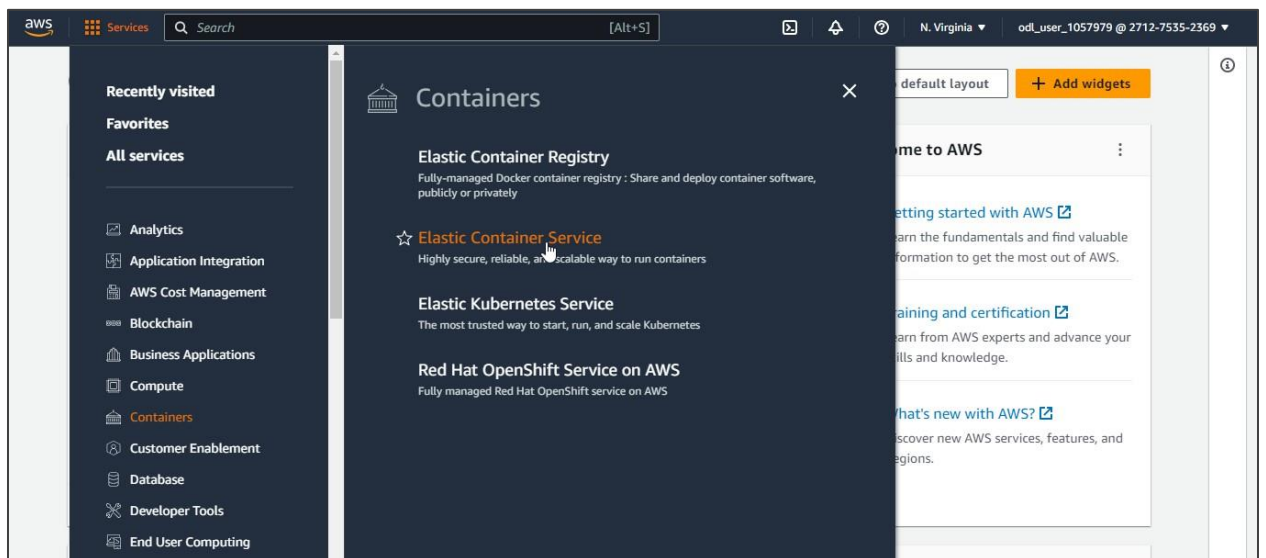


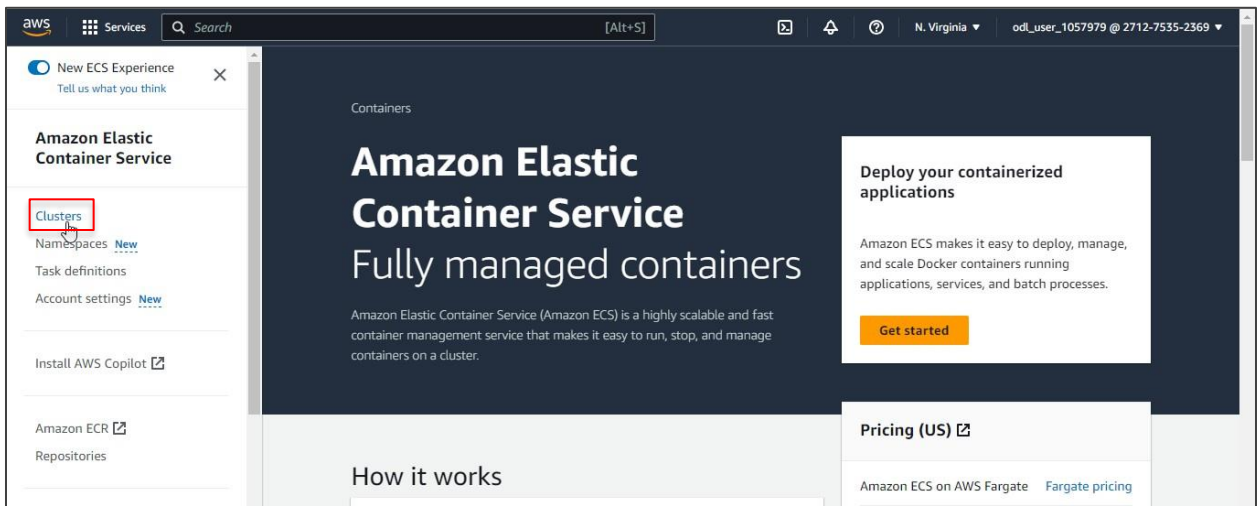
ECS Cluster

Step 1: Create an ECS cluster

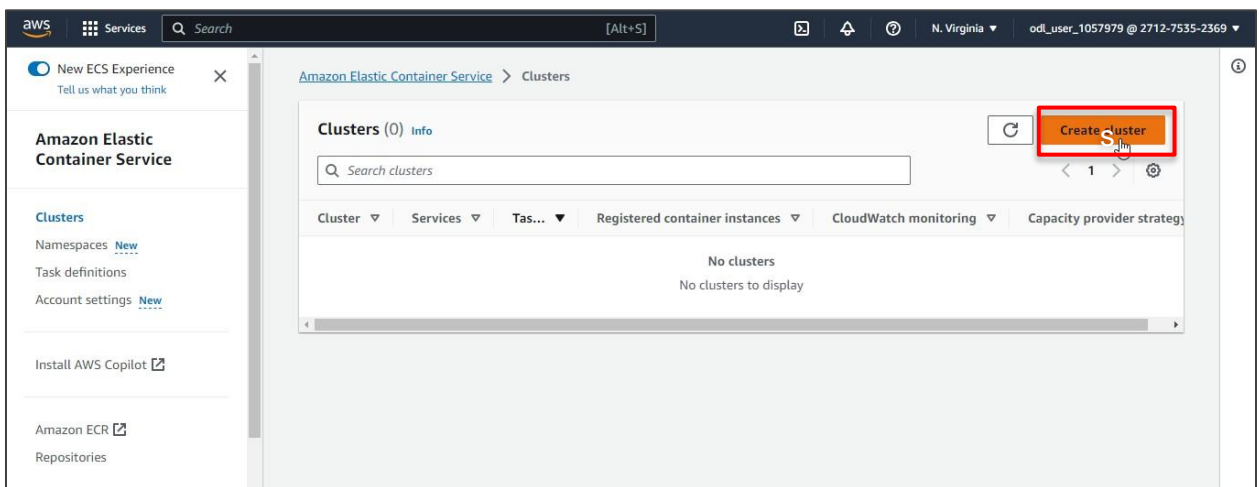
1.1 Navigate to the AWS Console, select **Services** and then choose **ECS** under **Containers**



1.2 Click on **Clusters** in the navigation pane



1.3 Click **Create cluster**



1.4 Create an ECS cluster per the settings shown in the screenshots:

This screenshot shows the 'Create cluster' page in the AWS Management Console. The left sidebar contains the 'Amazon Elastic Container Service' menu with options like Clusters, Namespaces, Task definitions, and Account settings. The main content area is titled 'Create cluster' and includes a description of ECS clusters. The 'Cluster configuration' section has a 'Cluster name' field set to 'MyCluster' and a 'Default namespace' dropdown also set to 'MyCluster'. Below this, the 'Infrastructure' section is expanded, showing that the cluster is configured for AWS Fargate (serverless) with two capacity providers. A 'Customized' button is visible at the bottom of the infrastructure section.

aws Services Search [Alt+S] N. Virginia odl_user_1057979 @ 2712-7535-2369

New ECS Experience Tell us what you think

Amazon Elastic Container Service

Clusters
Namespaces New
Task definitions
Account settings New

Install AWS Copilot

Amazon ECR Repositories

AWS Batch

Amazon Elastic Container Service > Create cluster

Create cluster

An Amazon ECS cluster groups together tasks, and services, and allows for shared capacity and common configurations. All of your tasks, services, and capacity must belong to a cluster.

Cluster configuration

Cluster name
MyCluster
There can be a maximum of 255 characters. The valid characters are letters (uppercase and lowercase), numbers, hyphens, and underscores.

