Hands-on tutorials

Deploy Docker Containers on Amazon ECS

Deploy Docker Containers on Amazon ECS: Hands-on tutorials

Table of Contents

Deploy Docker Containers on Amazon ECS

AWS experience	Beginner
Time to complete	10 minutes
Cost to complete	Cost will vary by region, and will be around $0.004 / hour of running the container
Services used	Amazon ECS AWS Fargate Elastic Load Balancing
Last updated	August 11, 2022

Overview

Amazon Elastic Container Service (Amazon ECS) is the AWS service you use to run Docker applications on a scalable cluster. In this how-to guide, you will learn how to run a Docker-enabled sample application on an Amazon ECS cluster behind a load balancer, test the sample application, and delete your resources to avoid charges. This guide uses AWS Fargate, which has a ~$0.004 (less than half of a US cent) cost per hour when using the 0.25 vCPU / 0.5 GB configuration.

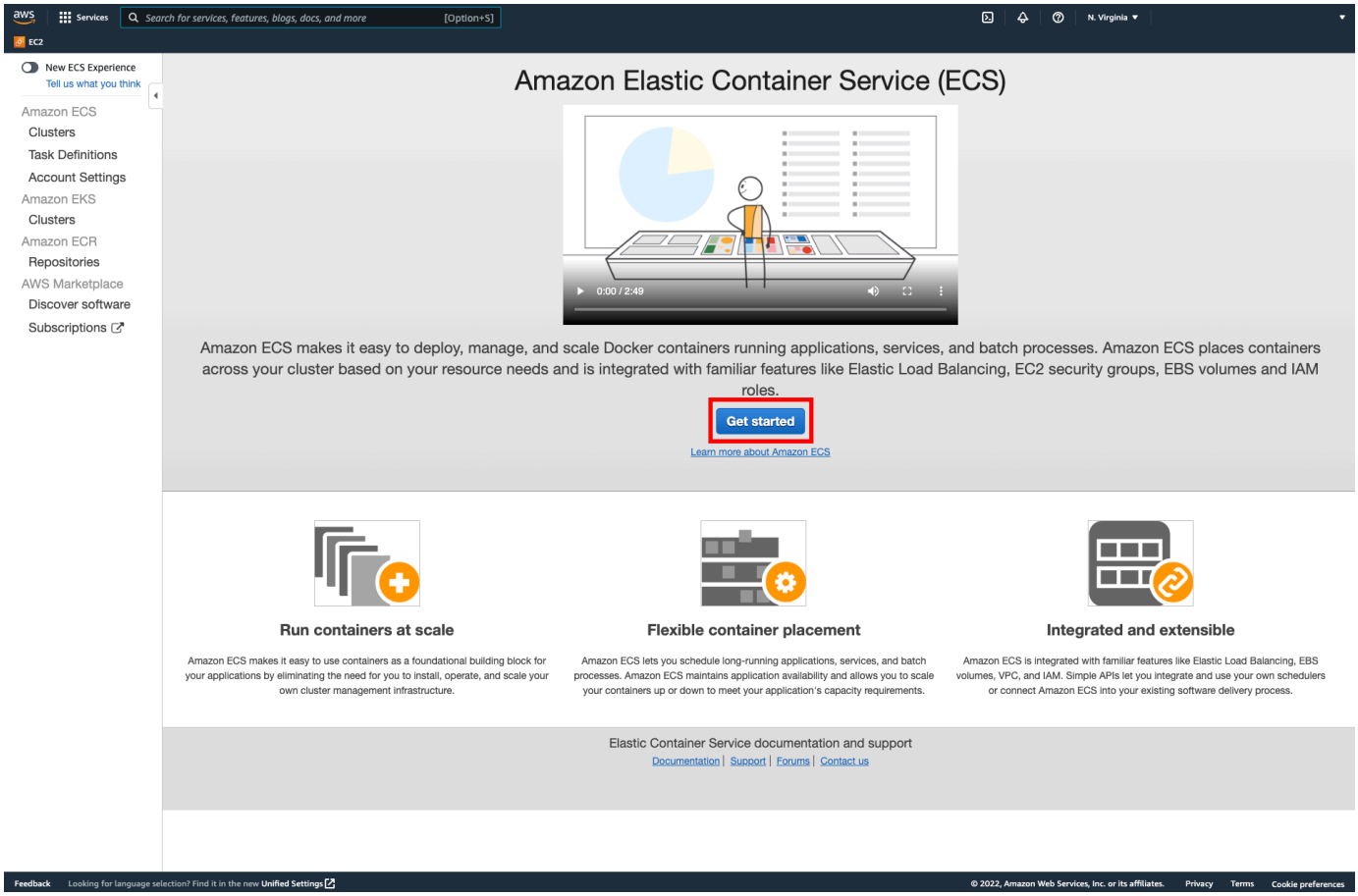
Implementation

Step 1: Set up your first run with Amazon ECS

The Amazon ECS first-run wizard will guide you through creating a cluster and launching a sample web application. In this step, you will enter the Amazon ECS console and launch the wizard.

- Launch the first-run wizard

  To launch the Amazon ECS first-run wizard, choose the **Get started** button. (If your layout looks different, disable the **New ECS Experience** toggle button at the top left of the console).

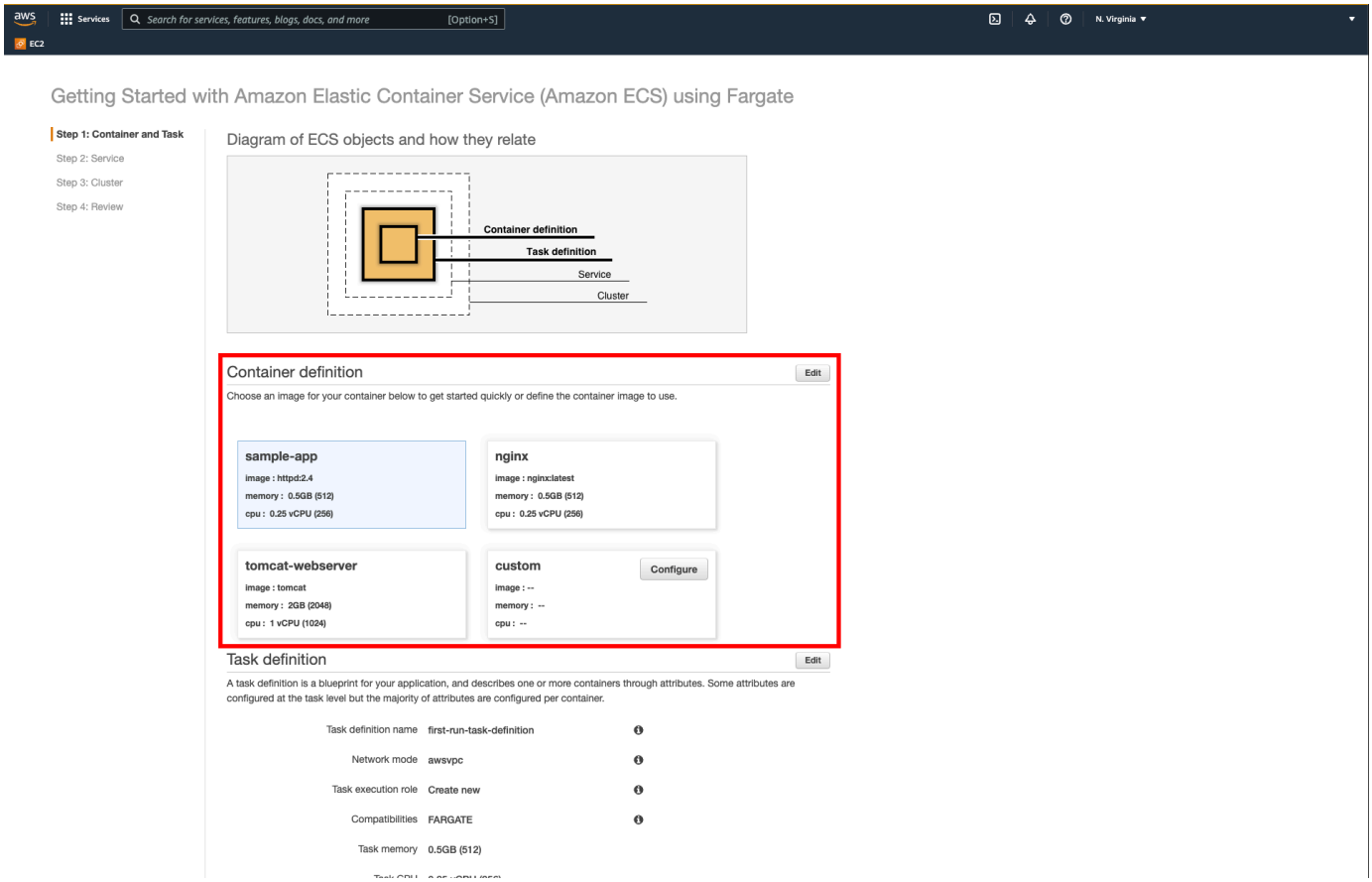


Step 2: Create container and task definition

A task definition is like a blueprint for your application. In this step, you will specify a task definition so Amazon ECS knows which Docker image to use for containers, how many containers to use in the task, and the resource allocation for each container.

