

Hands-on tutorials

Deploy Docker Containers on Amazon ECS



Deploy Docker Containers on Amazon ECS: Hands-on tutorials

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Deploy Docker Containers on Amazon ECS	i
Overview	1
Implementation	1
Congratulations	18

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).

The screenshot shows the AWS console interface for Amazon ECS. The top navigation bar includes the AWS logo, a search bar, and the region 'N. Virginia'. The left sidebar lists navigation options such as 'New ECS Experience', 'Amazon ECS Clusters', 'Task Definitions', and 'Account Settings'. The main content area is titled 'Amazon Elastic Container Service (ECS)' and features a video player with a stick figure character. Below the video is a 'Get started' button, which is highlighted with a red box. Underneath the button is a link to 'Learn more about Amazon ECS'. The page is divided into three columns, each with an icon and a title: 'Run containers at scale', 'Flexible container placement', and 'Integrated and extensible'. Each column contains a brief description of the feature. At the bottom of the page, there is a footer with a feedback link and copyright information.

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**.

Getting Started with Amazon Elastic Container Service (Amazon ECS) using Fargate

Step 1: Container and Task
Step 2: Service
Step 3: Cluster
Step 4: Review

Diagram of ECS objects and how they relate

Container definition
Task definition
Service
Cluster

Container definition

Choose an image for your container below to get started quickly or define the container image to use.

sample-app
image : httpd:2.4
memory : 0.5GB (512)
cpu : 0.25 vCPU (256)

nginx
image : nginx:latest
memory : 0.5GB (512)
cpu : 0.25 vCPU (256)

tomcat-webserver
image : tomcat
memory : 2GB (2048)
cpu : 1 vCPU (1024)

custom
image : --
memory : --
cpu : --
Configure

Task definition

A task definition is a blueprint for your application, and describes one or more containers through attributes. Some attributes are configured at the task level but the majority of attributes are configured per container.

Task definition name	first-run-task-definition	⊖
Network mode	awsvpc	⊖
Task execution role	Create new	⊖
Compatibilities	FARGATE	⊖
Task memory	0.5GB (512)	
Task CPU	0.25 vCPU (256)	

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](#).

Getting Started with Amazon Elastic Container Service (Amazon ECS) using Fargate

Step 1: Container and Task
 Step 2: Service
 Step 3: Cluster
 Step 4: Review

Diagram of ECS objects and how they relate

Container definition
 Choose an image for your container below to get started quickly or define the container image to use.

sample-app
 image : httpd2.4
 memory : 0.5GB (512)
 cpu : 0.25 vCPU (256)

nginx
 image : nginx:latest
 memory : 0.5GB (512)
 cpu : 0.25 vCPU (256)

tomcat-webservlet
 image : tomcat
 memory : 2GB (2048)
 cpu : 1 vCPU (1024)

custom
 image : --
 memory : --
 cpu : --

Task definition
 A task definition is a blueprint for your application, and describes one or more containers through attributes. Some attributes are configured at the task level but the majority of attributes are configured per container.

Task definition name	first-run-task-definition	⊕
Network mode	awsvpc	⊕
Task execution role	Create new	⊕
Compatibilities	FARGATE	⊕
Task memory	0.5GB (512)	

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.

The screenshot shows the AWS Management Console interface for setting up Amazon ECS. At the top, there's a navigation bar with the AWS logo, a search bar, and the region 'N. Virginia'. Below this, the page title is 'Getting Started with Amazon Elastic Container Service (Amazon ECS) using Fargate'. A sidebar on the left lists four steps: 'Step 1: Container and Task', 'Step 2: Service' (which is highlighted in orange), 'Step 3: Cluster', and 'Step 4: Review'. The main content area is titled 'Diagram of ECS objects and how they relate' and contains a diagram showing a 'Container definition' and a 'Task definition' nested within a 'Service', which is in turn nested within a 'Cluster'. Below the diagram is a form titled 'Define your service' with an 'Edit' button. The form includes the following fields: 'Service name' (sample-app-service), 'Number of desired tasks' (1), 'Security group' (Automatically create new), and 'Load balancer type' (None selected, Application Load Balancer unselected). At the bottom of the form are 'Cancel', 'Previous', and 'Next' buttons. A footer bar at the very bottom contains a feedback link, a language selection notice, and copyright information for Amazon Web Services, Inc.

