Lesson 10 Demo 05

Running Task on a Fargate Cluster

Objective: To run a task on a Fargate cluster to deploy a containerized application without managing the underlying infrastructure

Tools required: An AWS account

Prerequisites: None

Steps to be followed:

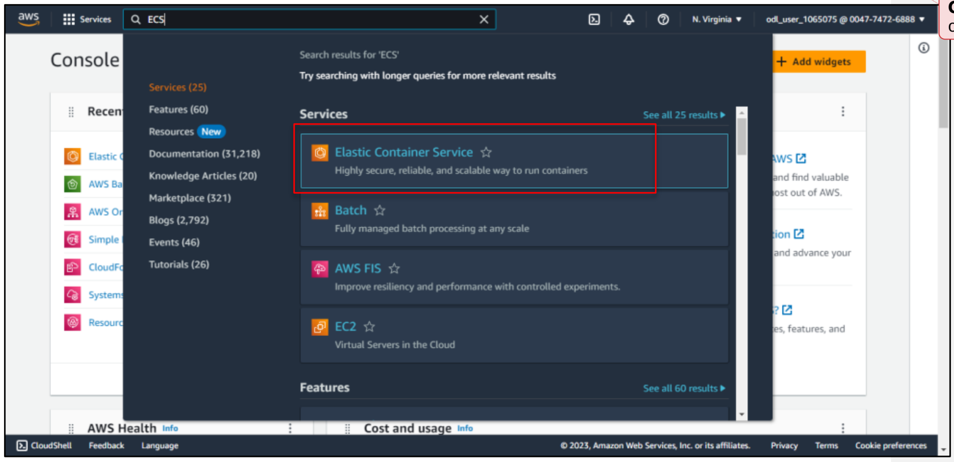
1. Create a Fargate cluster
2. Create a task definition
3. Run the Fargate Cluster

Step 1: Create a Fargate cluster

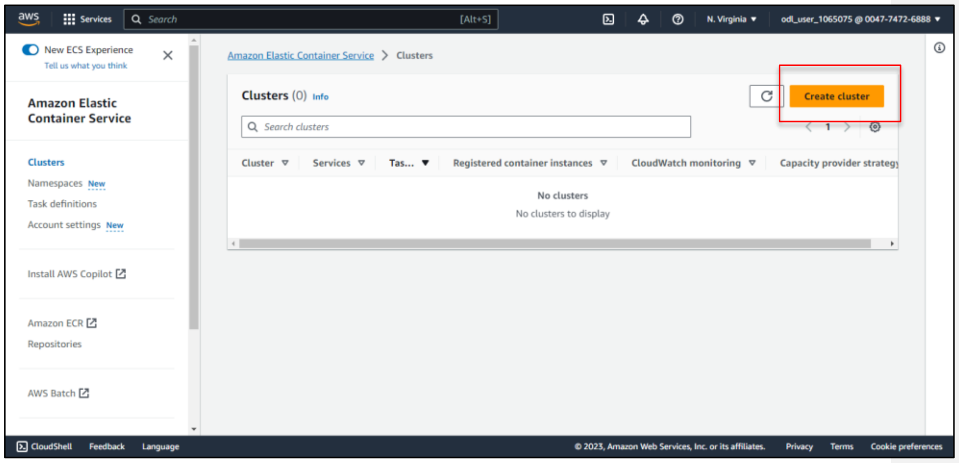
1.1 Navigate to the **AWS Management Console**, search for **ECS** and select **Elastic Container Service**

Commented [SS1]: Global - Start the sentence with an action verb to maintain the parallelism

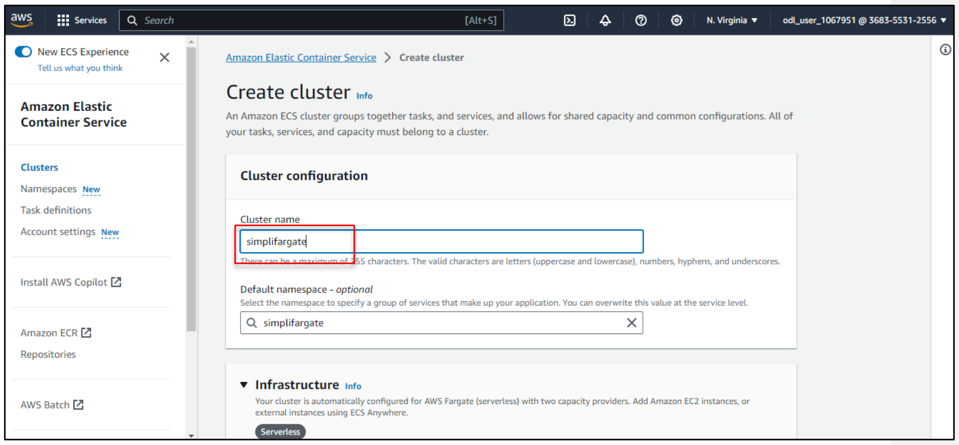
Commented [SS2]: Global - Add the highlight in the images of the steps



1.2 Click on **Create cluster** in the **Clusters**



1.3 Add the **Cluster name**, specify **AWS Fargate (serverless)** for the infrastructure, leave other settings at default, and click **Create**