1. Select a task definition

   In the **Container definition** field, select **sample-app**.

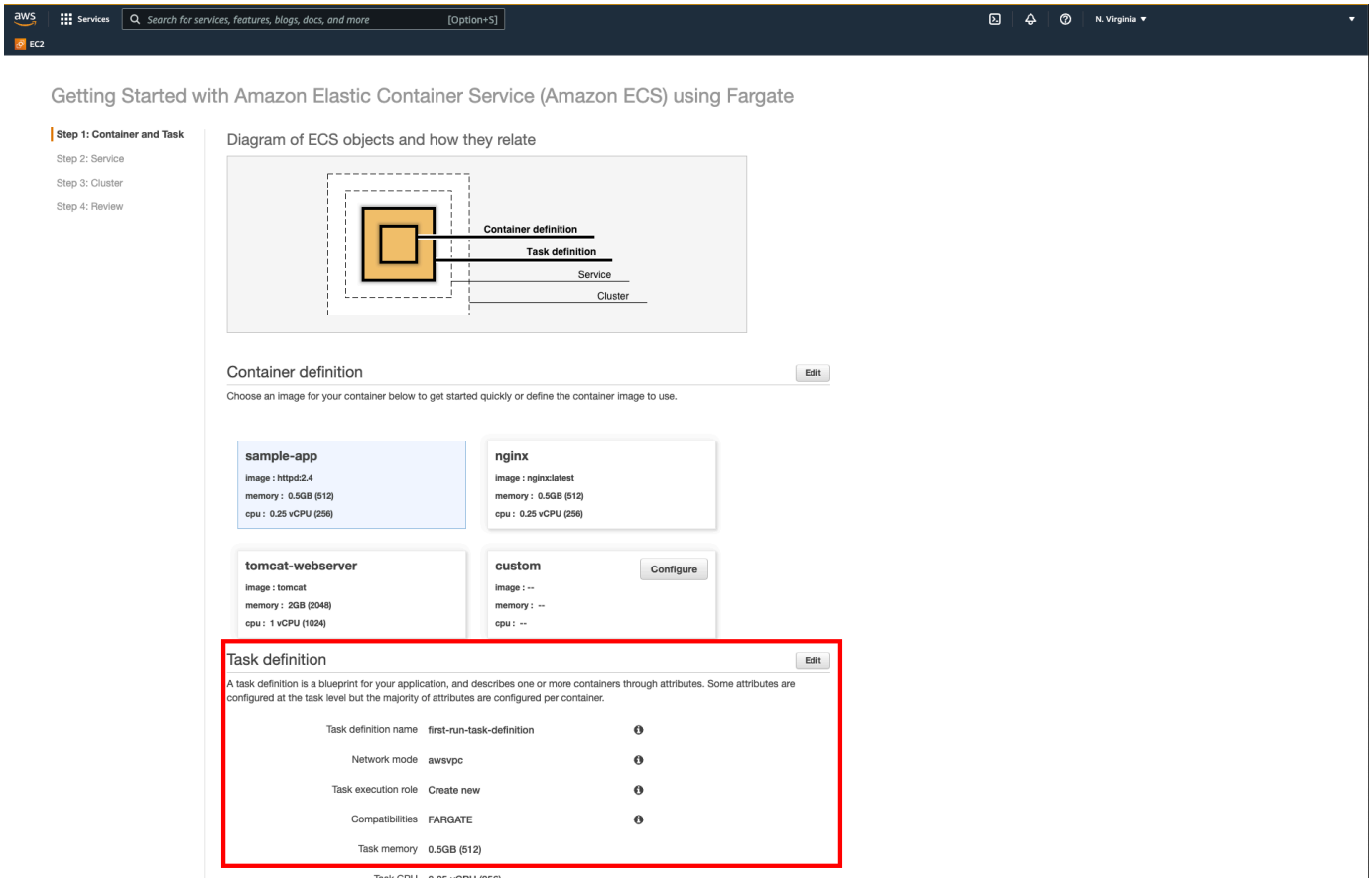


2. Review the default values

   The task definition comes preloaded with default configuration values.

   Review the default values and choose **Next.**

   If you prefer to modify the configurations or would like to learn more, see Task definition parameters.



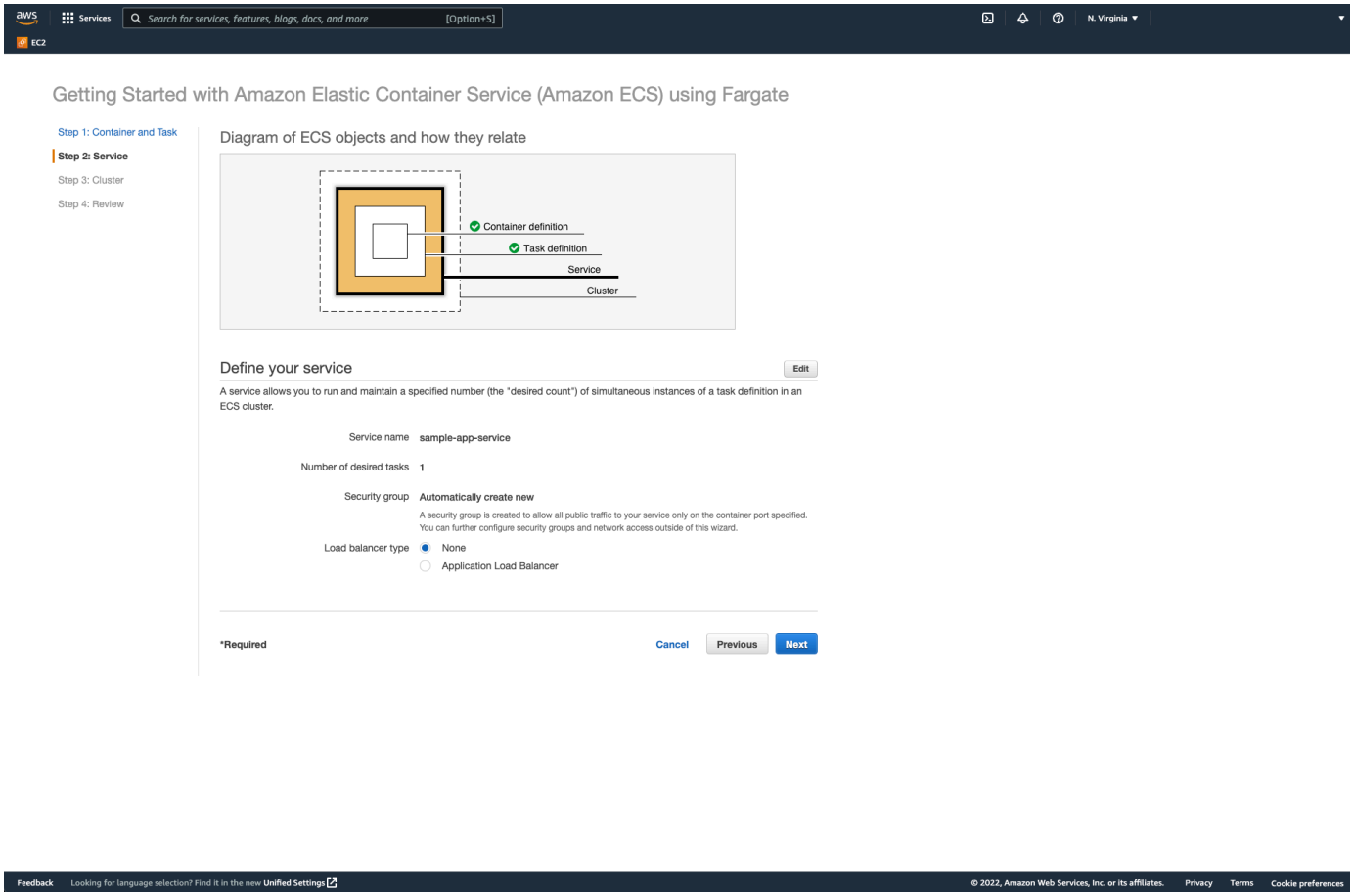
Step 3: Define your service

Now that you have created a task definition, you will configure the Amazon ECS service. A service launches and maintains copies of the task definition in your cluster. For example, by running an application as a service, Amazon ECS will auto-recover any stopped tasks and maintain the number of copies you specify.

1. Review service options

   Service options come preloaded with default configuration values.

   - **Service name**: The default **sample-app-service** is a web-based "Hello World" application provided by AWS. It is meant to run indefinitely; because it is running as a service, it will restart if the task becomes unhealthy or unexpectedly stops.
   - **Number of desired tasks**: Leave the default value of 1. This will create one copy of your task.