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**.

Getting Started with Amazon Elastic Container Service (Amazon ECS) using Fargate

Step 1: Container and Task
Step 2: Service
Step 3: Cluster
Step 4: Review

Diagram of ECS objects and how they relate

Define your service Edit

A service allows you to run and maintain a specified number (the "desired count") of simultaneous instances of a task definition in an ECS cluster.

Service name `sample-app-service`

Number of desired tasks `1`

Security group `Automatically create new`

Two security groups are created to secure your service: An Application Load Balancer security group that allows all traffic on the Application Load Balancer port and an Amazon ECS security group that allows all traffic ONLY from the Application Load Balancer security group. You can further configure security groups and network access outside of this wizard.

Load balancer type None Application Load Balancer

Load balancer listener port `80`

Load balancer listener protocol `HTTP`

*Required Cancel Previous Next

Feedback [Looking for language selection? Find it in the new Unified Settings](#)

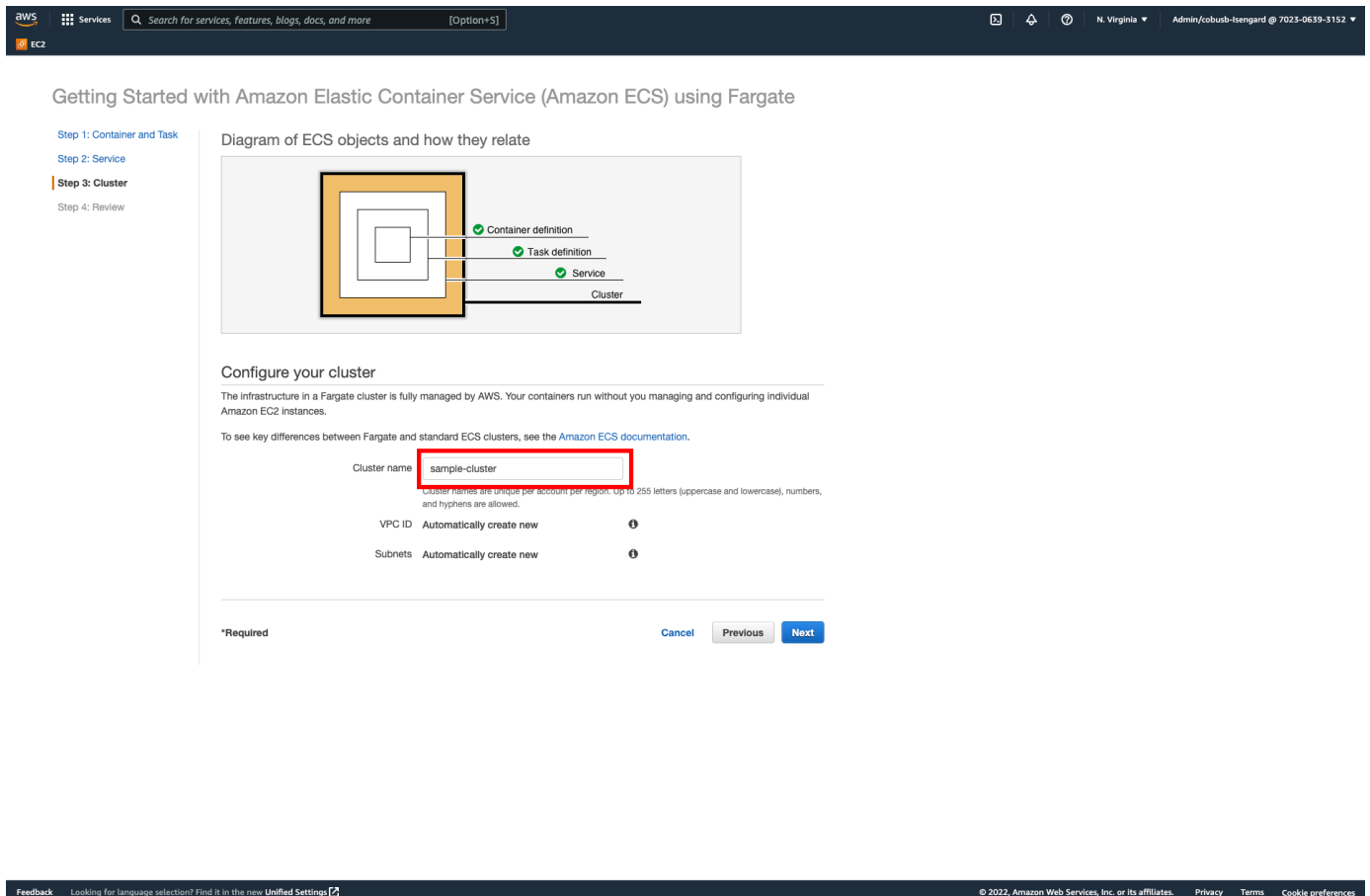
© 2022, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