Default namespace - optional
Select the namespace to specify a group of services that make up your application. You can overwrite this value at the service level.
MyCluster

Infrastructure
Your cluster is automatically configured for AWS Fargate (serverless) with two capacity providers. Add Amazon EC2 instances, or external instances using ECS Anywhere.
Customized

CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

This screenshot shows the 'Provisioning model' section of the 'Create cluster' page. It allows the user to select between 'AWS Fargate (serverless)' and 'Amazon EC2 instances'. 'Amazon EC2 instances' is selected. Below this, the 'Auto Scaling group (ASG)' is set to 'Create new ASG'. The 'Provisioning model' section has two options: 'On-demand' and 'Spot'. 'Spot' is selected, and a callout box provides details about Spot instances. The 'Allocation strategy' is set to 'Price capacity optimized (recommended)'. The 'Container instance Amazon Machine Image (AMI)' section is partially visible at the bottom.

Amazon Elastic Container Service

Clusters
Namespaces
Task definitions
Account settings

Install AWS Copilot

Amazon ECR Repositories

AWS Batch

Documentation Discover products

☐ AWS Fargate (serverless)
Pay as you go. Use if you have tiny, batch, or burst workloads or for zero maintenance overhead. The cluster has Fargate and Fargate Spot capacity providers by default.

☒ Amazon EC2 instances
Manual configurations. Use for large workloads with consistent resource demands.

Auto Scaling group (ASG)
Use Auto Scaling groups to scale the Amazon EC2 instances in the cluster.
Create new ASG

Provisioning model
Select a provisioning model for your instances

☐ On-demand
With on-demand instances, you pay for compute capacity by the hour, with no long-term commitments or upfront payments.