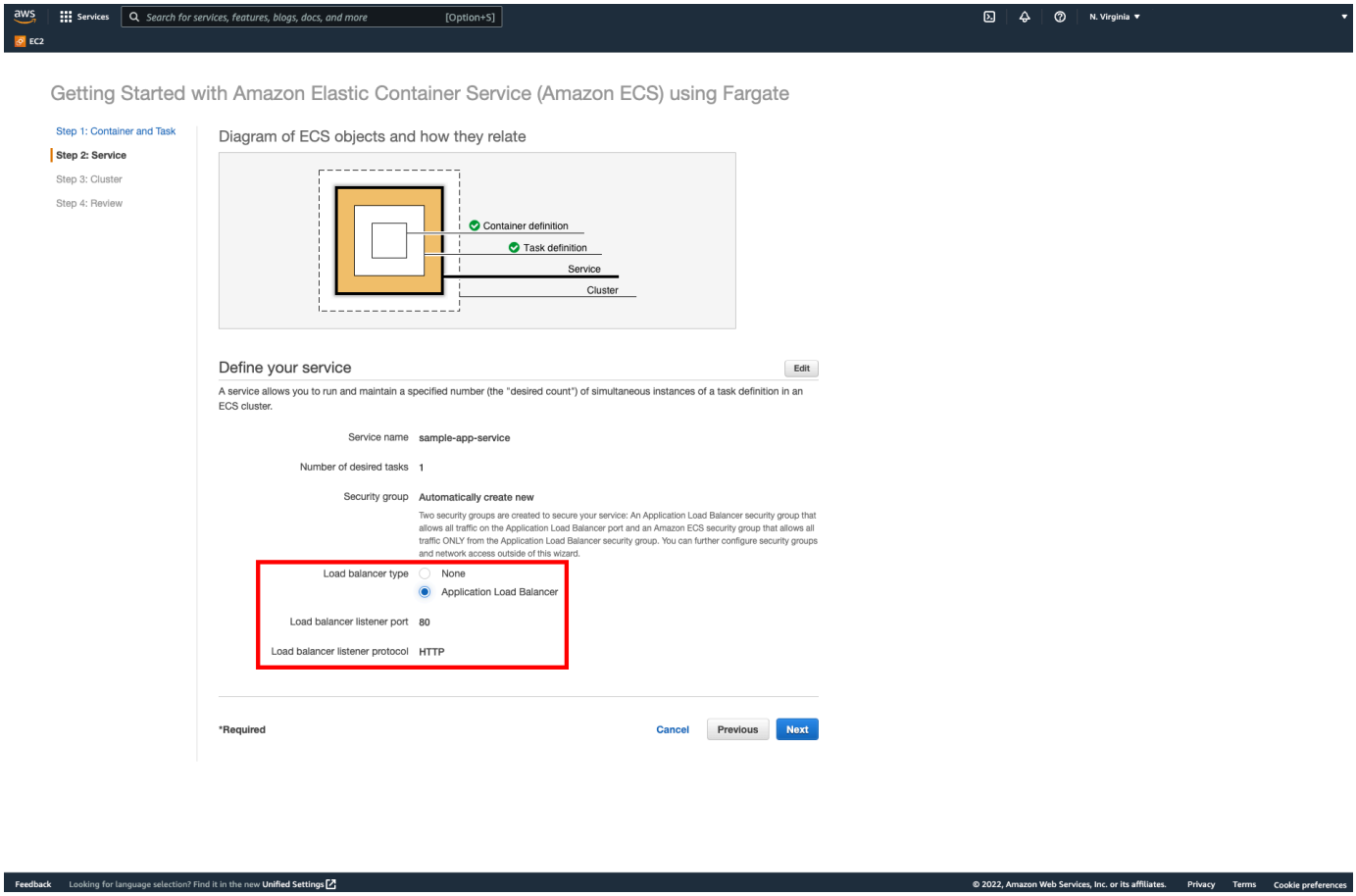
2. Review load balancing settings

   Load balancing: You have the option to use a load balancer with your service. Amazon ECS can create an Elastic Load Balancing (ELB) load balancer to distribute the traffic across the container instances your task is launched on.

   Select the **Application Load Balancer** option**.**

   The default values for **Load balancer listener port** and **Load balancer listener protocol** are set up for the sample application. For more information on load balancing configuration, see Service load balancing.

   Review your settings and choose **Next.**

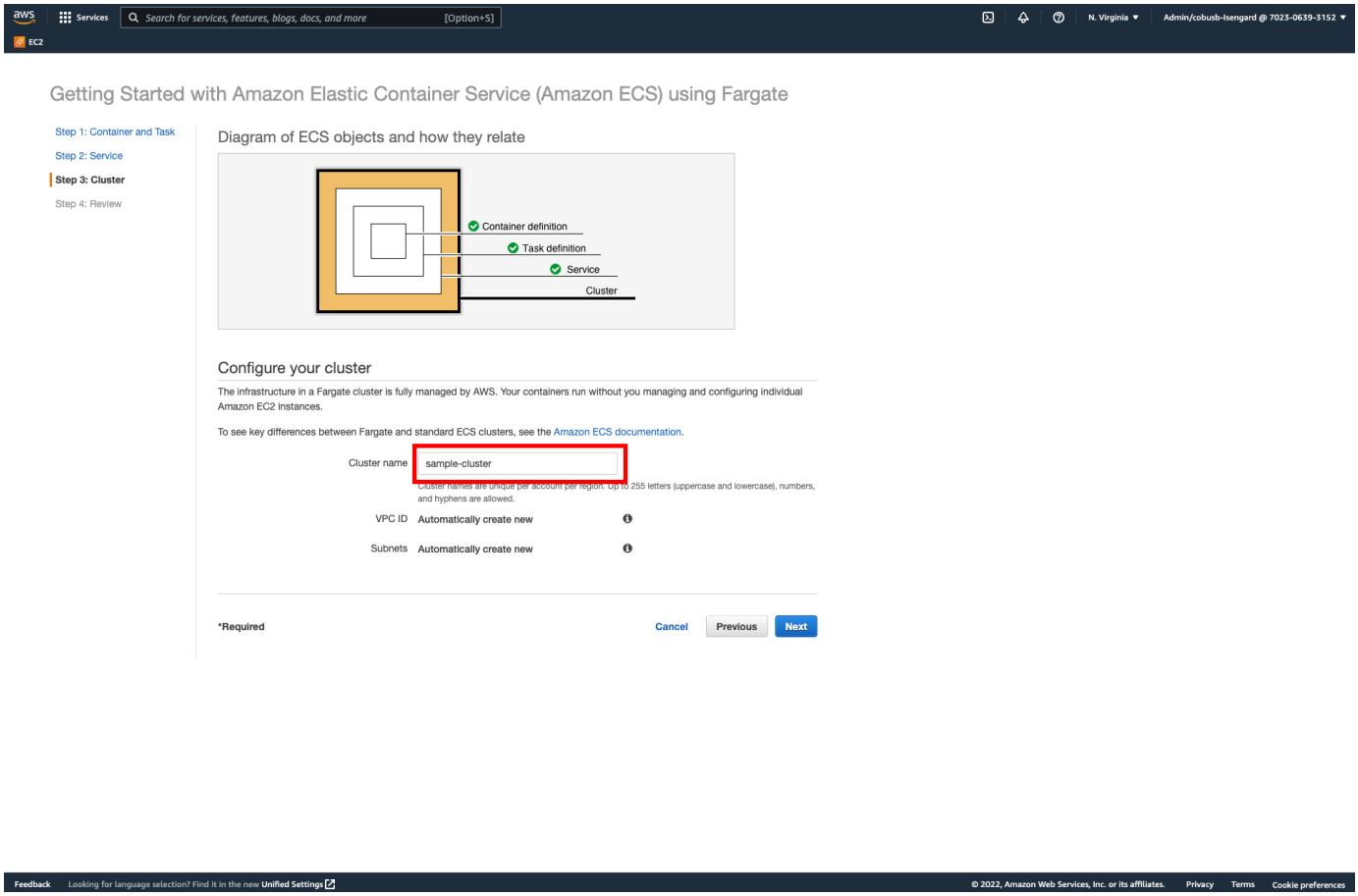


Step 4: Configure your cluster

Your Amazon ECS tasks run on a cluster, which uses AWS Fargate to provide the compute engine so that you do not need to manage servers. In this step, you will configure the cluster.

- Set cluster name

  In the **Cluster name** field, enter **sample-cluster** and choose **Next**.