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**.



Getting Started with Amazon Elastic Container Service (Amazon ECS) using Fargate

Step 1: Container and Task
Step 2: Service
Step 3: Cluster
Step 4: Review

Diagram of ECS objects and how they relate

Container definition
Task definition
Service
Cluster

Configure your cluster

The infrastructure in a Fargate cluster is fully managed by AWS. Your containers run without you managing and configuring individual Amazon EC2 instances.

To see key differences between Fargate and standard ECS clusters, see the [Amazon ECS documentation](#).

Cluster name

Cluster names are unique per account per region. Up to 255 letters (uppercase and lowercase), numbers, and hyphens are allowed.

VPC ID

Subnets

*Required

Feedback Looking for language selection? Find it in the new [Unified Settings](#)

© 2022, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

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**.

Getting Started with Amazon Elastic Container Service (Amazon ECS) using Fargate

Step 1: Container and Task
Step 2: Service
Step 3: Cluster
Step 4: Review

Diagram of ECS objects and how they relate

Review

Review the configuration you've set up before creating your task definition, service, and cluster.

Task definition [Edit](#)

Task definition name: first-run-task-definition
Network mode: awavpc
Task execution role: Create new
Container name: sample-app
Image: httpd:2.4
Memory: 512
Port: 80
Protocol: HTTP

Service [Edit](#)

Service name: sample-app-service
Number of desired tasks: 1
Load balancer listener port: 80
Load balancer listener protocol: HTTP

Cluster [Edit](#)

Cluster name: sample-cluster
VPC ID: Automatically create new
Subnets: Automatically create new

*Required [Cancel](#) [Previous](#) [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**.

Getting Started with Amazon Elastic Container Service (Amazon ECS) using Fargate

Launch Status

We are creating resources for your service. This may take up to 10 minutes. When we're complete, you can view your service.

[Back](#) [View service](#)

Additional features that you can add to your service after creation

Scale based on metrics
You can configure scaling rules based on CloudWatch metrics

Preparing service : 10 of 10 complete

Resource Name	Status
ECS resource creation	complete
Cluster sample-cluster	complete
Task definition first-run-task-definition:1	complete
Service sample-app-service	complete
Additional AWS service integrations	complete
Log group /ecs/first-run-task-definition	complete
CloudFormation stack EC2ContainerService-sample-cluster	complete
VPC vpc-0ee9af62a5e550290	complete
Subnet 1 subnet-02bbee854da600e2b	complete
Subnet 2 subnet-04d7c111f8a5ac6d0	complete
Security group sg-0751d90ab53cbbc18	complete
Load balancer arn:aws:elasticloadbalancing:us-east-1:702306393152:loadbalancer/app/EC2Co-EcsEI-A7GQKEJBDN77/9571de62498cdae0	complete

Feedback [Looking for language selection? Find it in the new Unified Settings](#)

© 2022, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

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.