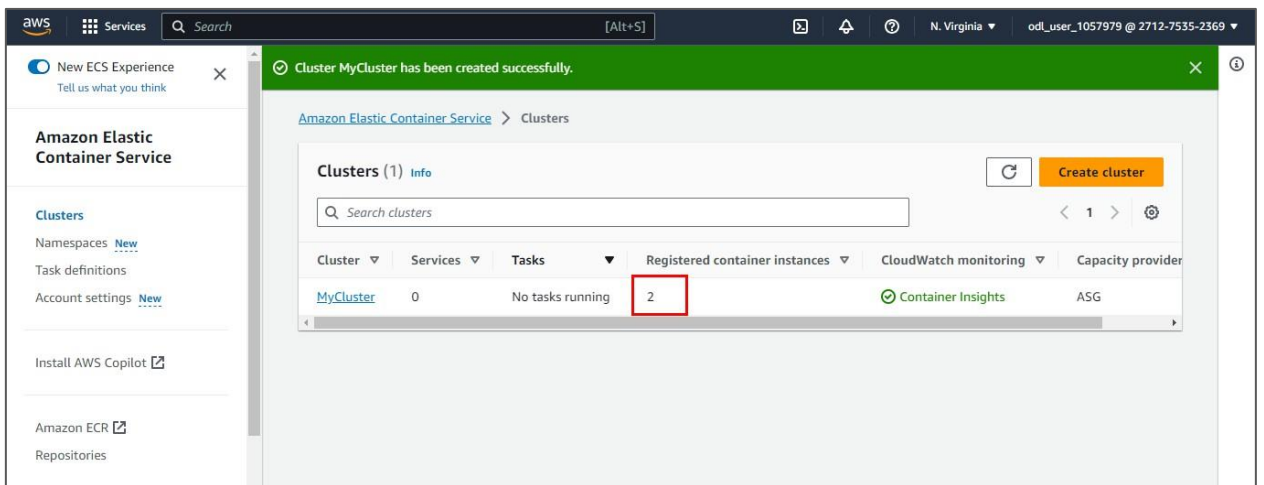
☒ Spot
Amazon EC2 Spot instances let you take advantage of unused EC2 capacity in the AWS cloud. Spot instances are available at up to a 90% discount compared to on-demand prices.

Allocation strategy
Choose the spot instance allocation strategy.
Price capacity optimized (recommended)

Container instance Amazon Machine Image (AMI)
Choose the Amazon ECS-optimized AMI for your instance.

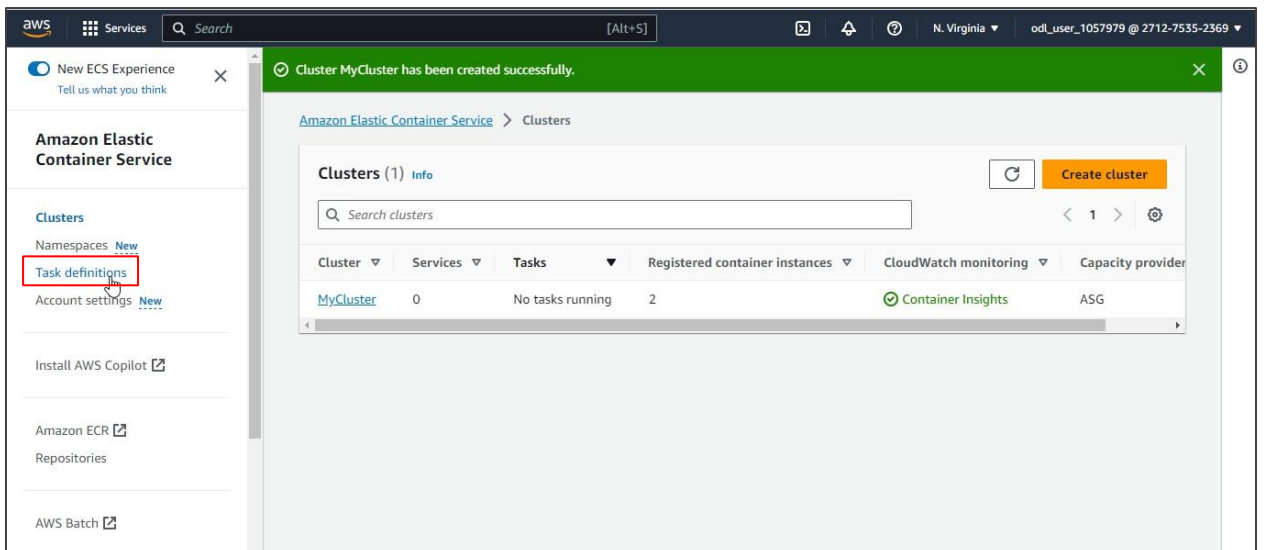
1.5 Click **Create**

Note: Wait a few minutes for the EC2 instances to register with the cluster.