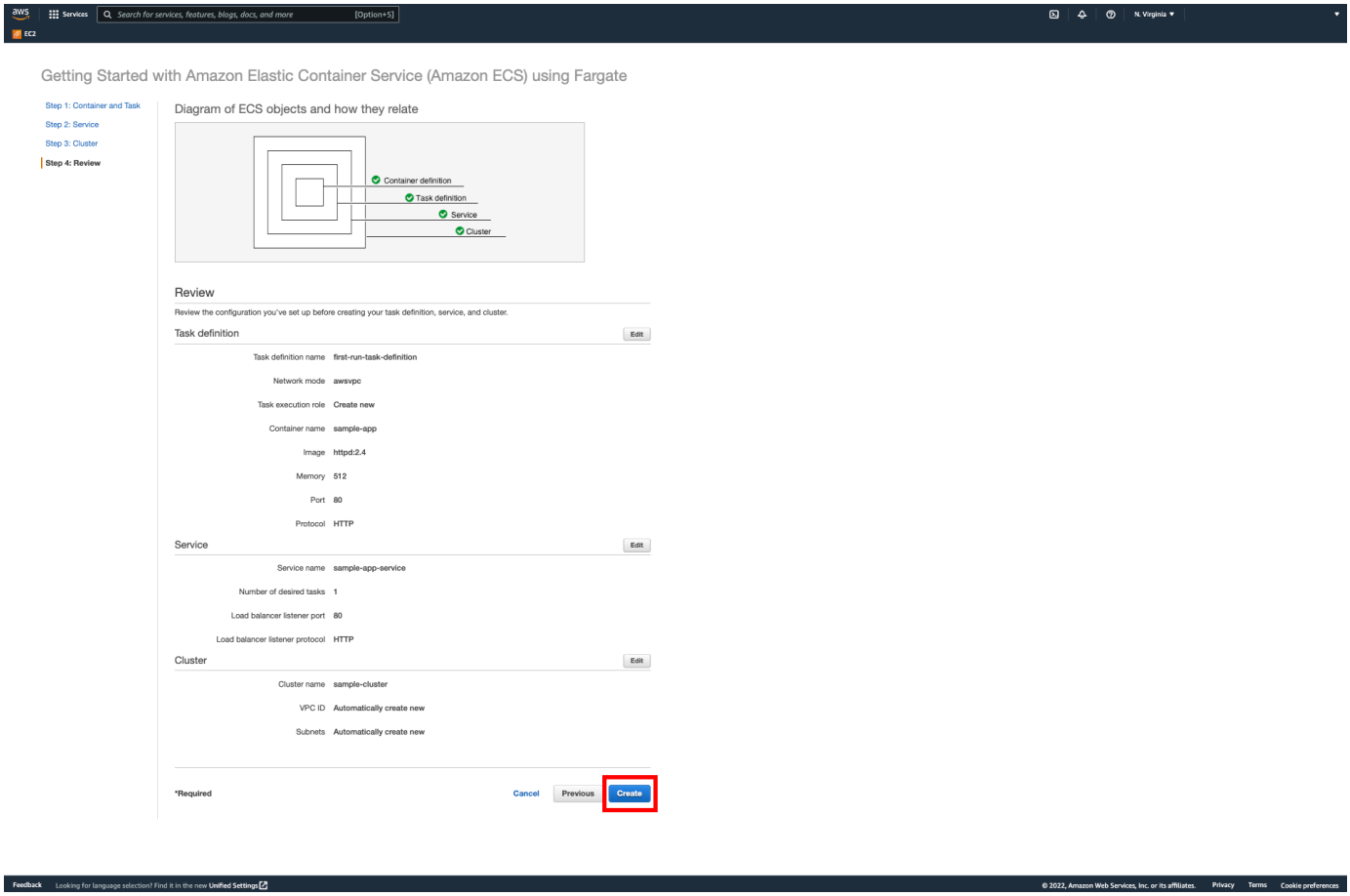


Step 5: Launch and view your resources

In the previous steps, you configured your task definition (which is like an application blueprint), the Amazon ECS service (which launches and maintains copies of your task definitions), and your cluster. In this step, you will review, launch, and view the resources you create.

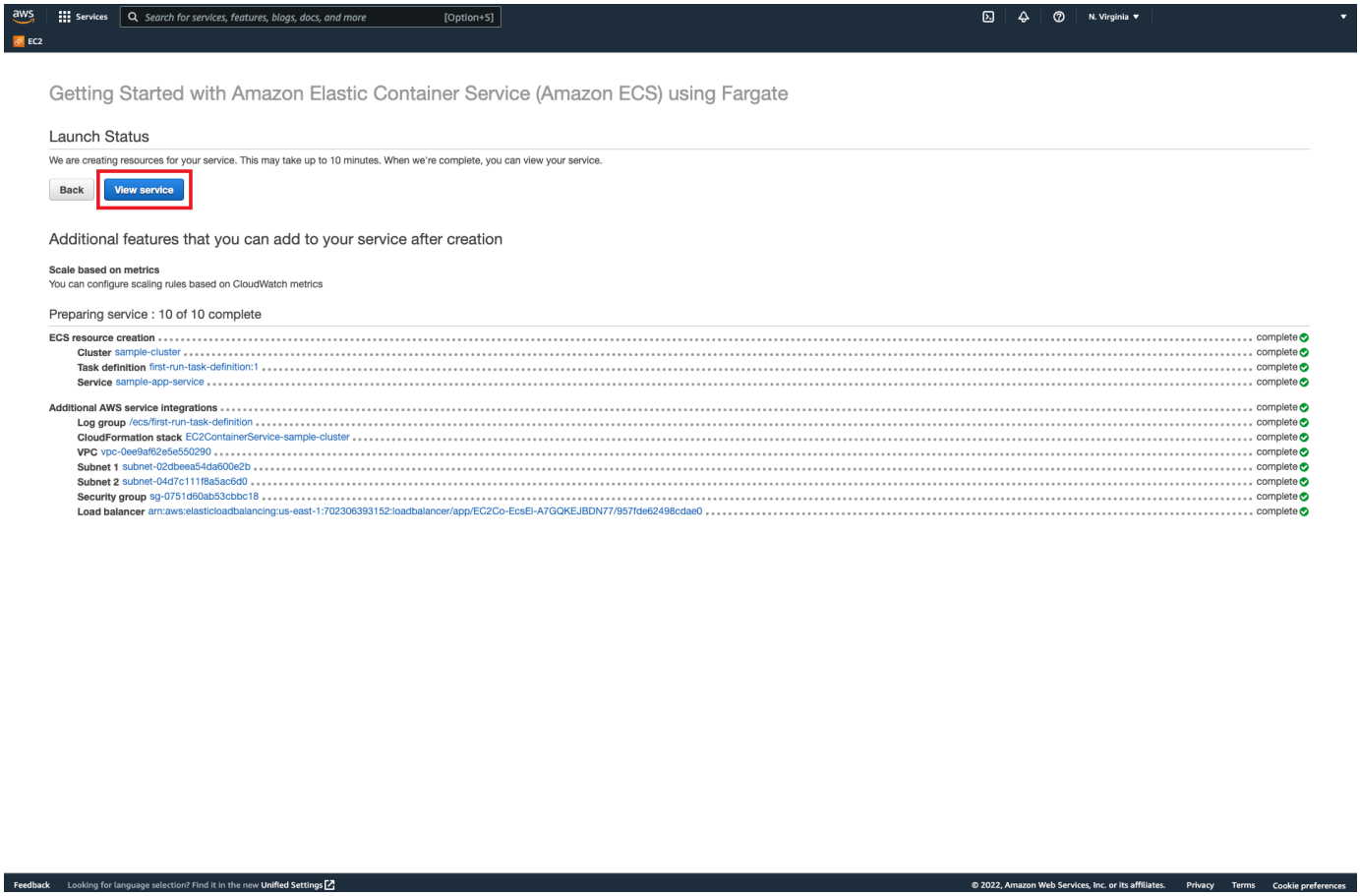
1. Review task definition

   You have a final chance to review your task definition, task configuration, and cluster configuration before launching. Choose **Create.**



2. View service status

   You are on a **Launch Status** page that shows the status of your launch and describes each step of the process. After the launch is complete, choose **View service.**