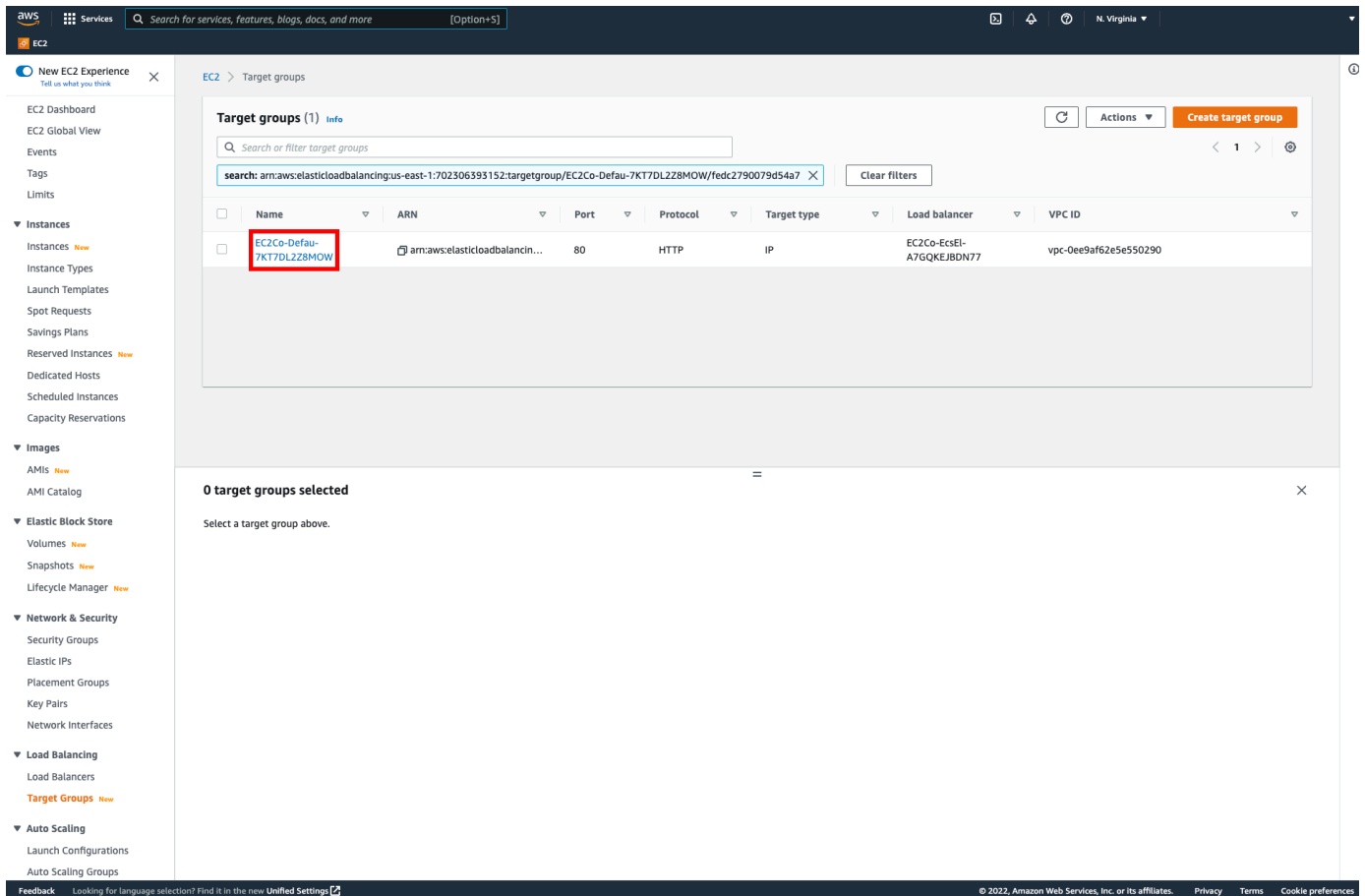
1. View details about the application

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

The screenshot displays the AWS Management Console interface for an Amazon ECS service. The breadcrumb navigation shows 'Clusters > sample-cluster > Service: sample-app-service'. The service name is 'sample-app-service'. The cluster is 'sample-cluster' with a status of 'ACTIVE'. The task definition is 'first-run-task-definition:1'. The service type is 'REPLICA', launch type is 'FARGATE', and service role is 'AWSServiceRoleForECS'. The service was created by 'arn:aws:iam::702306393152:role/Admin'. The service has 1 desired count, 0 pending count, and 1 running count. Below the service details, there are tabs for 'Details', 'Tasks', 'Events', 'Auto Scaling', 'Deployments', 'Metrics', 'Tags', and 'Logs'. The 'Load Balancing' section shows a table with one target group: 'EC2Co-Defau-7K77DL228MOW' with container name 'sample-app' and container port '80'. The 'Network Access' section shows health check grace period '0', allowed VPC 'vpc-0ee9af62e5e550290', allowed subnets 'subnet-02dbaea54da50e2b,subnet-04d7c111f8a5ac6d0', security groups 'sg-0751d60ab53cbbc18', and auto-assign public IP 'ENABLED'.