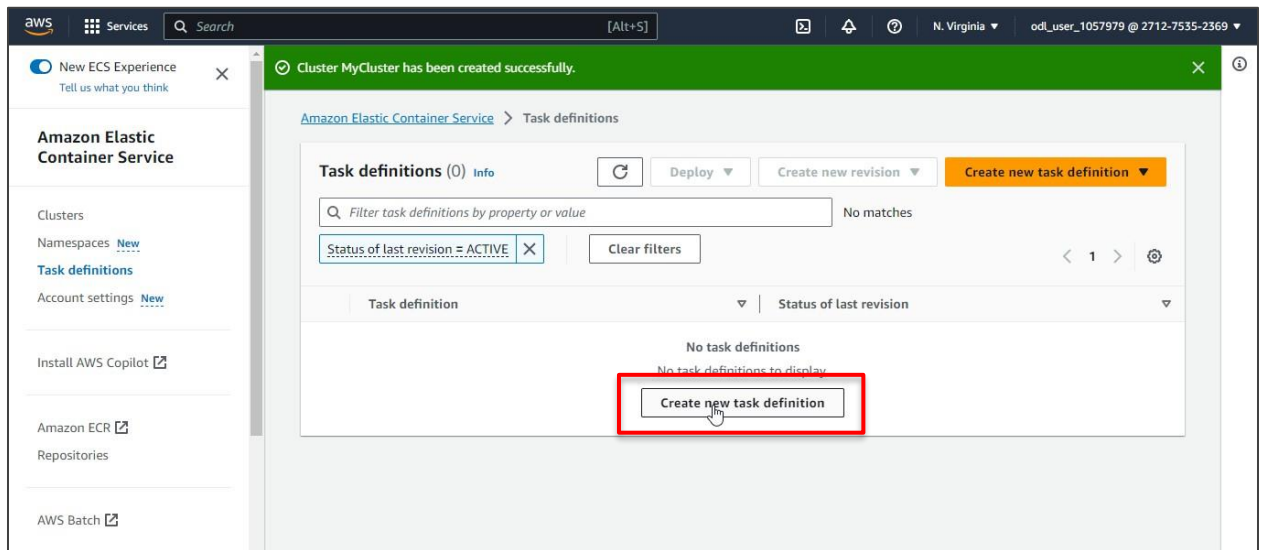


Step 2: Create a task definition

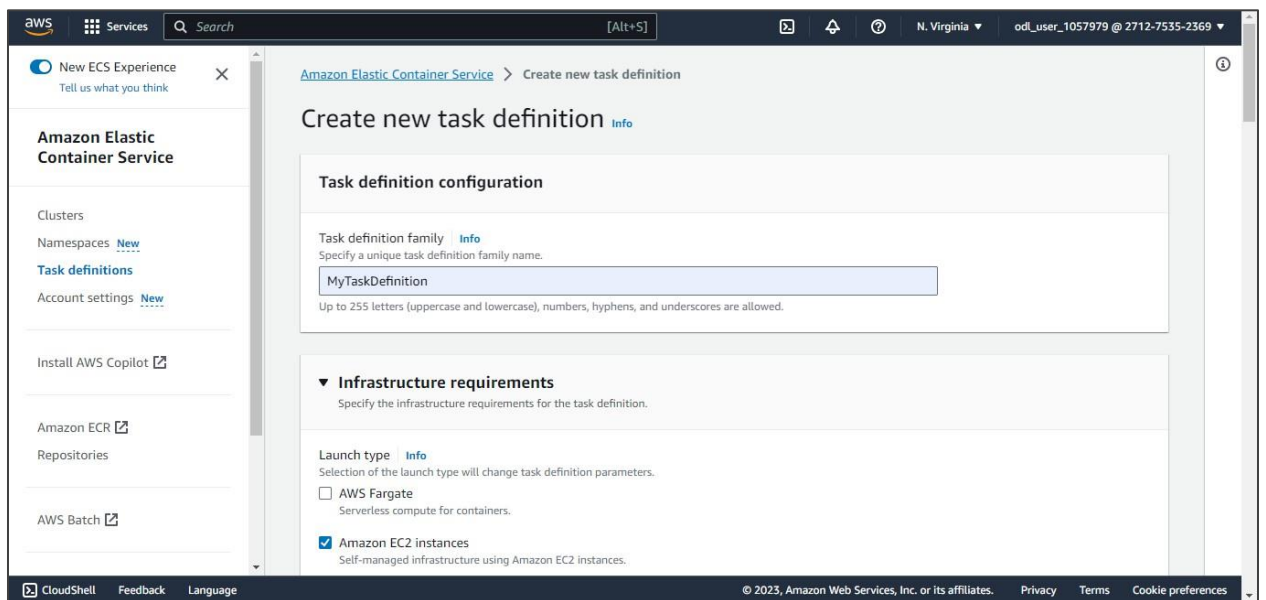
2.1 In the navigation panel, click **Task definitions**