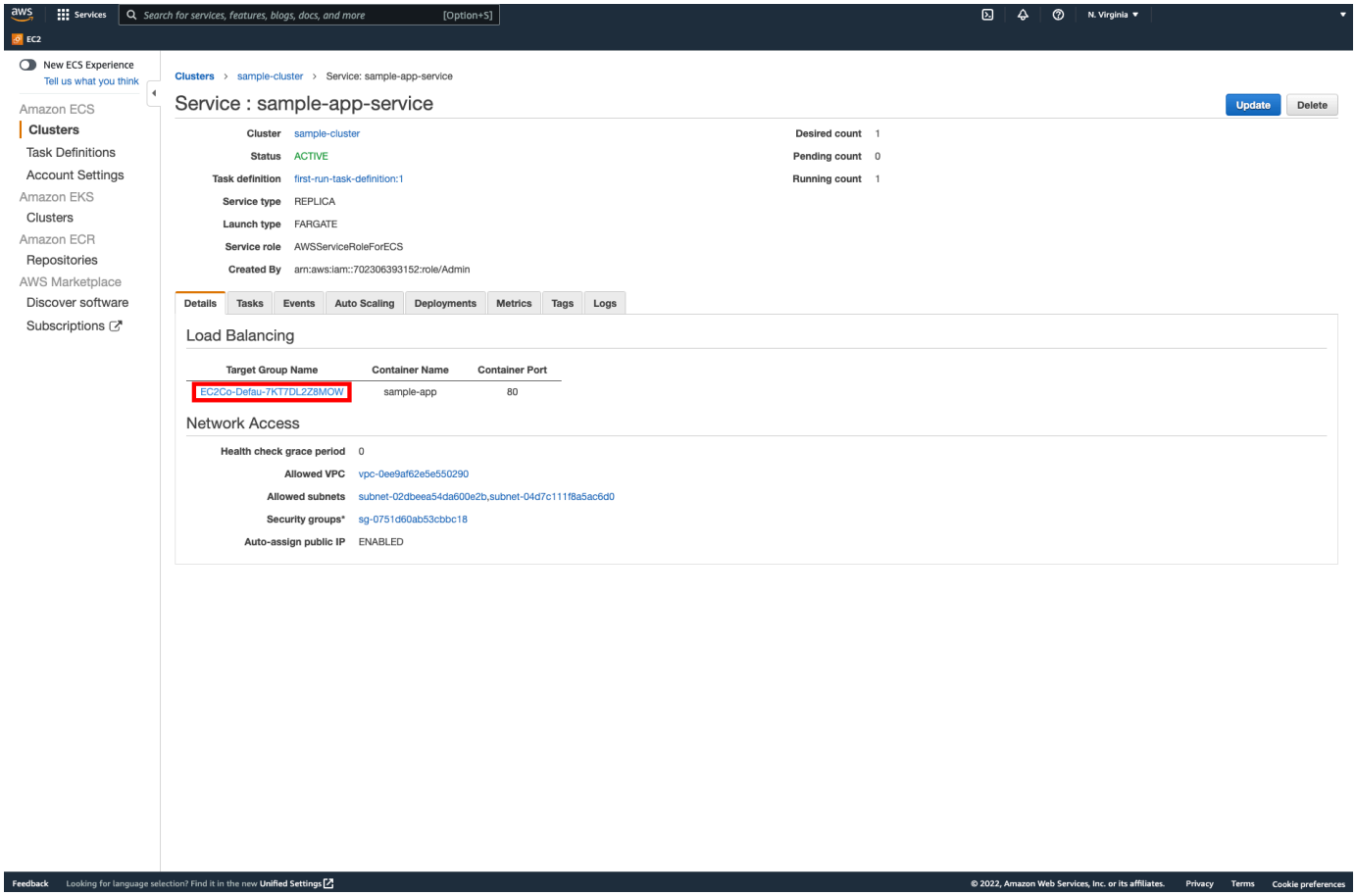


Step 6: Open the sample application

In this step, you will verify that the sample application is up and running by pointing your browser to the load balancer DNS name.

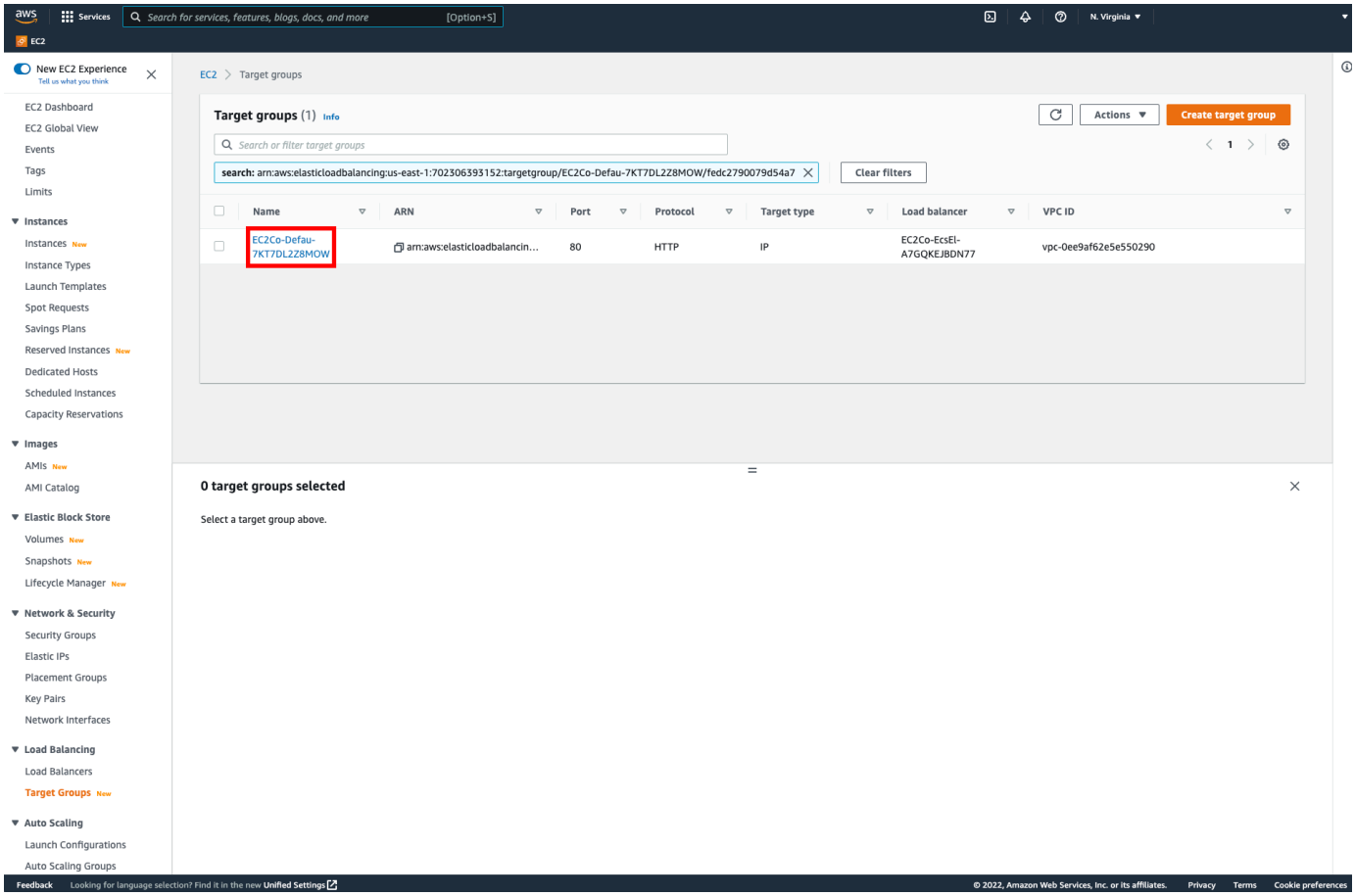
1. View details about the application

   On the sample-app-service page, select the **Details** tab and select the entry under **Target Group Name.**