Create cluster Info

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: simplifargate

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

Default namespace - *optional*

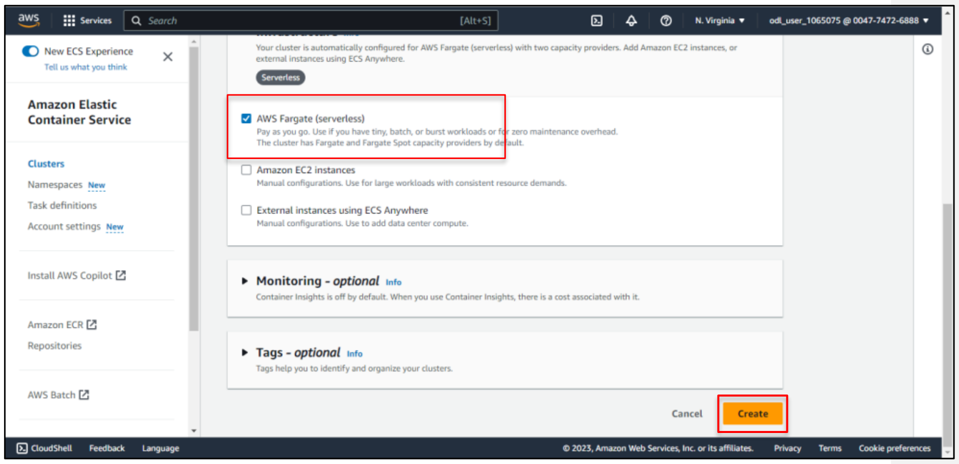
Select the namespace to specify a group of services that make up your application. You can overwrite this value at the service level.

simplifargate

Infrastructure Info

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

Serverless



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

Serverless

[✓] 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.

[ ] External instances using ECS Anywhere
Manual configurations. Use to add data center compute.

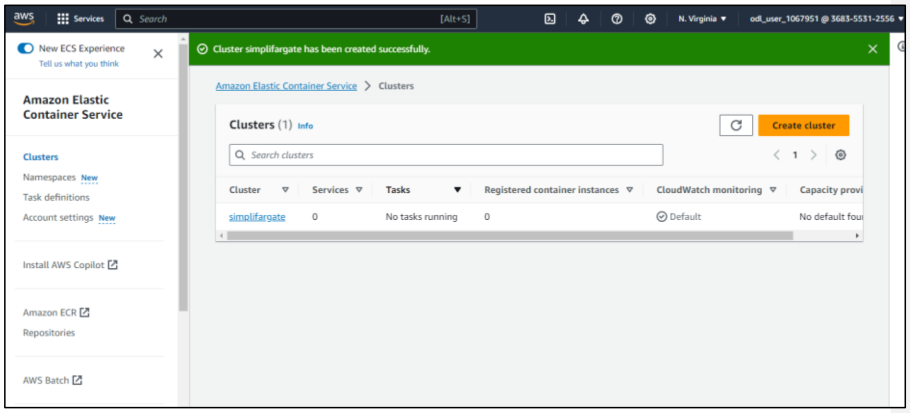
Monitoring - *optional* Info

Container Insights is off by default. When you use Container Insights, there is a cost associated with it.

Tags - *optional* Info

Tags help you to identify and organize your clusters.

1.4 Verify the cluster creation as shown below:



Cluster simplifargate has been created successfully.

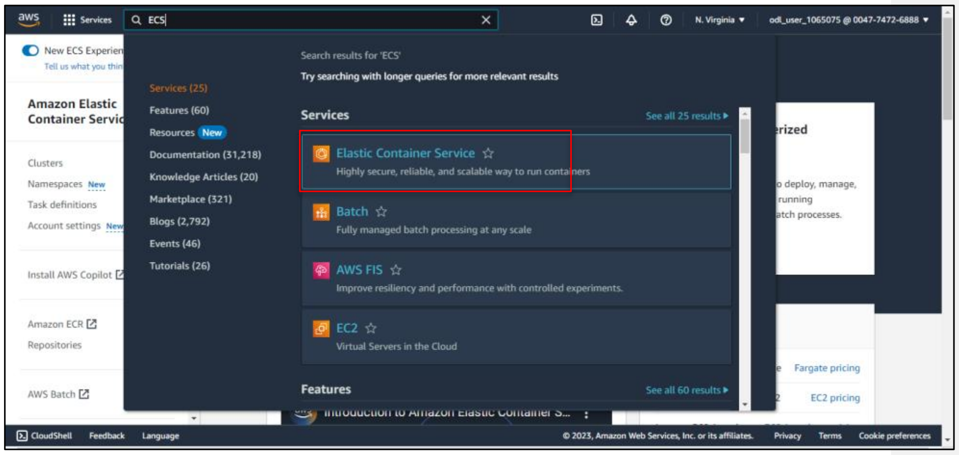
Cluster	Services	Tasks	Registered container instances	CloudWatch monitoring	Capacity provi
simplifargate	0	No tasks running	0	Default	No default fou

Note: Do not close the above tab. It will be necessary for reference. ECS Cluster will be created.

Commented [SS3]: Rephrase to > Do not close the above tab as it will be necessary for reference when creating the ECS Cluster.