2. View target group details

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



Name	ARN	Port	Protocol	Target type	Load balancer	VPC ID
EC2Co-Defau-7KT7DL2Z8MOW	arn:aws:elasticloadbalancing:us-east-1:702306393152:targetgroup/EC2Co-Defau-7KT7DL2Z8MOW/fedc2790079d54a7	80	HTTP	IP	EC2Co-EcsE1-A7GQKEJBDN77	vpc-0ee9af62e5e550290

3. Select the load balancer

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

The screenshot displays the AWS Management Console interface for an Amazon EC2 Target Group. The breadcrumb navigation shows the path: EC2 > Target groups > EC2Co-Defau-7KT7DL2Z8MOW. The main heading is 'EC2Co-Defau-7KT7DL2Z8MOW'. The 'Details' section shows the following information:

Target type: IP	Protocol: Port: HTTP: 80	Protocol version: HTTP1	VPC: vpc-0ee9af62e5e550290		
IP address type: IPv4	Load balancer: EC2CO-ECS-EL-A7GQKEJBDN77				
Total targets: 1	Healthy: 1	Unhealthy: 0	Unused: 0	Initial: 0	Draining: 0

The 'Registered targets' section shows one target:

IP address	Port	Zone	Health status	Health status details
10.0.0.190	80	us-east-1a	healthy	

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.

The screenshot shows the AWS Management Console interface for creating and configuring an Amazon Elastic Load Balancing (ALB) instance. The console displays the 'Basic Configuration' tab for the load balancer 'EC2Co-EcsEI-A7GQKEJBDN77'. The 'DNS name' field is highlighted with a red box, showing 'EC2Co-EcsEI-A7GQKEJBDN77-655917556.us-east-1.elb.amazonaws.com'. Other configuration details include VPC ID, Availability Zones, and Security Groups.

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).