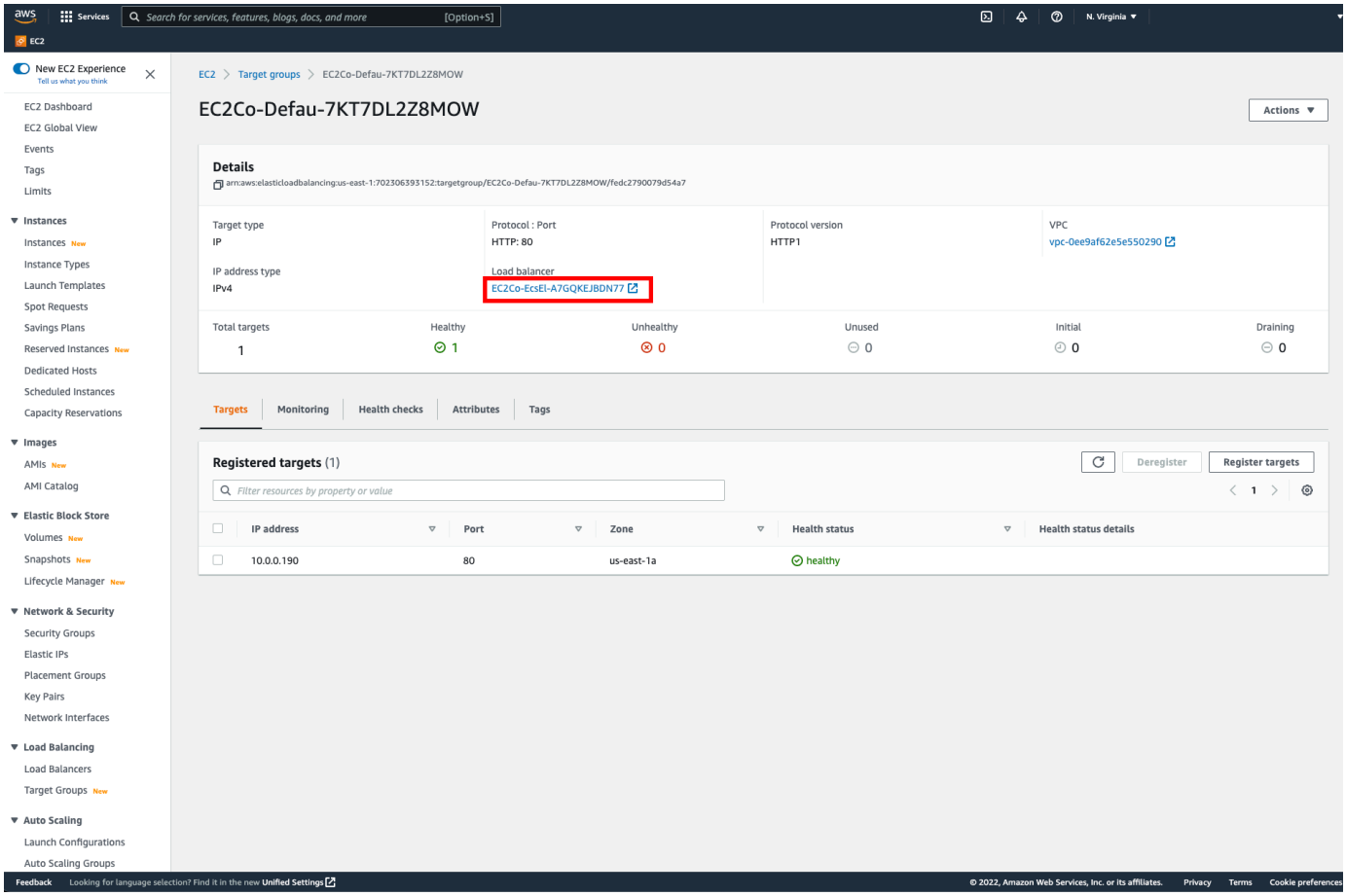
2. View target group details

   On the **Target groups** page, select the target group name.



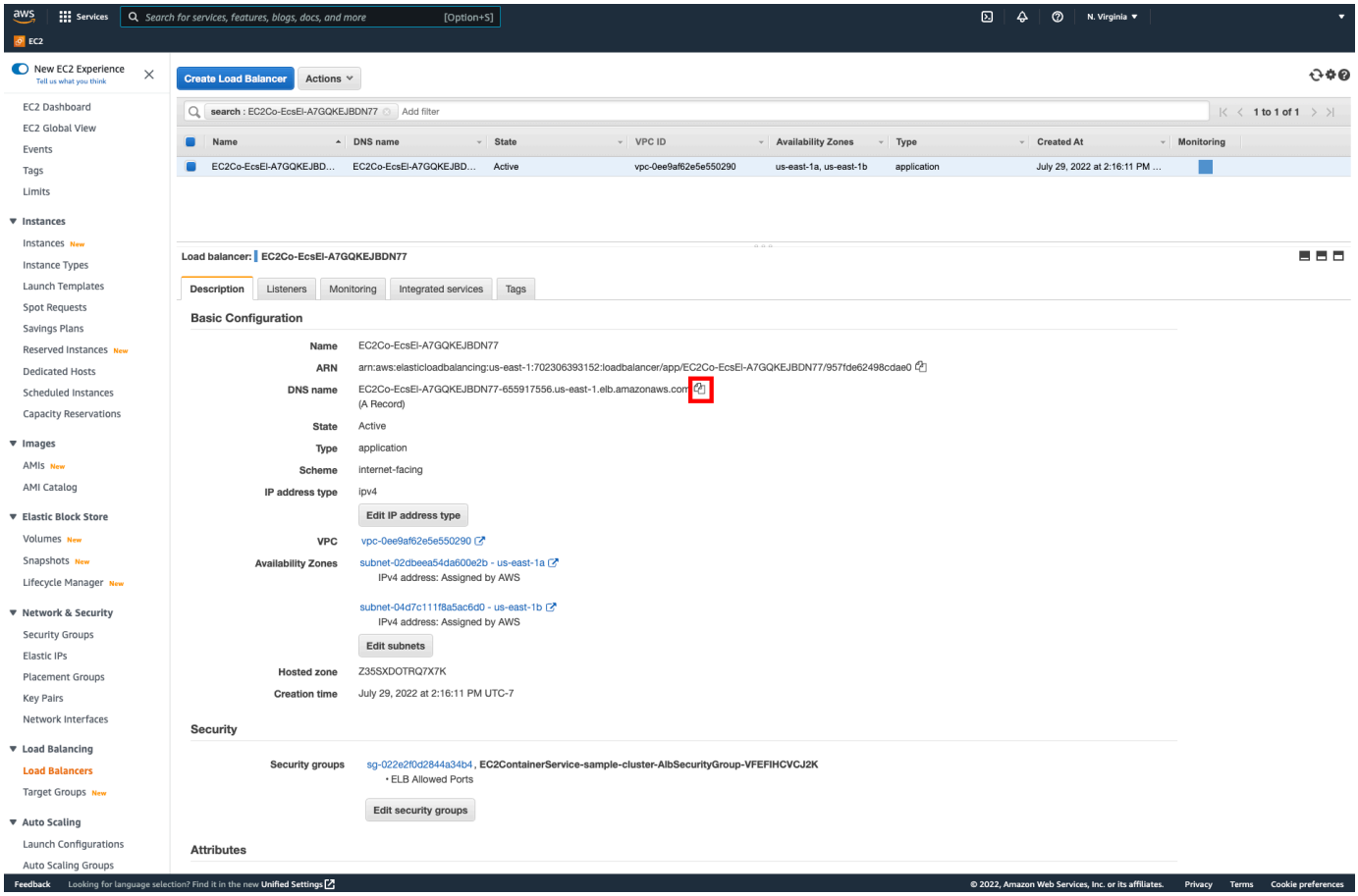
3. Select the load balancer

   In the **Details** section, choose the **Load balancer** link.



4. Copy the DNS name of the application

   In the **Description** tab, select the two page icon next to the load balancer DNS to copy the DNS name to your clipboard.



5. View the sample application

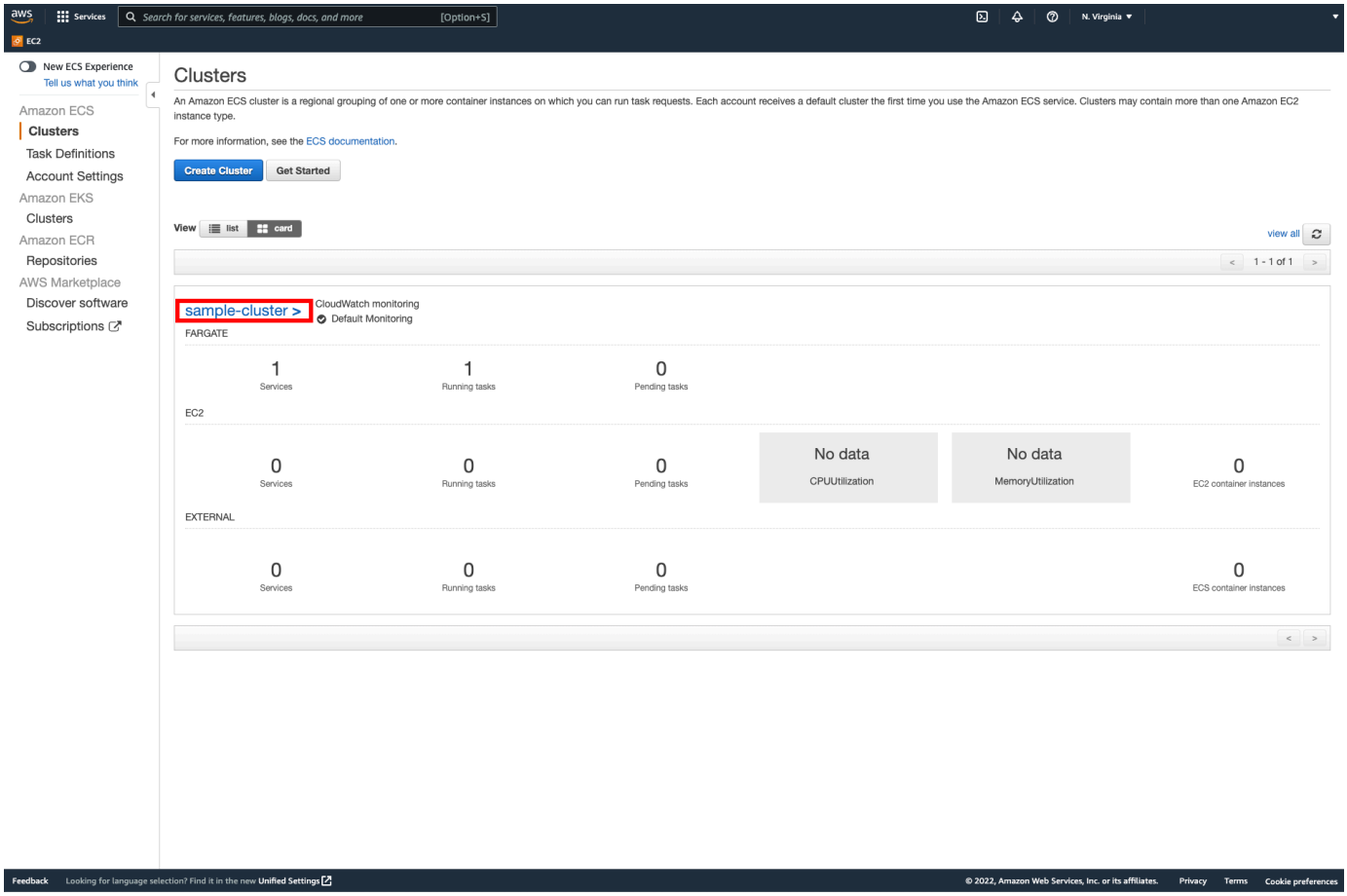
Paste the name into a new browser window, and press Enter to view the sample application (in this case, a static webpage).