Step 2: Create a task definition

2.1 Navigate to **AWS Management Console**, search for **ECS** and select **Elastic Container Service**



Search results for 'ECS'
Try searching with longer queries for more relevant results

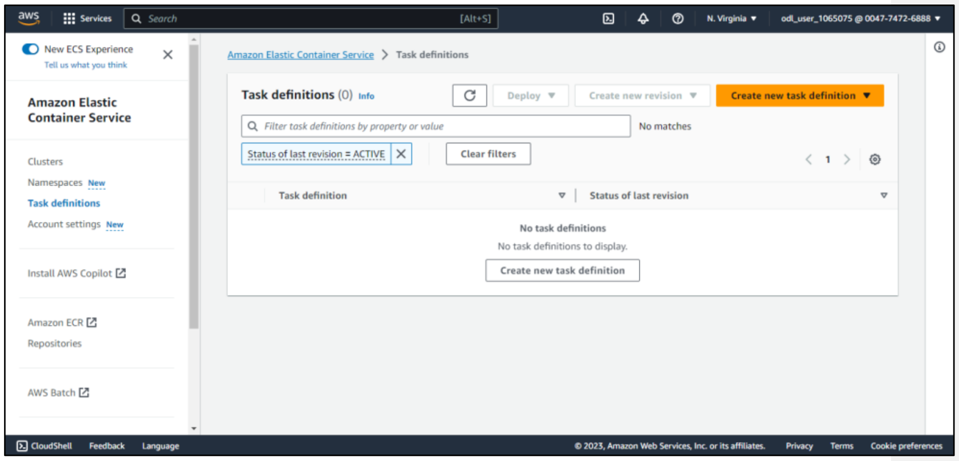
Elastic Container Service
Highly secure, reliable, and scalable way to run containers

Batch
Fully managed batch processing at any scale

AWS FIS
Improve resiliency and performance with controlled experiments.

EC2
Virtual Servers in the Cloud

2.2 Click on **Task definitions** and on **Create new task definition** On the left panel of the ECS console



Amazon Elastic Container Service > Task definitions

Task definitions (0) Info

Deploy | Create new revision | Create new task definition

Filter task definitions by property or value — No matches

Status of last revision = ACTIVE | Clear filters

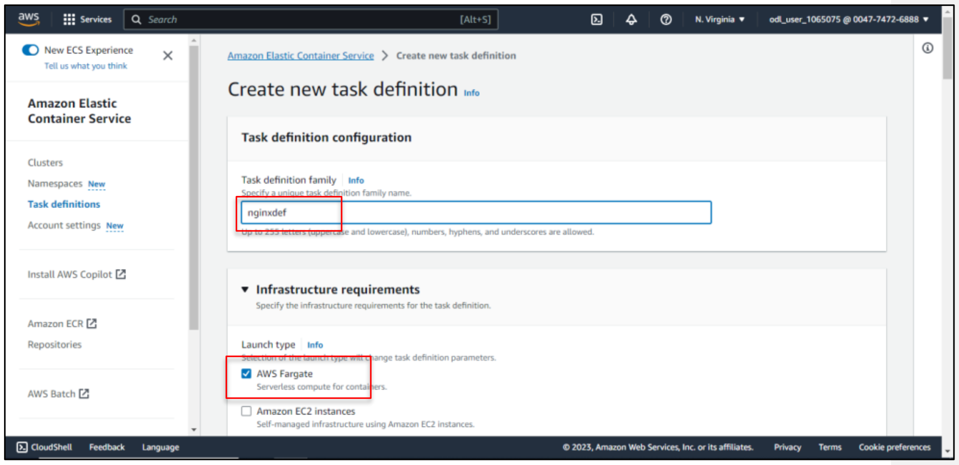
Task definition	Status of last revision

No task definitions

No task definitions to display.

Create new task definition

2.3 Specify task definition **family = nginxdef, Launch type = AWS Fargate, CPU = 0.25 vCPU,** and **Memory = 0.5 GB** In the Task definition configuration page



Amazon Elastic Container Service > Create new task definition

Create new task definition Info

Task definition configuration

Task definition family Info

Specify a unique task definition family name.

nginxdef

Up to 255 letters (uppercase and lowercase), numbers, hyphens, and underscores are allowed.

Infrastructure requirements

Specify the infrastructure requirements for the task definition.

Launch type Info

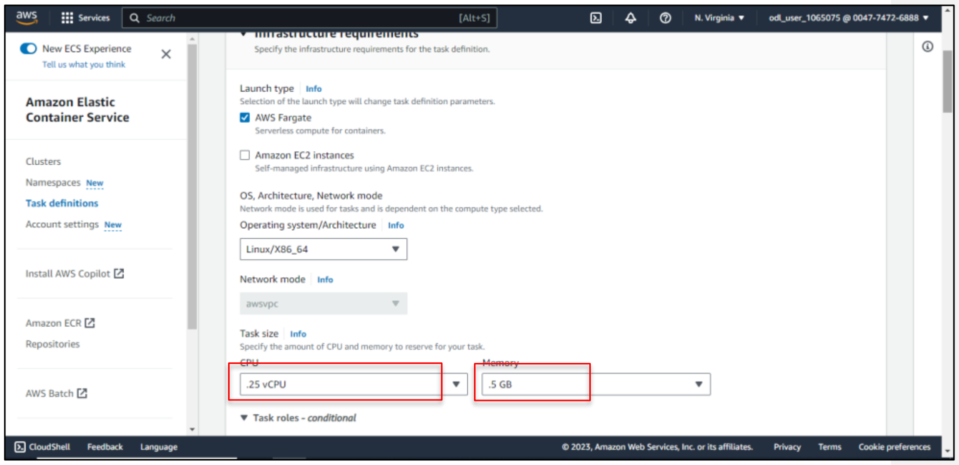
Selection of the launch type will change task definition parameters.

[✓] AWS Fargate

Serverless compute for containers.

[ ] Amazon EC2 instances

Self-managed infrastructure using Amazon EC2 instances.



Infrastructure requirements
Specify the infrastructure requirements for the task definition.

