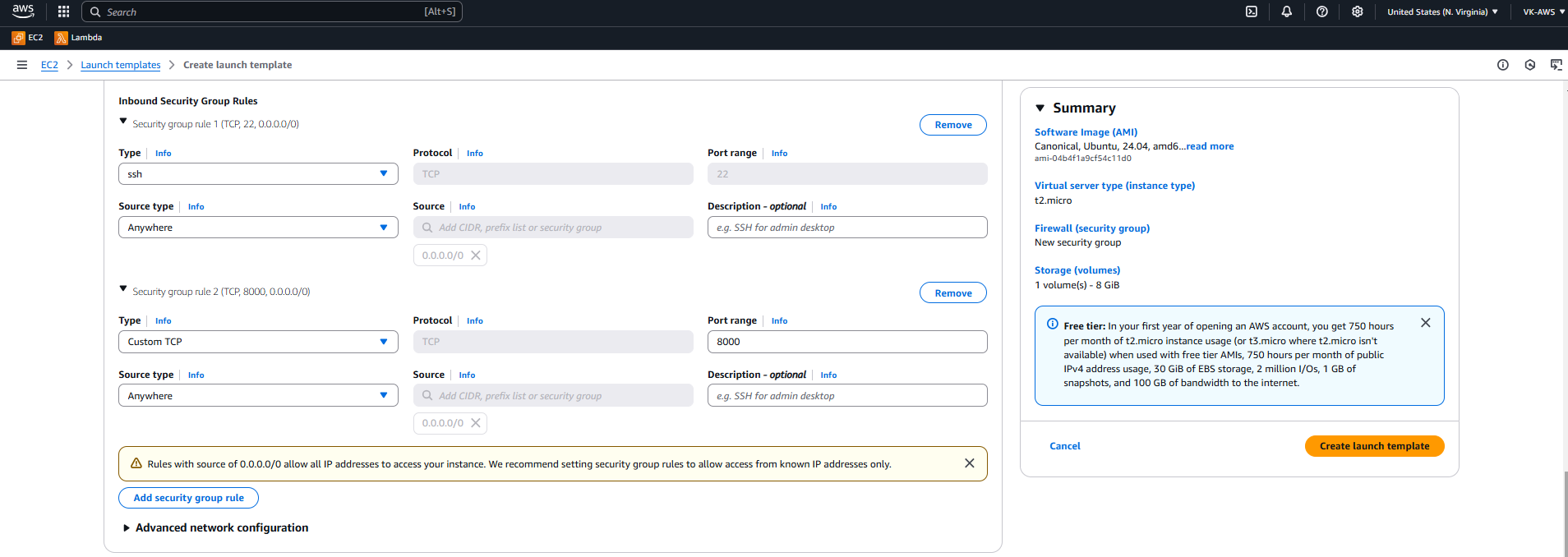


A screenshot of a computer

Description automatically generated

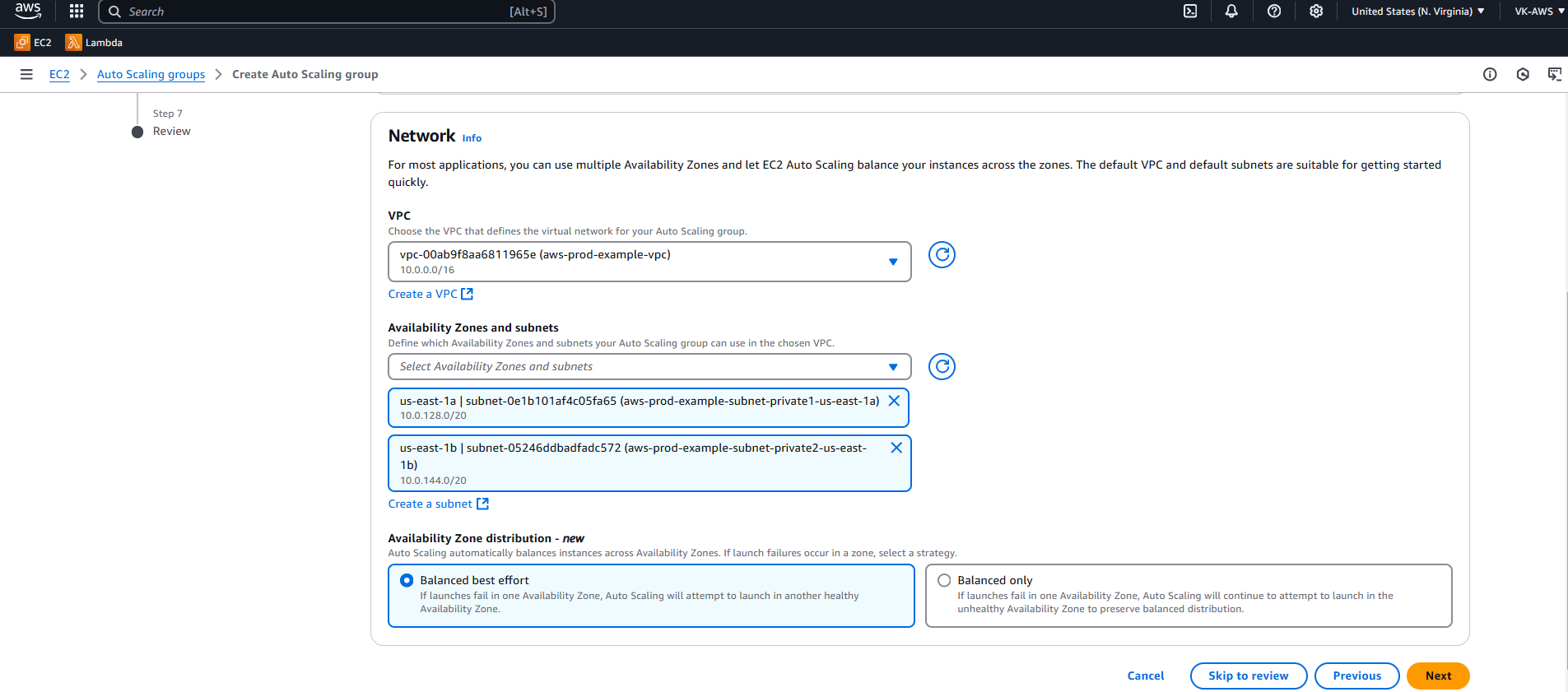
A screenshot of a computer