(Optional) Clean up resources

Throughout this guide, you've launched three resources: an Amazon ECS cluster, AWS Fargate to run your container, and a load balancer. In this step, you can clean up all your resources to avoid unwanted charges.

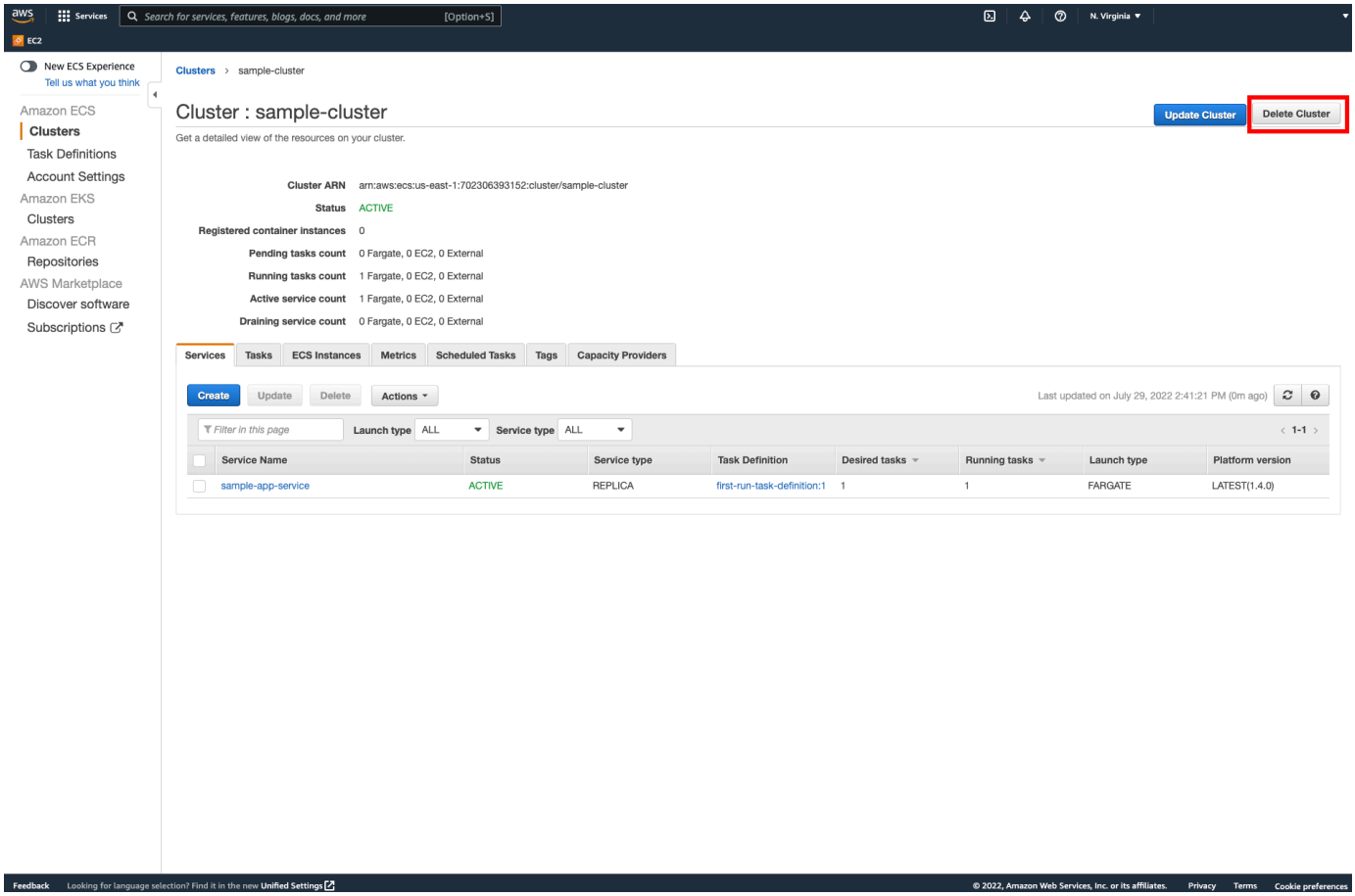
1. Select the cluster

   Navigate back to the Amazon ECS console page and select the cluster name (sample-cluster).



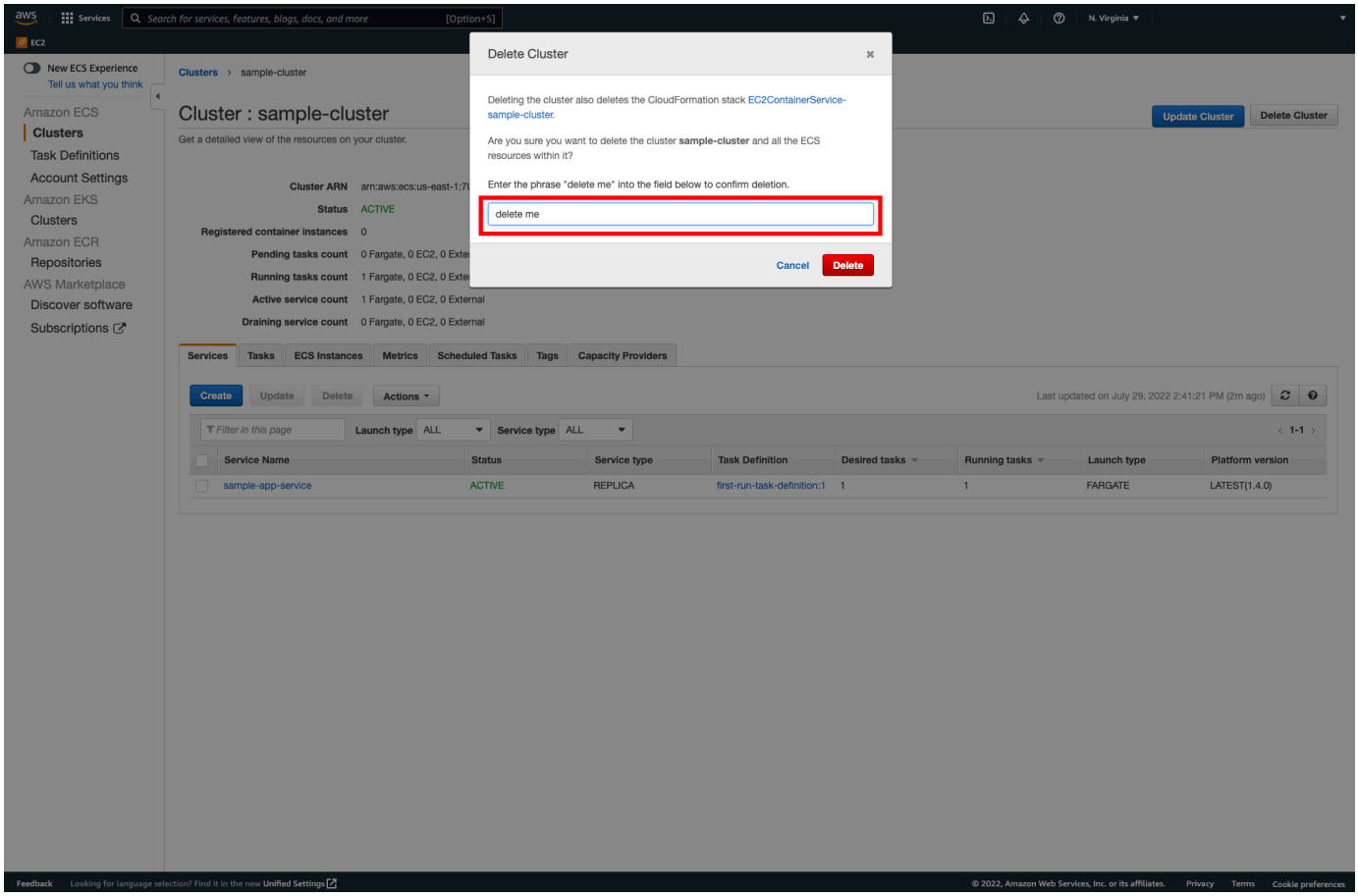
2. Choose Delete Cluster

   Choose **Delete Cluster** to delete the cluster.