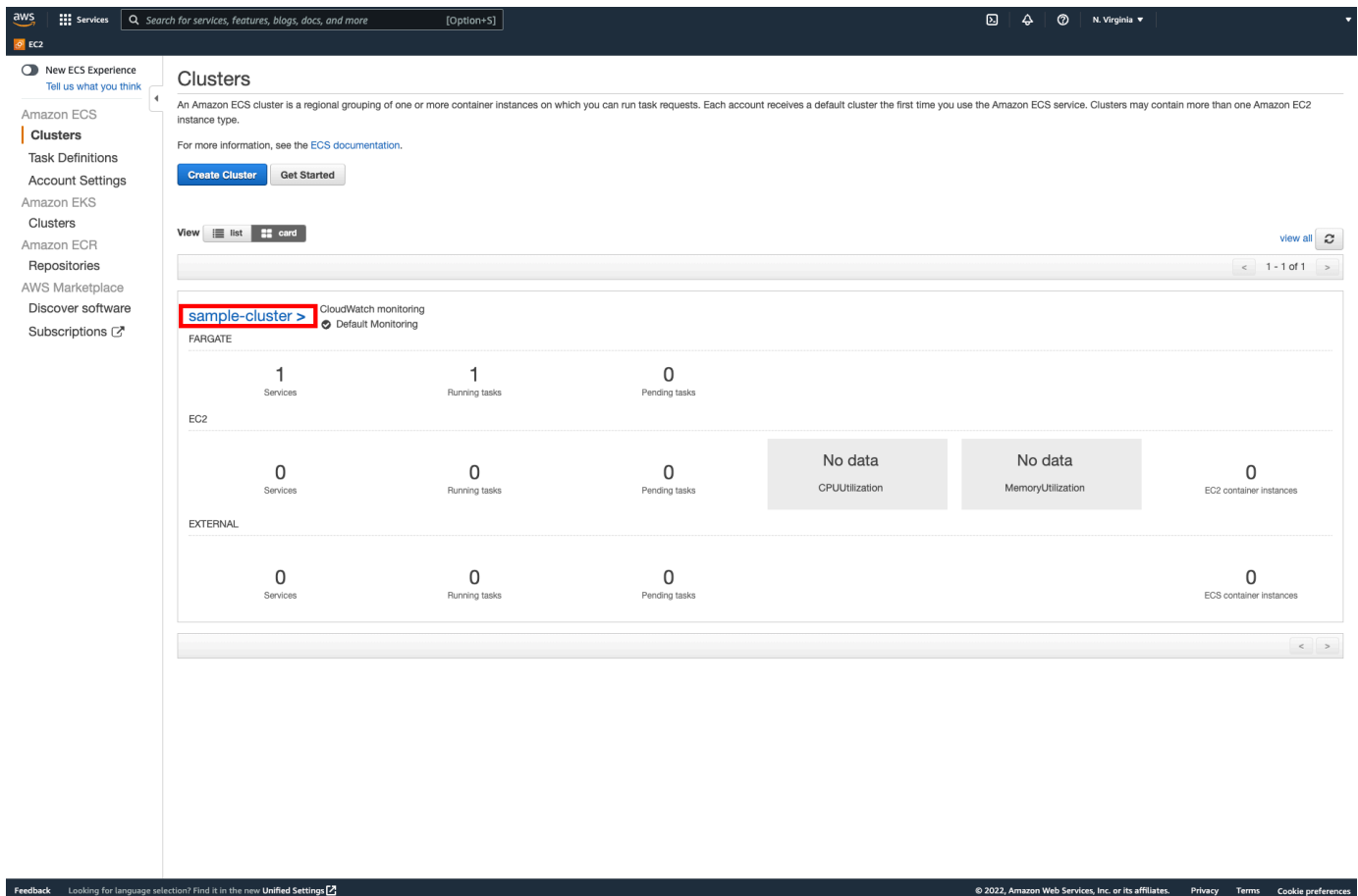


(Optional) Clean up resources

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).



The screenshot shows the Amazon ECS console interface. The left sidebar contains navigation options: Amazon ECS, Clusters, Task Definitions, Account Settings, Amazon EKS, Amazon ECR, Repositories, AWS Marketplace, Discover software, and Subscriptions. The main content area is titled 'Clusters' and includes a 'Create Cluster' button and a 'Get Started' button. Below this, there are view options (list, card) and a 'view all' link. The cluster list shows one cluster named 'sample-cluster' with a red box around its name. The cluster details are displayed in a table format:

Instance Type	Services	Running tasks	Pending tasks	CPU Utilization	Memory Utilization	ECS container instances
FARGATE	1	1	0			
EC2	0	0	0	No data	No data	0
EXTERNAL	0	0	0			0

2. Choose Delete Cluster

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

The screenshot shows the AWS Management Console interface for an Amazon ECS cluster. The cluster name is 'sample-cluster' and its status is 'ACTIVE'. The console displays various metrics such as 'Registered container instances', 'Pending tasks count', 'Running tasks count', 'Active service count', and 'Draining service count'. A table below shows the details of the services running on the cluster.

Service Name	Status	Service type	Task Definition	Desired tasks	Running tasks	Launch type	Platform version
sample-app-service	ACTIVE	REPLICA	first-run-task-definition:1	1	1	FARGATE	LATEST(1.4.0)

3. Confirm cluster deletion

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

The screenshot shows the AWS Management Console interface for an Amazon ECS cluster named 'sample-cluster'. A 'Delete Cluster' dialog box is open, prompting the user to confirm the deletion of the cluster and all associated resources. The dialog box contains the following text:

Delete Cluster

Deleting the cluster also deletes the CloudFormation stack `EC2ContainerService-sample-cluster`.

Are you sure you want to delete the cluster `sample-cluster` and all the ECS resources within it?

Enter the phrase "delete me" into the field below to confirm deletion.