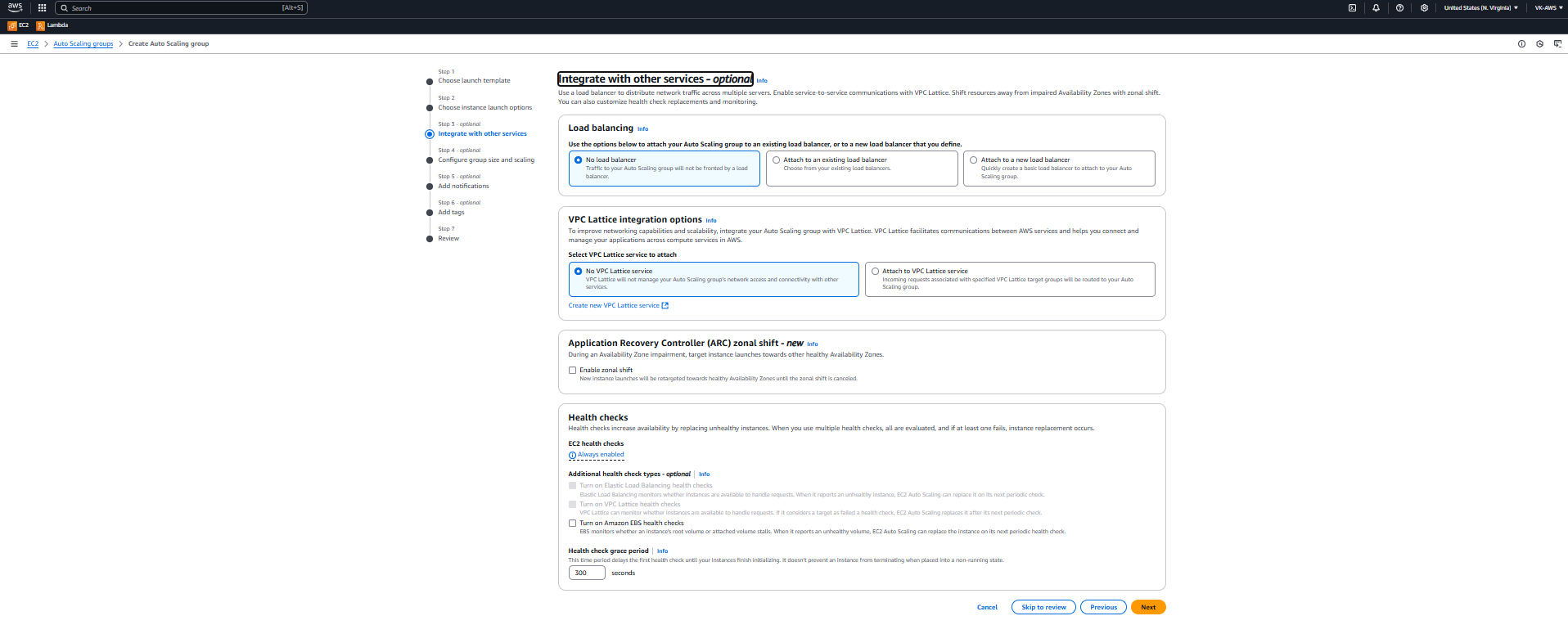
Description automatically generated



A screenshot of a computer

Description automatically generated





A screenshot of a computer