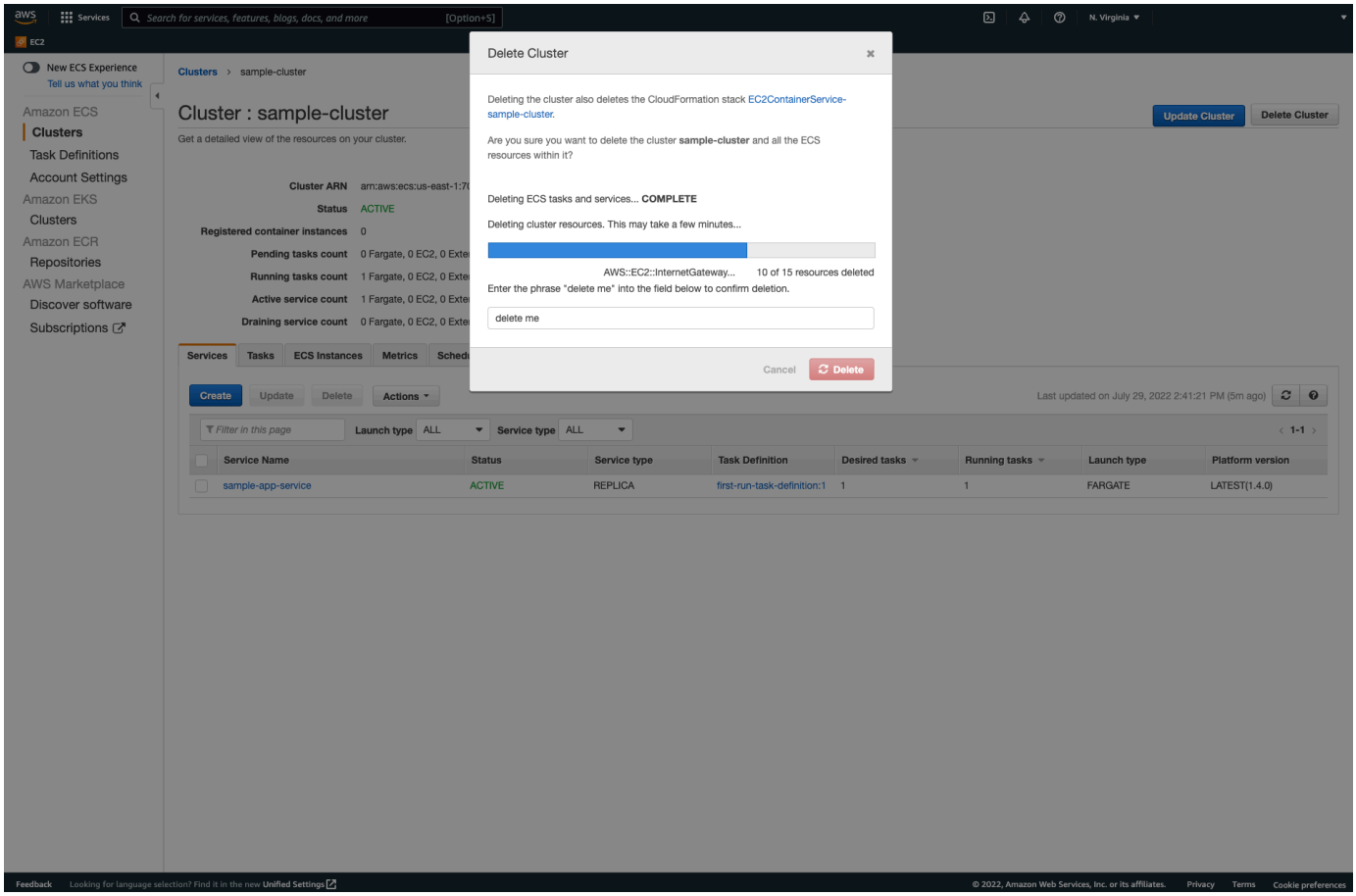
3. Confirm cluster deletion

   Enter **delete me** in the dialog box and choose **Delete**.



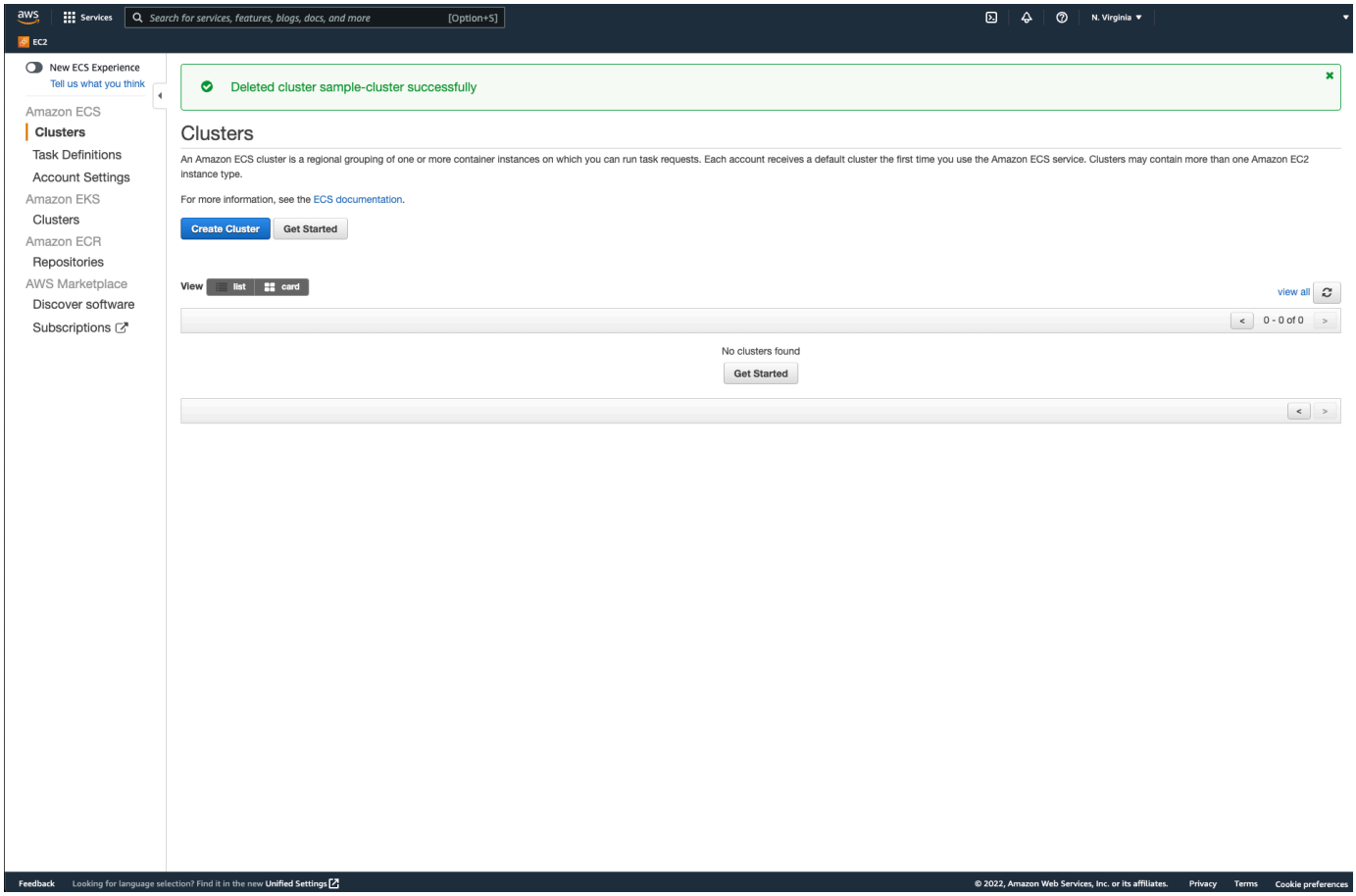
4. Monitor cluster deletion

   You will now see the progress as all the resources created are deleted.



5. Cluster deletion complete

   Once everything has been deleted, you will see the **Deleted cluster sample-cluster successfully** message in green. You have now completed this guide.



Congratulations

Congratulations! You have learned how to configure and deploy your Docker-enabled application to Amazon ECS, and how to delete resources that are no longer needed. Amazon ECS is a highly scalable, high performance container management service that supports Docker containers and allows you to easily run applications on a managed cluster of Amazon EC2 instances.