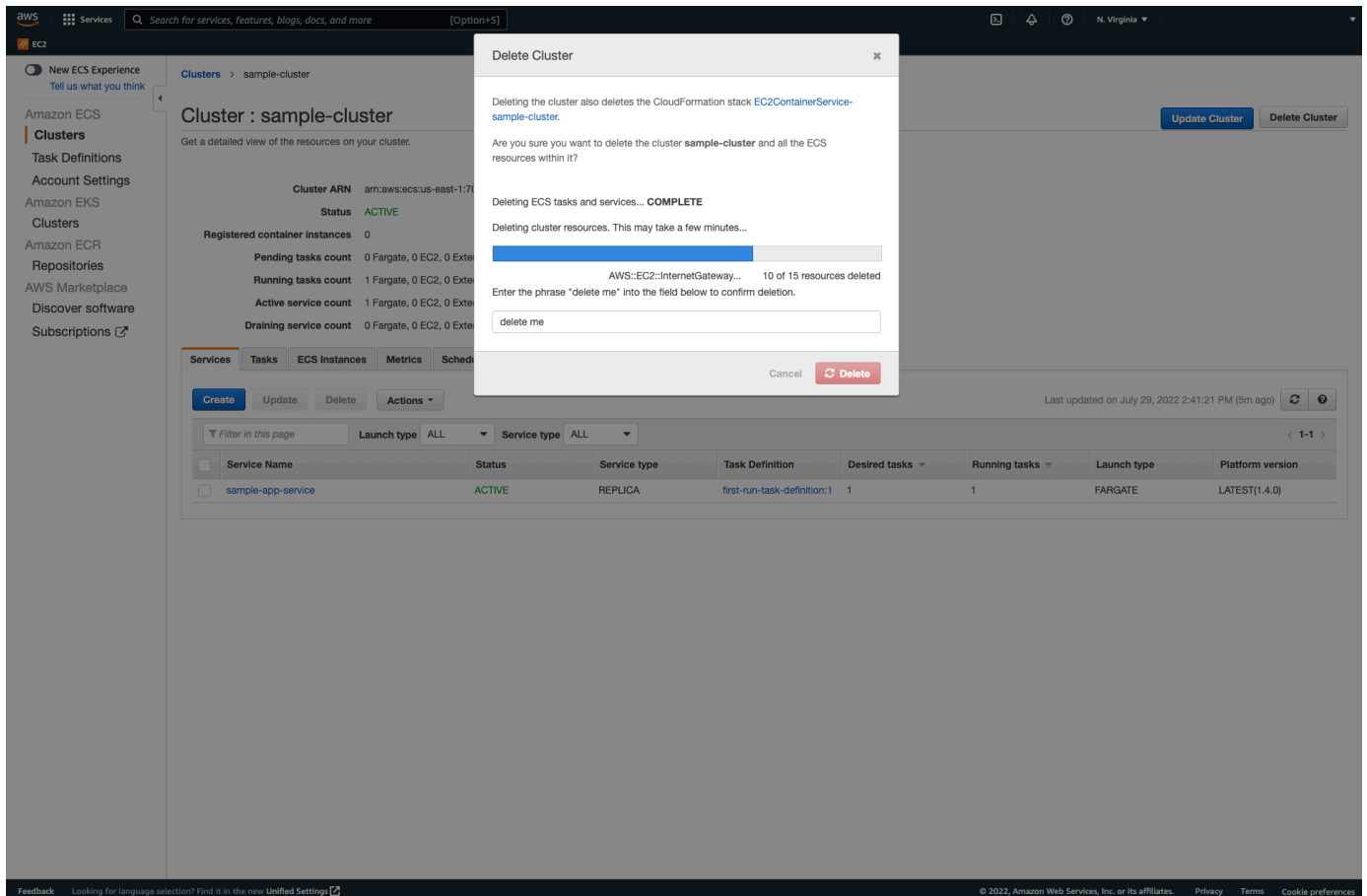
The text 'delete me' is entered into the confirmation field, which is highlighted with a red box. The dialog box also includes 'Cancel' and 'Delete' buttons.

The background shows the cluster details for 'sample-cluster', including its ARN, status (ACTIVE), and a table of services. The table has the following columns: Service Name, Status, Service type, Task Definition, Desired tasks, Running tasks, Launch type, and Platform version.