Description automatically generated

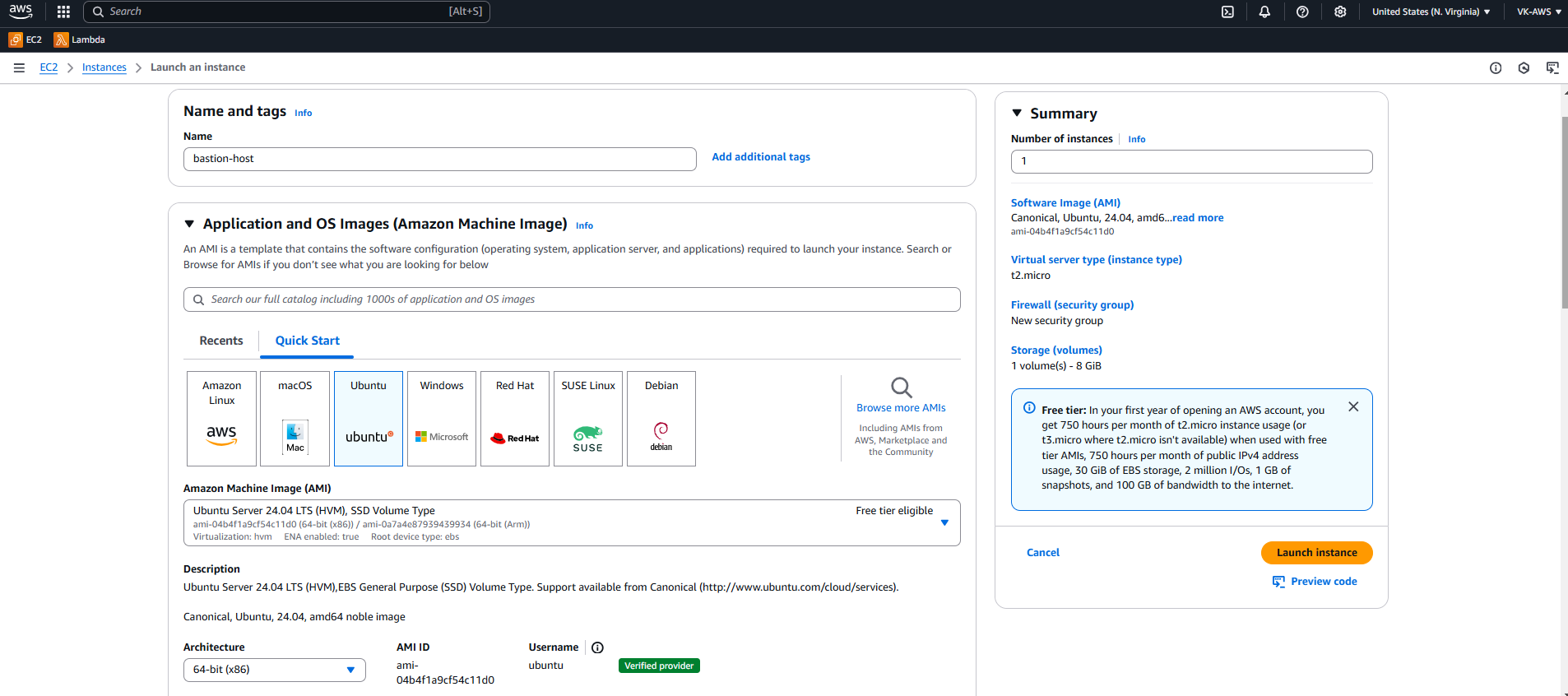
A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Creating Bastion-Host:



Make sure edit Network settings

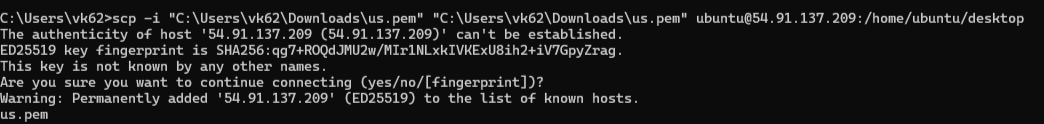
A screenshot of a computer

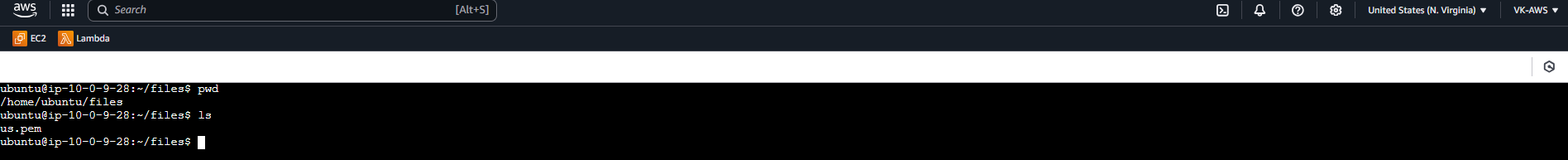
Description automatically generated

Copy the .pem file to bastion host

Windows:

scp -i "C:\Users\vk62\Downloads\us.pem" "C:\Users\vk62\Downloads\us.pem" [ubuntu@54.91.137.209:/home/ubuntu/files](mailto:ubuntu@54.91.137.209:/home/ubuntu/files)





Change file permission: “chmod 400 us.pem”