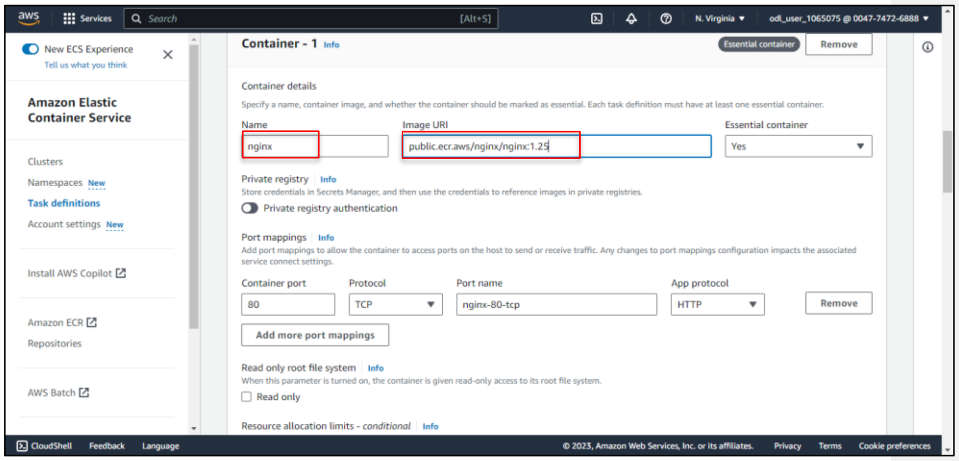
Launch type Info
Selection of the launch type will change task definition parameters.
☑ AWS Fargate
Serverless compute for containers.
☐ Amazon EC2 instances
Self-managed infrastructure using Amazon EC2 instances.

OS, Architecture, Network mode
Network mode is used for tasks and is dependent on the compute type selected.
Operating system/Architecture Info: Linux/X86_64
Network mode Info: awsvpc

Task size Info
Specify the amount of CPU and memory to reserve for your task.
CPU: .25 vCPU
Memory: .5 GB

▼ Task roles - *conditional*

2.4 For Container-1 details, enter **Name = nginx** and **Image URI = public.ecr.aws/nginx/nginx:1.25**



Container - 1 Info

Container details
Specify a name, container image, and whether the container should be marked as essential. Each task definition must have at least one essential container.

Name: nginx
Image URI: public.ecr.aws/nginx/nginx:1.25
Essential container: Yes

Private registry Info
Store credentials in Secrets Manager, and then use the credentials to reference images in private registries.
Private registry authentication

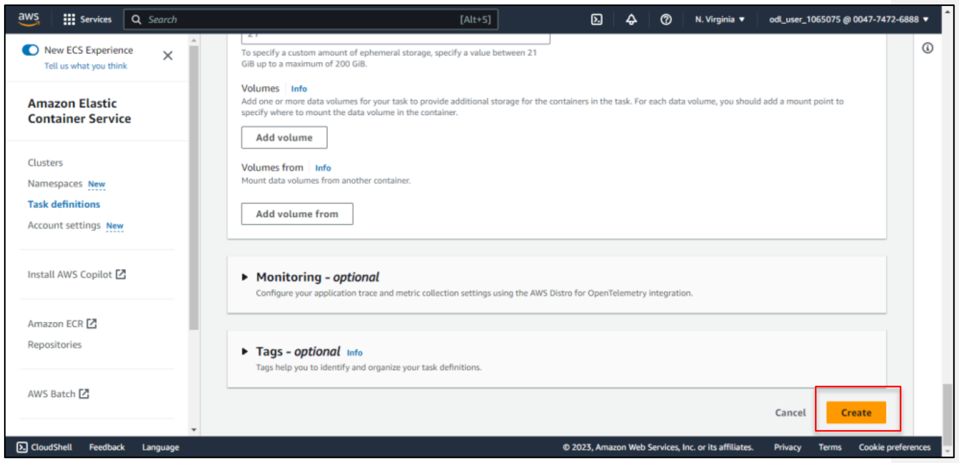
Port mappings Info
Add port mappings to allow the container to access ports on the host to send or receive traffic. Any changes to port mappings configuration impacts the associated service connect settings.

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

Read only root file system Info
When this parameter is turned on, the container is given read-only access to its root file system.
☐ Read only

Resource allocation limits - *conditional* Info

2.5 Leave other options default and click **Create**



To specify a custom amount of ephemeral storage, specify a value between 21 GiB up to a maximum of 200 GiB.

Volumes Info

Add one or more data volumes for your task to provide additional storage for the containers in the task. For each data volume, you should add a mount point to specify where to mount the data volume in the container.

Volumes from Info

Mount data volumes from another container.

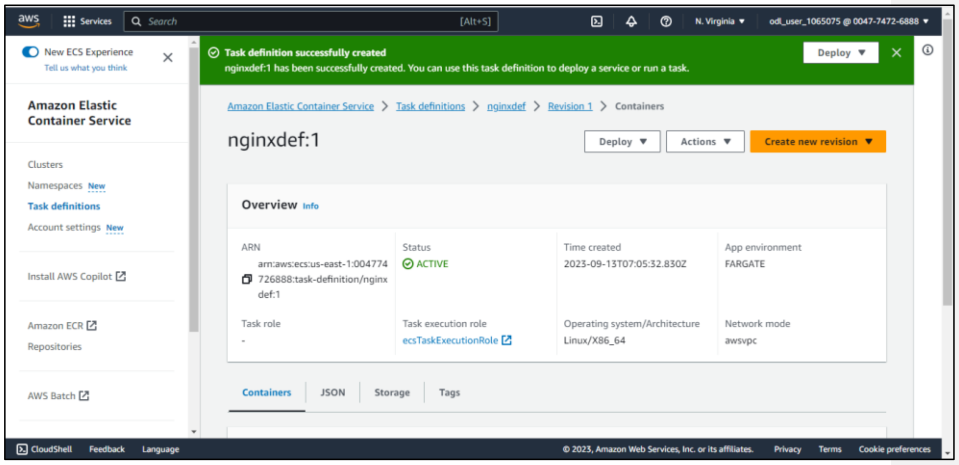
Monitoring - *optional*

Configure your application trace and metric collection settings using the AWS Distro for OpenTelemetry integration.