2.2 Click Create new task definition



2.3 Create the task definition per the settings shown in the screenshots:

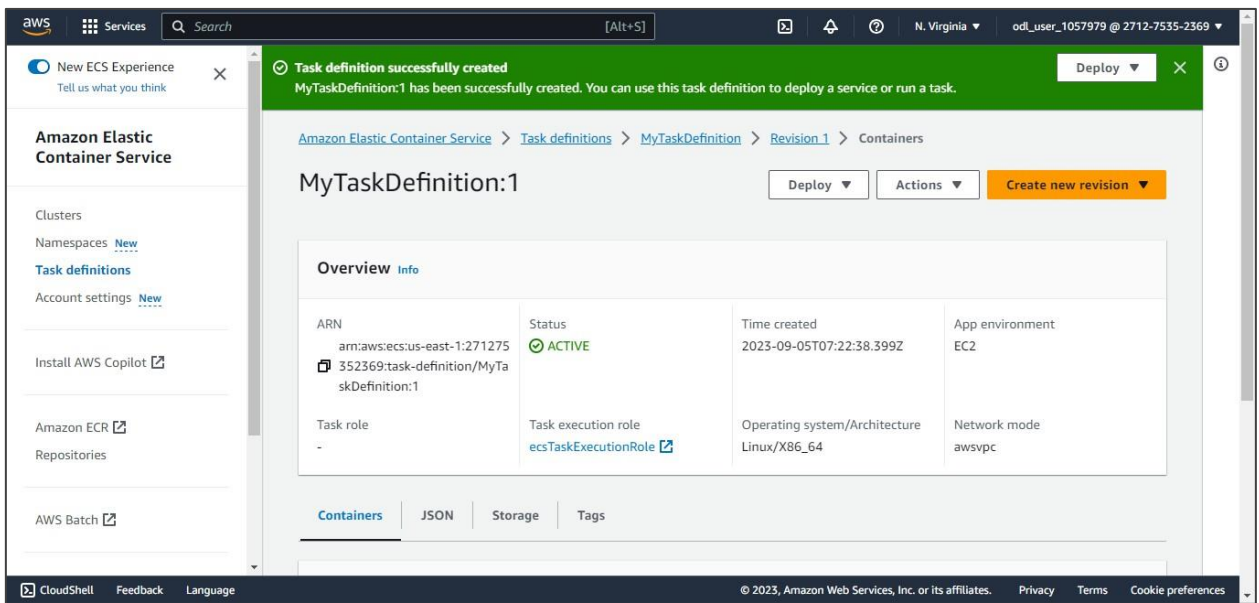


The screenshot shows the AWS ECS console interface for configuring a container. The left sidebar contains navigation links for Clusters, Namespaces, Task definitions, Account settings, Install AWS Copilot, Amazon ECR, Repositories, and AWS Batch. The main content area is titled 'Container - 1' and includes an 'Info' icon. It features a 'Container details' section with a 'Name' field (value: 'container'), an 'Image URI' field (value: 'nginx:latest'), and an 'Essential container' dropdown (value: 'Yes'). Below this is a 'Private registry' section with a 'Private registry authentication' checkbox. The 'Port mappings' section includes a table with columns for Container port, Protocol, Port name, and App protocol. The table has one row with values: 80, TCP, container-80-tcp, and HTTP. A 'Read only root file system' checkbox is also present. At the top right, there are buttons for 'Essential container' and 'Remove'.

Container port	Protocol	Port name	App protocol
80	TCP	container-80-tcp	HTTP

2.4 Click Create

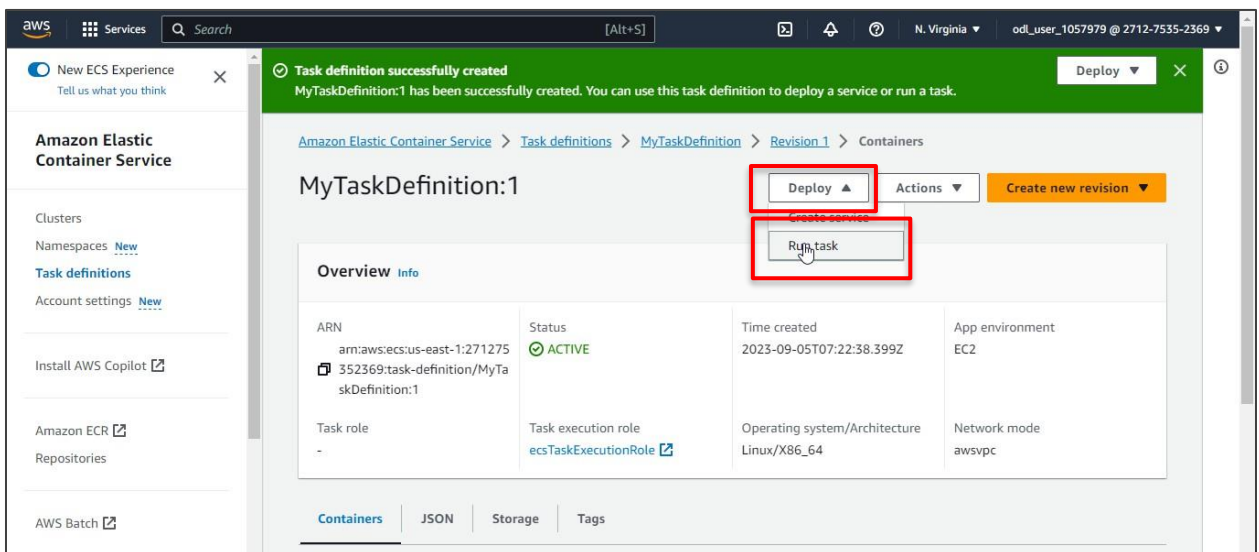
The screenshot shows the AWS ECS console interface for configuring storage and monitoring. The left sidebar is the same as the previous screenshot. The main content area is titled 'Storage - optional' and includes a 'Volumes' section with an 'Add volume' button. Below this is a 'Monitoring - optional' section with a 'Configure your application trace and metric collection settings using the AWS Distro for OpenTelemetry integration.' button. At the bottom right, there are 'Cancel' and 'Create' buttons. The 'Create' button is highlighted with a red box.