Ssh to instance 1 using private IP

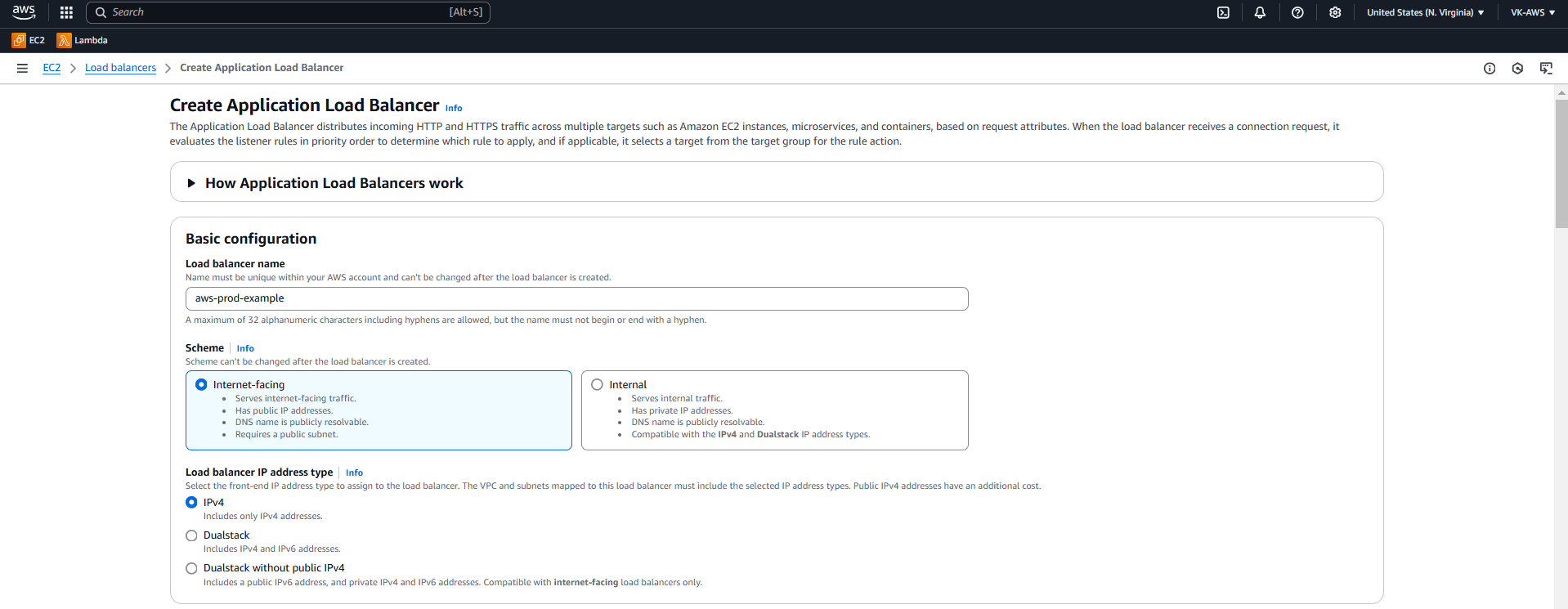
A black and white rectangular object

Description automatically generated

A black and white rectangle with a white line

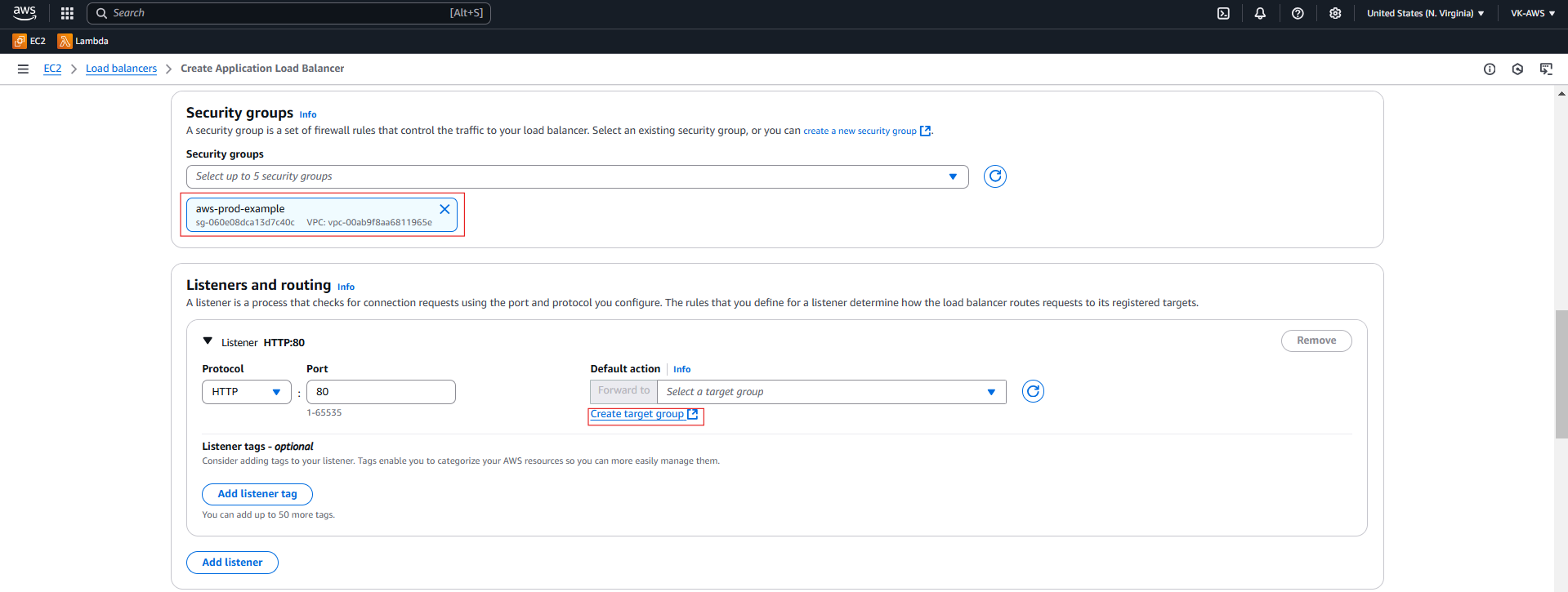
Description automatically generated

Creating Load balancer:



A screenshot of a computer

Description automatically generated



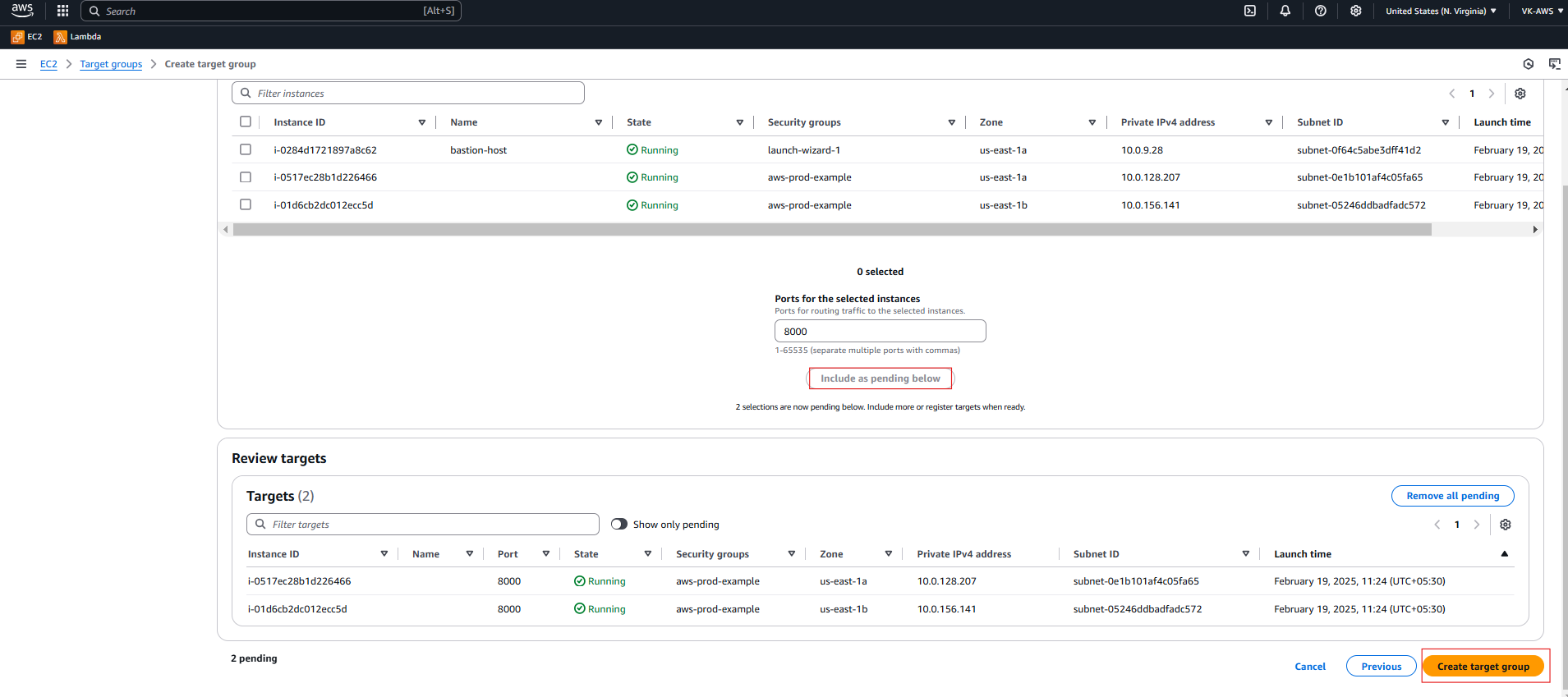
A screenshot of a computer

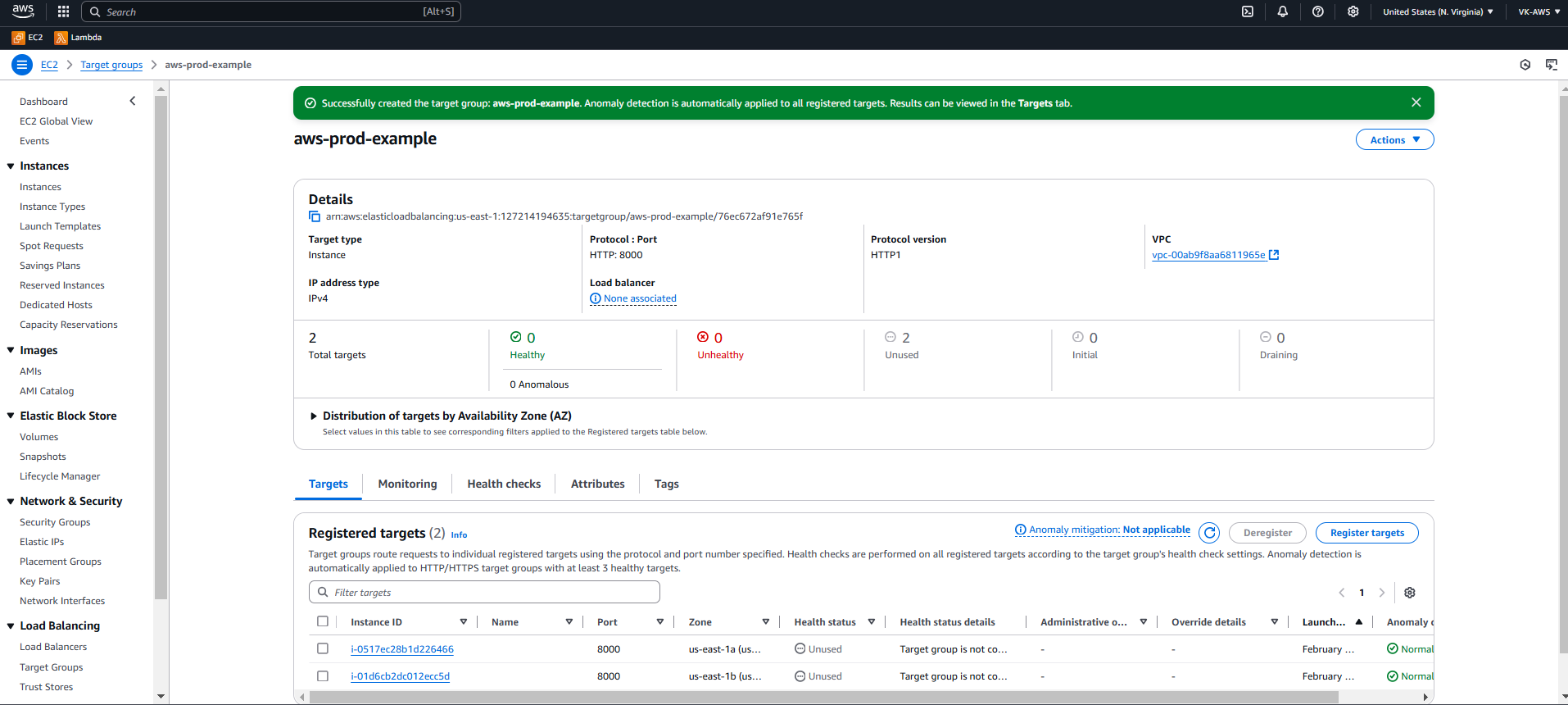
Description automatically generated

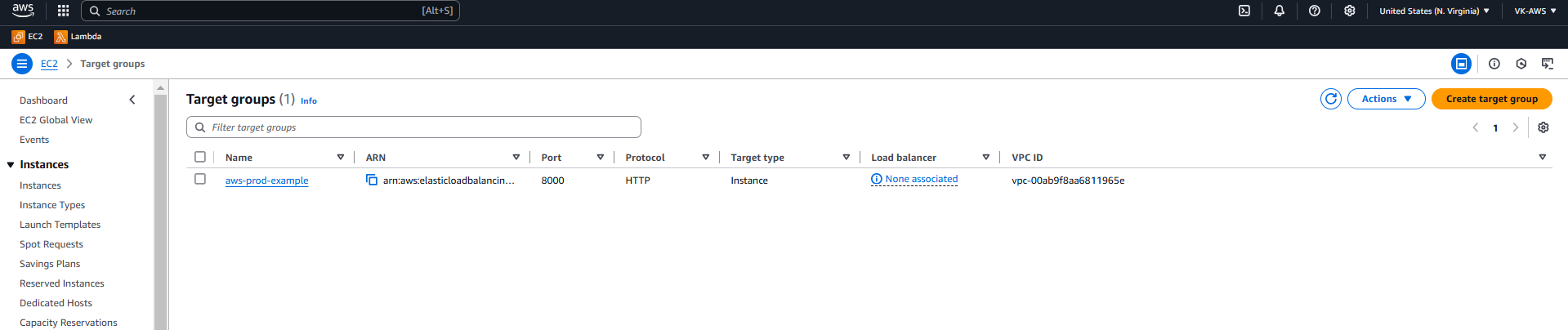
A screenshot of a computer