Tags - *optional* Info

Tags help you to identify and organize your task definitions.

The task definition has been created successfully as shown:



Task definition successfully created

nginxdef:1 has been successfully created. You can use this task definition to deploy a service or run a task.

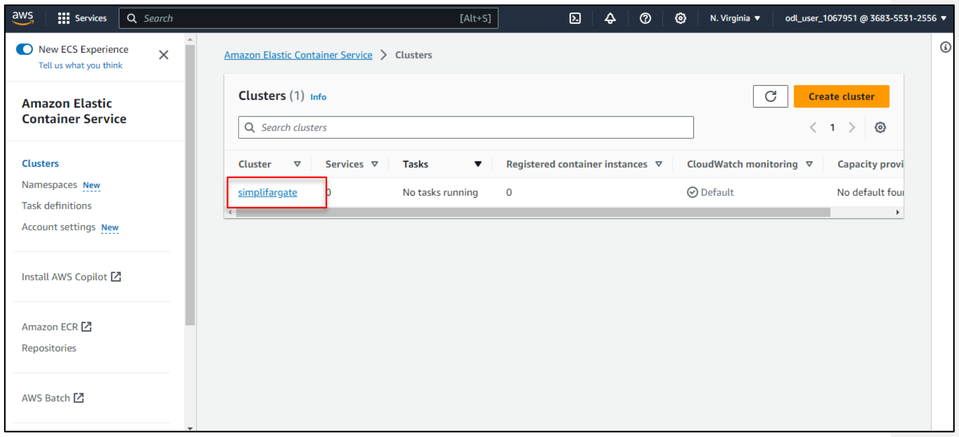
nginxdef:1

Overview Info

ARN arn:aws:ecs:us-east-1:004774726888:task-definition/nginxdef:1	Status ACTIVE	Time created 2023-09-13T07:05:32.830Z	App environment FARGATE
Task role -	Task execution role ecsTaskExecutionRole	Operating system/Architecture Linux/X86_64	Network mode awsvpc

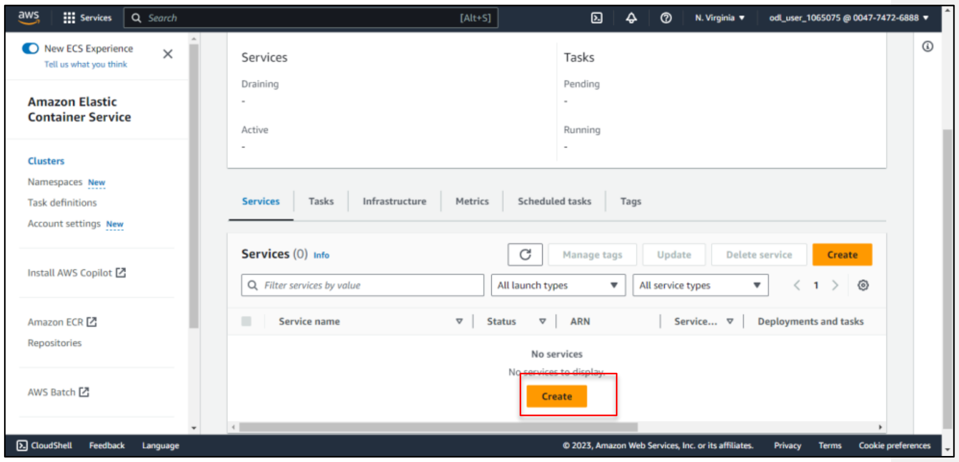
Step 3: Run Fargate Cluster

3.1 Return to the **ECS** home page and open the newly created cluster from **Clusters**

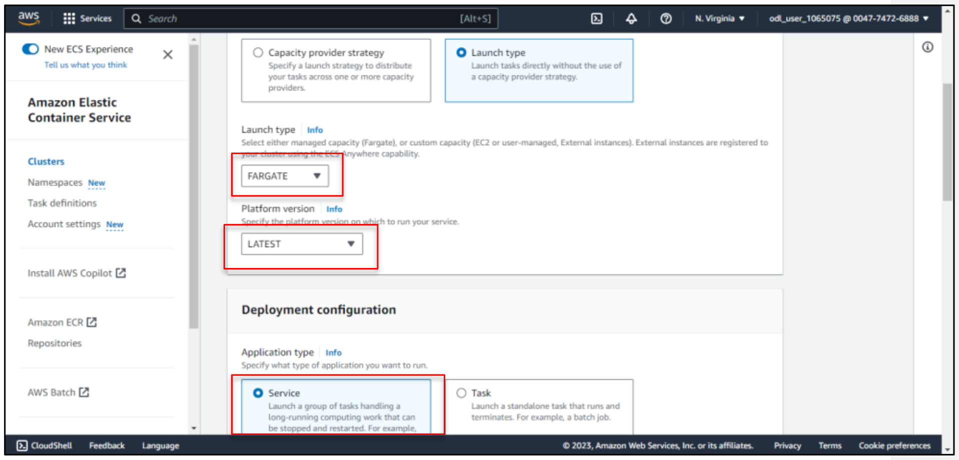


Cluster	Services	Tasks	Registered container instances	CloudWatch monitoring	Capacity provi
simplifargate	0	No tasks running	0	Default	No default fou