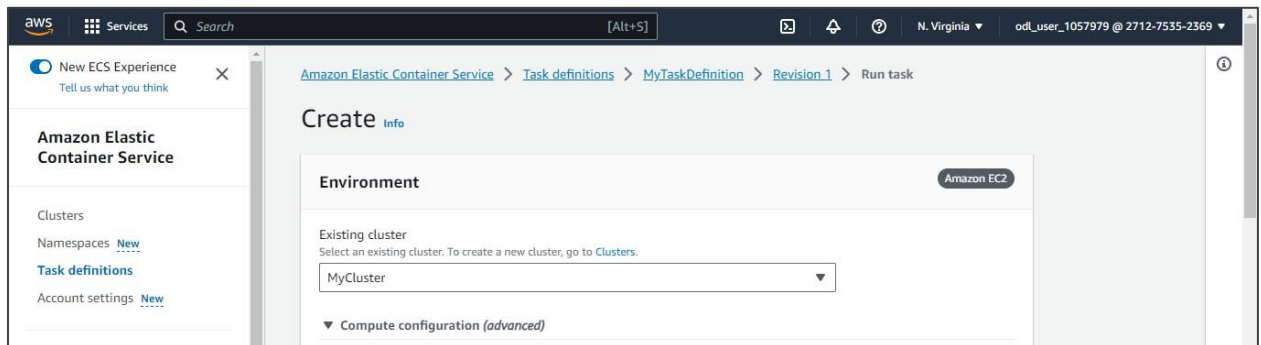
The task definition has been created successfully.

Step 3: Run the task definition on the cluster

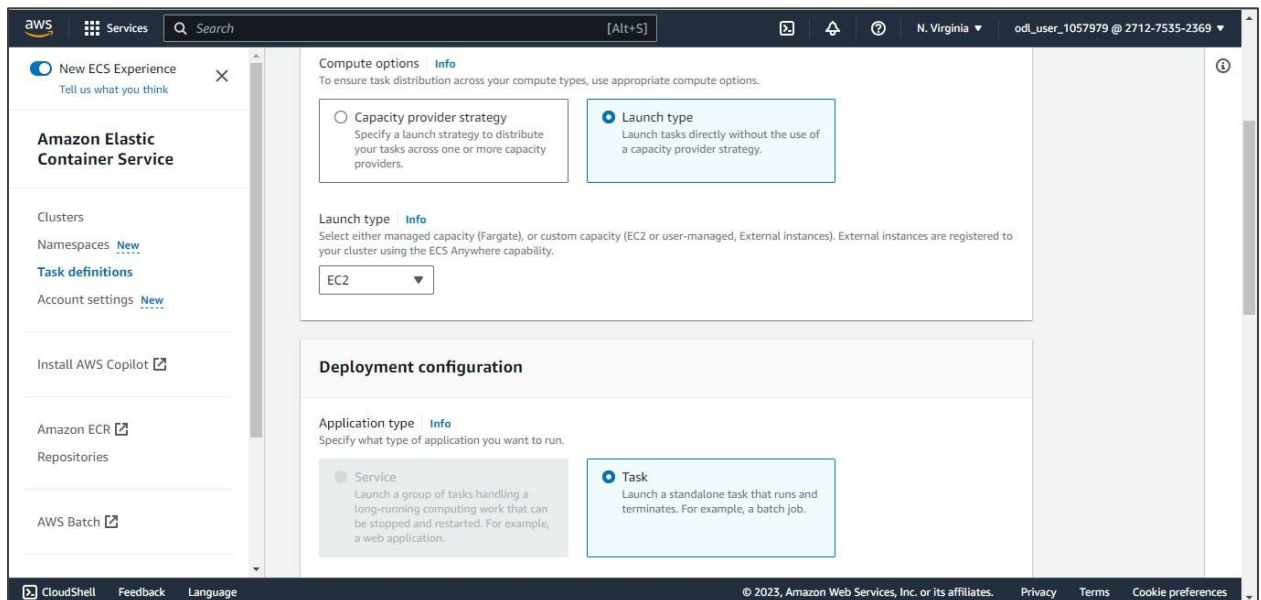
1.1 Select **Deploy** and click **Run task** to run the task definition