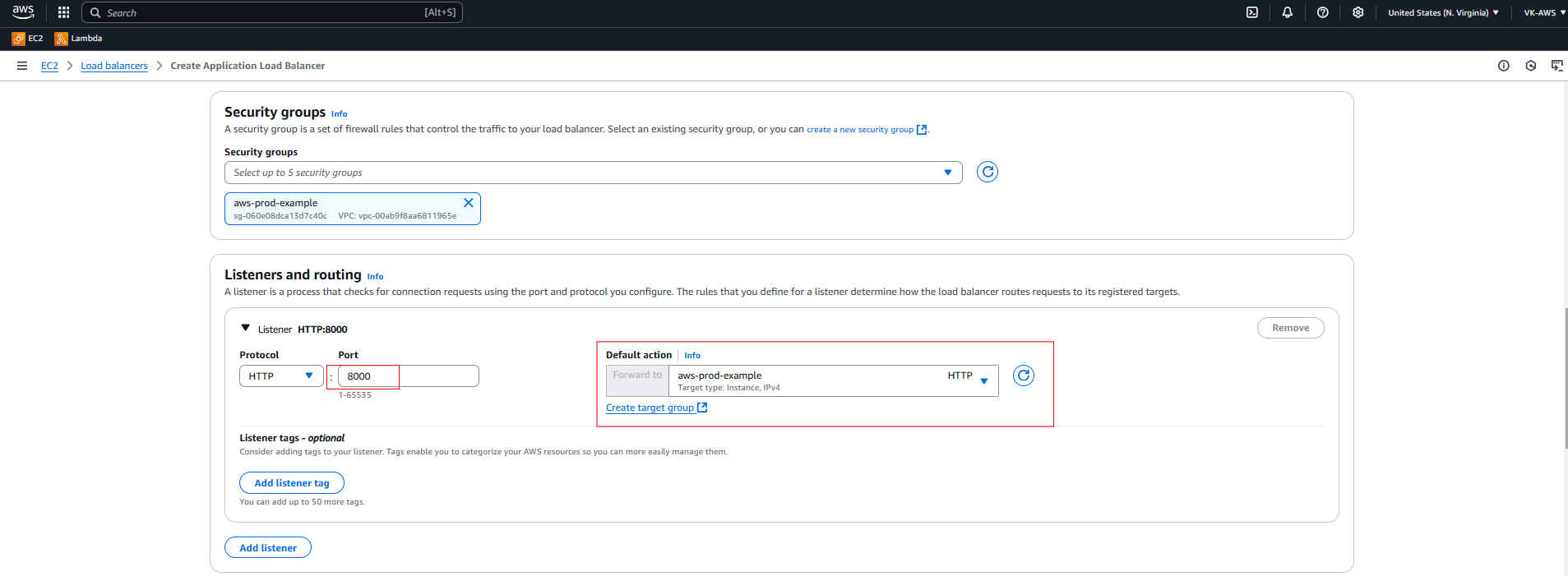
Description automatically generated

Click on “include as below” 🡪 Create target group









Keep the port as 80

**✅ Why Port 80 Works**

* **Port 80 is a standard HTTP port** used by browsers.
* The **ALB allows traffic on port 80 by default**.
* Your security group is correctly allowing HTTP traffic on port 80.

**❌ Why Port 8000 Doesn't Work**

1. **ALB Doesn't Handle Port 8000 by Default**
   * ALB is designed for **web traffic (port 80/443)**.
   * It might not properly handle requests on **non-standard ports like 8000**.