3.2 Click on **Create** under **Services**



3.3 Choose **FARGATE** under **Launch type**, select **LATEST** as the **Platform version**, and **Service** as the **Application type**

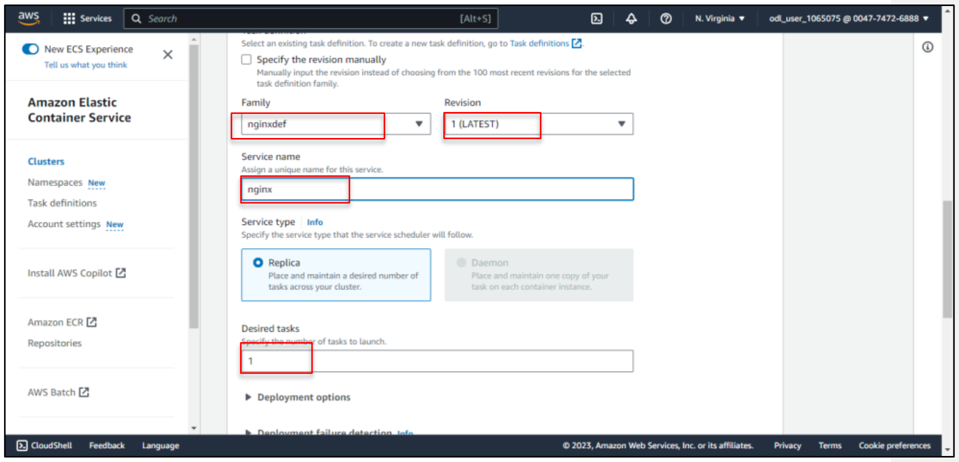


Launch type: FARGATE

Platform version: LATEST

Application type: Service

3.4 Choose family as **nginxdef** (created earlier), revision as **1 (LATEST), service name** as **nginx, Service type** as **Replica** and **Desired tasks** as **1**



Family: nginxdef

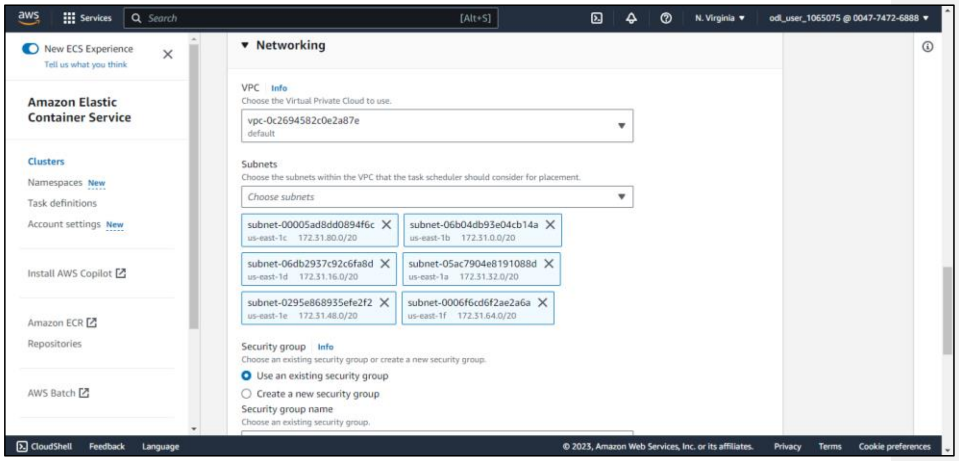
Revision: 1 (LATEST)

Service name: nginx

Service type: Replica

Desired tasks: 1

3.5 In the **Networking**, leave default **VPC** and **Load balancing** as **none**. Now, click **Create.**



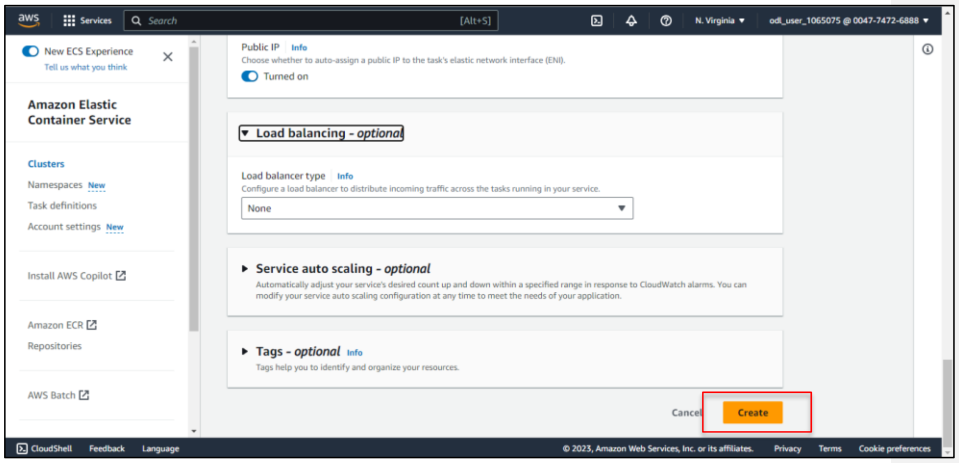
▼ Networking

VPC Info
Choose the Virtual Private Cloud to use.
vpc-0c2694582c0e2a87e
default

Subnets
Choose the subnets within the VPC that the task scheduler should consider for placement.
Choose subnets

subnet-00005ad8dd0894f6c us-east-1c 172.31.80.0/20
subnet-06b04db93e04cb14a us-east-1b 172.31.0.0/20
subnet-06db2937c92c6fa8d us-east-1d 172.31.16.0/20
subnet-05ac7904e8191088d us-east-1a 172.31.32.0/20
subnet-0295e868935efe2f2 us-east-1e 172.31.48.0/20
subnet-0006f6cd6f2ae2a6a us-east-1f 172.31.64.0/20