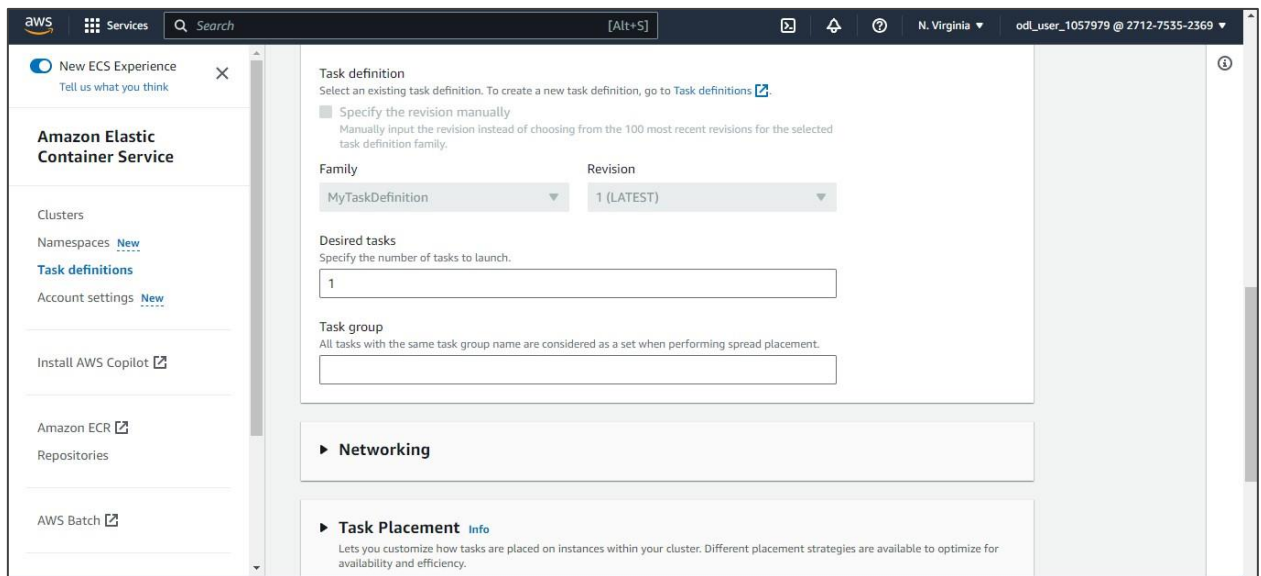


1.2 Choose the cluster that you created in the previous step

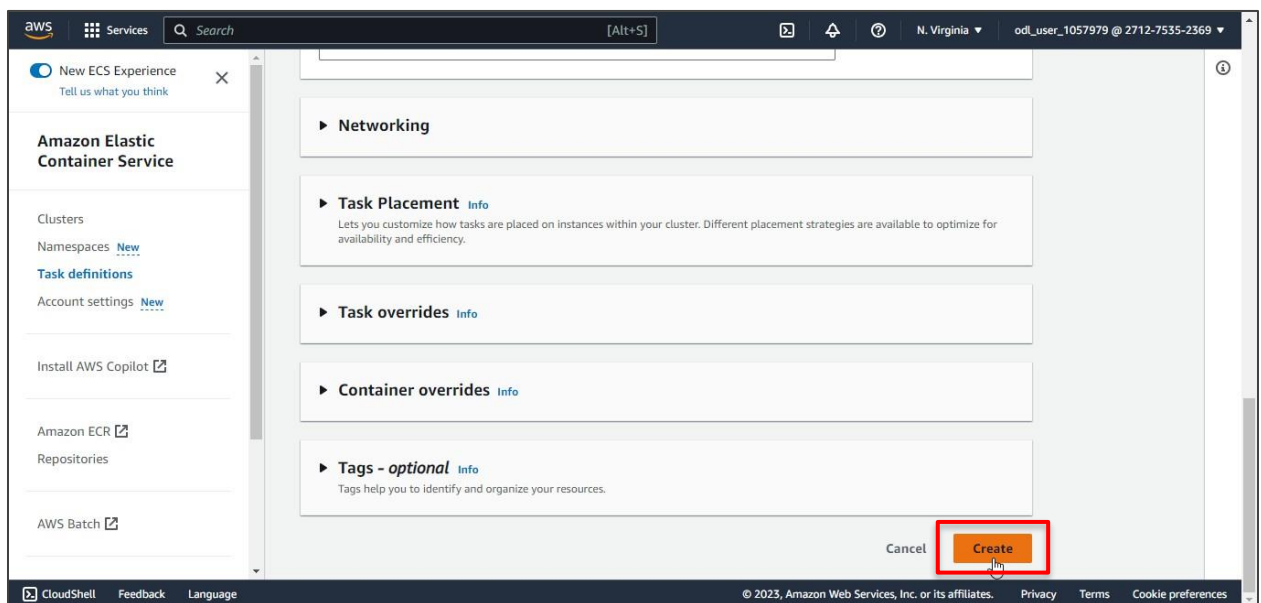


1.3 Configure the other settings per the screenshots:

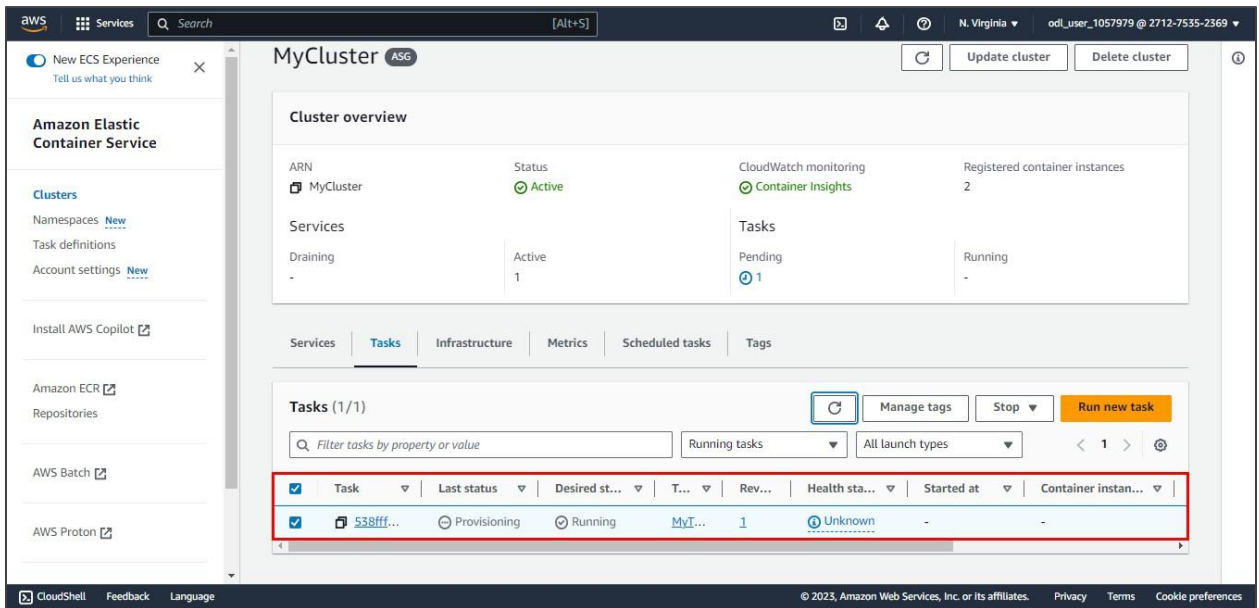
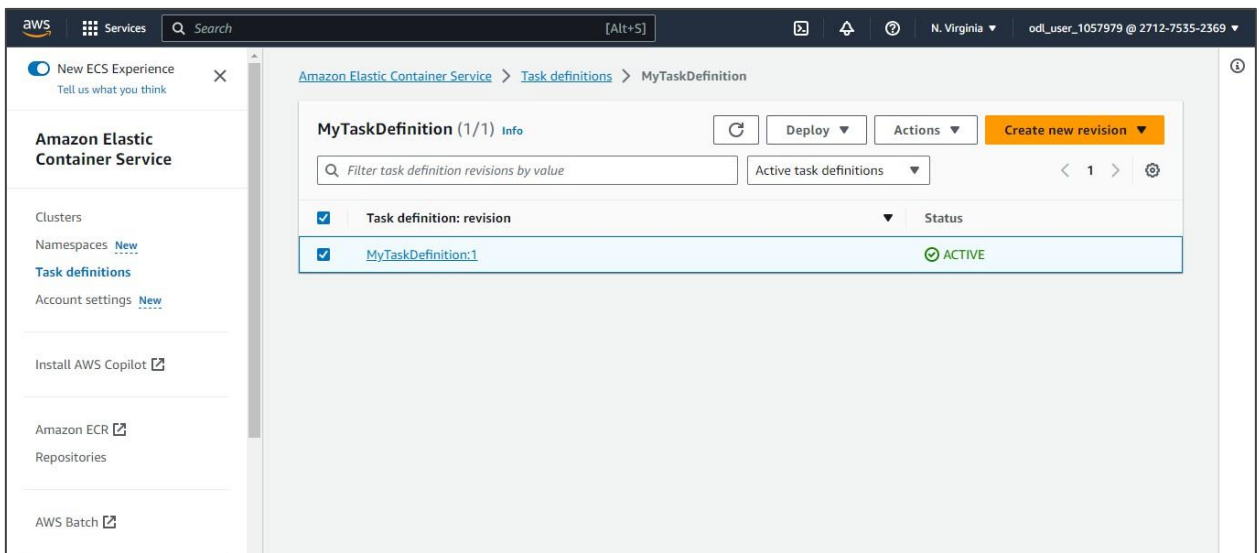




1.4 Choose Create



The task definition has been successfully executed.



A task has been added to **MyCluster** and triggered successfully as shown in the screenshot.

By following these steps, you have successfully created an ECS Cluster with Auto Scaling, defined and executed tasks within it.