Service Name	Status	Service type	Task Definition	Desired tasks	Running tasks	Launch type	Platform version
sample-app-service	ACTIVE	REPLICA	first-run-task-definition:1	1	1	FARGATE	LATEST(1.4.0)

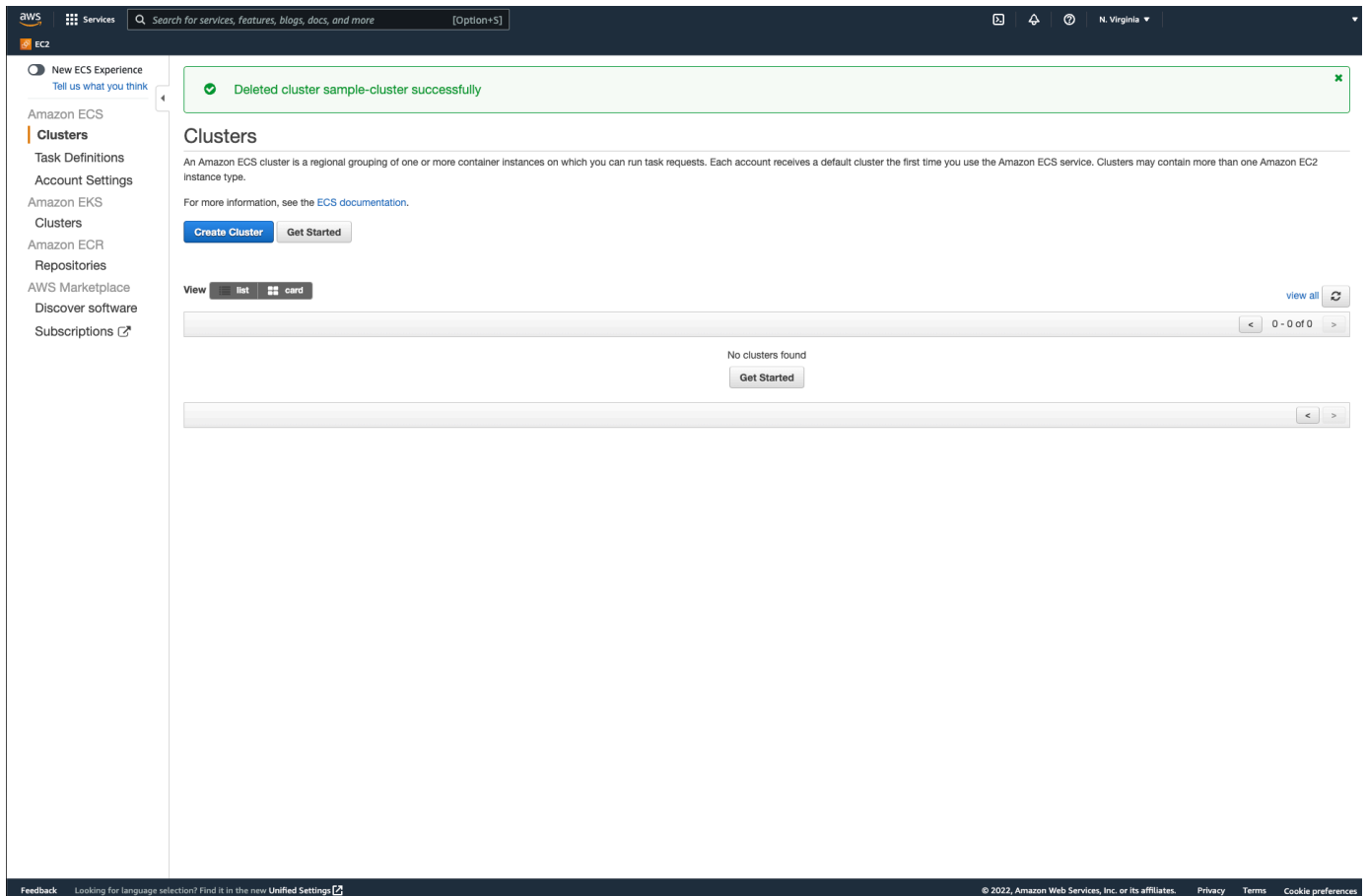
4. Monitor cluster deletion

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.