Security group Info
Choose an existing security group or create a new security group.
[✓ Use an existing security group]
Create a new security group
Security group name
Choose an existing security group.



Public IP Info
Choose whether to auto-assign a public IP to the task's elastic network interface (ENI).
Turned on

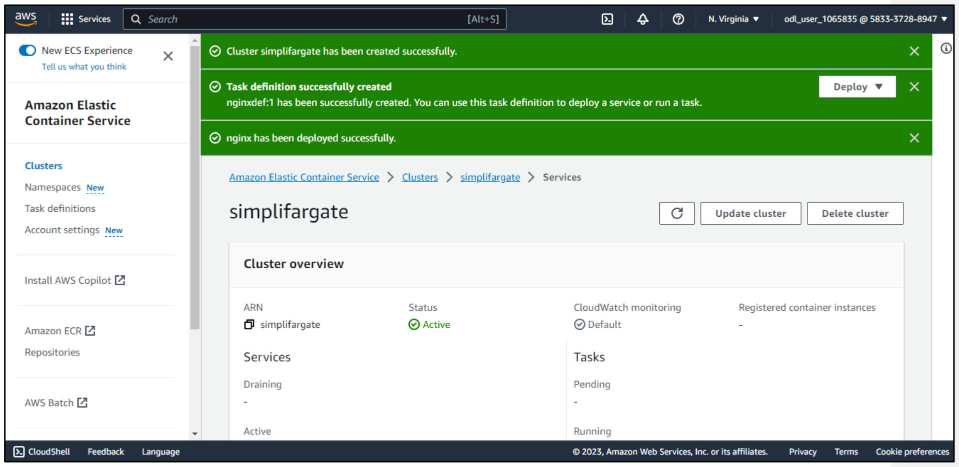
▼ Load balancing - *optional*

Load balancer type Info
Configure a load balancer to distribute incoming traffic across the tasks running in your service.
None

▸ Service auto scaling - *optional*
Automatically adjust your service's desired count up and down within a specified range in response to CloudWatch alarms. You can modify your service auto scaling configuration at any time to meet the needs of your application.

▸ Tags - *optional* Info
Tags help you to identify and organize your resources.

3.6 Wait until service creation is completed and 1/1 of tasks are active as shown below:



Cluster simplifargate has been created successfully.

Task definition successfully created
nginxdef:1 has been successfully created. You can use this task definition to deploy a service or run a task.

nginx has been deployed successfully.

Amazon Elastic Container Service > Clusters > simplifargate > Services

simplifargate

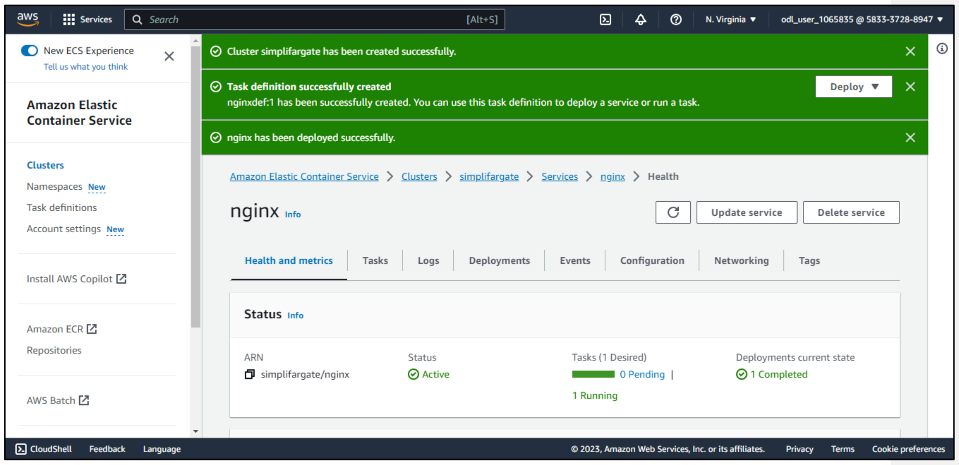
Cluster overview

ARN: simplifargate
Status: Active
CloudWatch monitoring: Default
Registered container instances: -

Services
Draining: -
Active

Tasks
Pending: -
Running

3.7 View the service details by clicking nginx once the service is running successfully, then click **Tasks**



Cluster simplifargate has been created successfully.

Task definition successfully created
nginxdef:1 has been successfully created. You can use this task definition to deploy a service or run a task.

nginx has been deployed successfully.

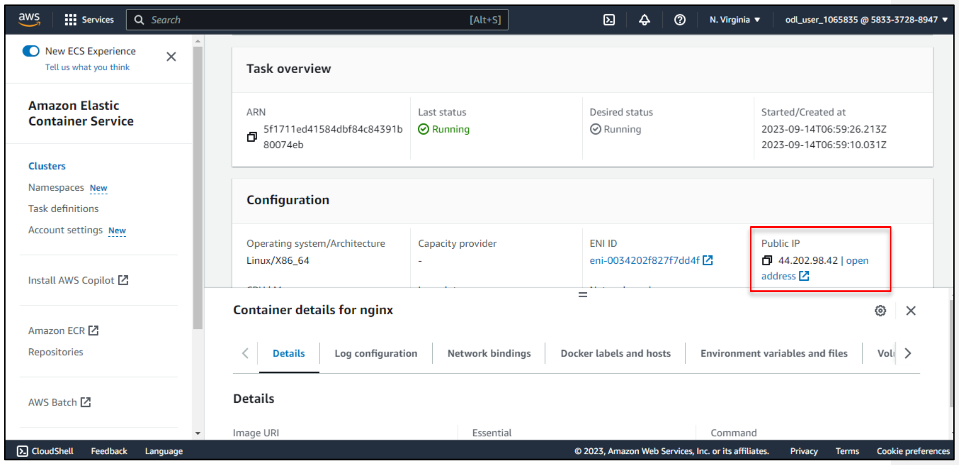
Amazon Elastic Container Service > Clusters > simplifargate > Services > nginx > Health

nginx

Status

ARN: simplifargate/nginx
Status: Active
Tasks (1 Desired): 0 Pending | 1 Running
Deployments current state: 1 Completed

3.8 Copy the public IP address and open it in a new browser



Task overview

ARN	Last status	Desired status	Started/Created at
5f1711ed41584dbf84c84391b80074eb	Running	Running	2023-09-14T06:59:26.213Z 2023-09-14T06:59:10.031Z

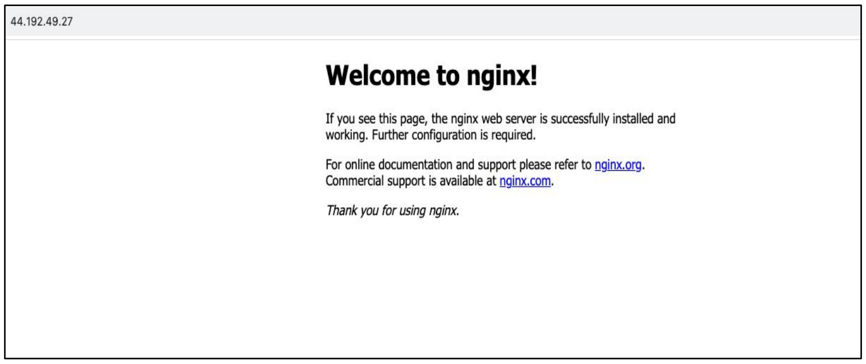
Configuration

Operating system/Architecture	Capacity provider	ENI ID	Public IP
Linux/X86_64	-	eni-0034202f827f7dd4f	44.202.98.42 | open address

Details

Image URI | Essential | Command

3.9 Open the URL in the new browser to see the nginx page loading as below:


44.192.49.27

Welcome to nginx!

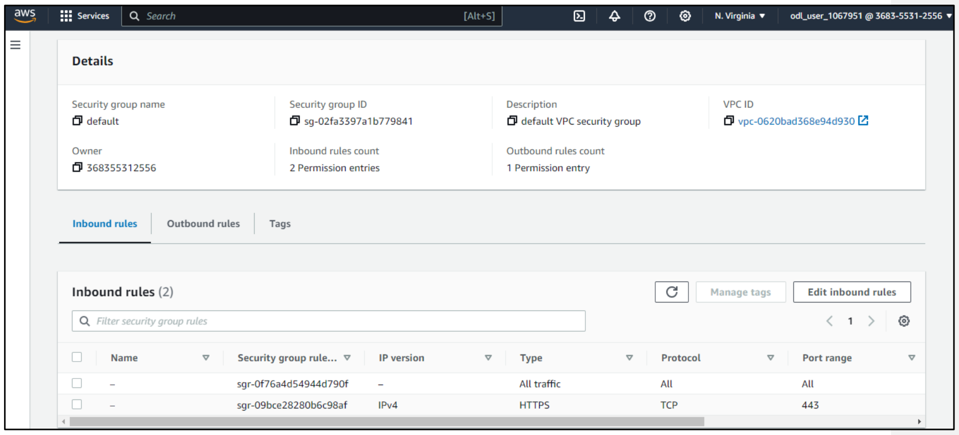
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Note : f the web page is not loading, go to **Task**, then **Networking**, then **Open security group**, and select the security group being used to ensure the port 80 inbound rule is allowed access from anywhere as shown:

Commented [SS4]: Rephrase to > If the web page is not loading, go to **Task**, then **Networking**, then **Open security group**, and select the security group being used to ensure the port 80 inbound rule is allowed access from anywhere as shown:



Details

Security group name: default
Security group ID: sg-02fa3397a1b779841
Description: default VPC security group
VPC ID: vpc-0620bad368e94d930
Owner: 368355312556
Inbound rules count: 2 Permission entries
Outbound rules count: 1 Permission entry

Inbound rules (2)

Name	Security group rule...	IP version	Type	Protocol	Port range
–	sgr-0f76a4d54944d790f	–	All traffic	All	All
–	sgr-09bce28280b6c98af	IPv4	HTTPS	TCP	443

By following these steps, you have successfully executed the Fargate cluster to deploy a containerized application without managing the underlying infrastructure.