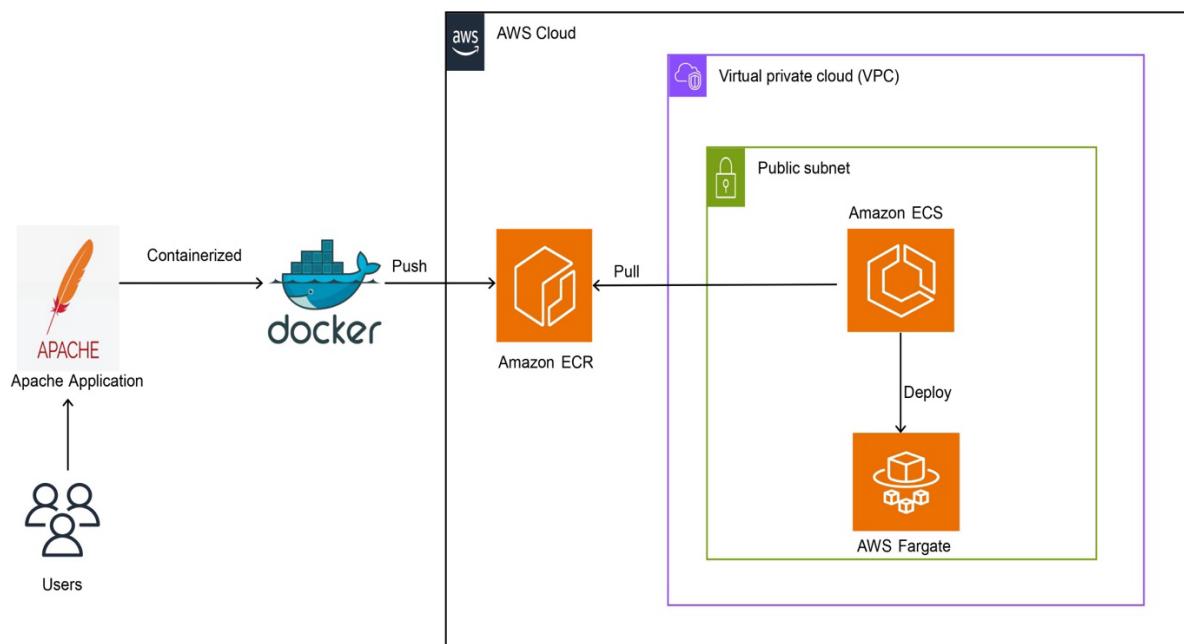


# Deploy Containerized Apache Application on ECS Using ECR



By  
**Mahendran Selvakumar**  
<https://devopstronaut.com/>



## 1. Create ECR Repository for ECS Container

Go to Amazon Elastic Container Registry and click "Create repository"

The screenshot shows the AWS ECR homepage. At the top, there's a navigation bar with the AWS logo, a search bar, and account information for Stockholm and Mahi. Below the navigation, the page title is "Amazon Elastic Container Registry". A large call-to-action button on the right says "Create a repository" with an orange "Create" button. To the left, there's a section titled "Share and deploy container software, publicly or privately". A small note at the bottom states: "Amazon Elastic Container Registry (ECR) is a fully managed container registry that makes it easy to store, manage, share and deploy your container images and artifacts anywhere." A "Pricing (US)" link is also visible.

Select "Private" and provide the repository name

The screenshot shows the "Create repository" form in the AWS ECR console. The URL in the browser is "Amazon ECR > Private registry > Repositories > Create repository". The main heading is "Create repository". Under "General settings", there are sections for "Visibility settings" (with "Private" selected), "Repository name" (containing "851725583489.dkr.ecr.eu-north-1.amazonaws.com/devopstronaut"), and "Tag immutability" (with "Disabled" selected). A note at the bottom states: "Once a repository has been created, the visibility setting of the repository can't be changed."



Click "Create repository"

**Image scan settings**

**Deprecation warning**  
The ScanOnPush configuration at the repository level has been deprecated in favour of registry-level scan filters.

**Scan on push**  
Enable scan on push to have each image automatically scanned after being pushed to a repository. If disabled, each image scan must be manually started to get scan results.  
 Disabled

**Encryption settings**

**KMS encryption**  
You can use AWS Key Management Service (KMS) (KMS) to encrypt images stored in this repository instead of using the default encryption settings.  
 Disabled

**i** The KMS encryption settings cannot be changed or disabled after the repository has been created.

**Cancel** **Create repository**

## 2. Create an IAM user with ECR permissions

Go to IAM and click "Create user"

**Identity and Access Management (IAM)**

**Users (1) Info**  
An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

User name	Path	Group	Last activity	MFA	Password age	Console
ansibledemouser	/	0	-	-	-	-

**Create user**

Provide a username



## Specify user details

### User details

User name

mahi

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = . @ \_ - (hyphen)

**Provide user access to the AWS Management Console - optional**

If you're providing console access to a person, it's a [best practice](#) to manage their access in IAM Identity Center.

**ⓘ If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user. [Learn more](#)**

Cancel

Next

Add the user to the admin group. Here, you can also restrict access with a custom policy

## Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

### Permissions options

**Add user to group**

Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

**Copy permissions**

Copy all group memberships, attached managed policies, and inline policies from an existing user.

**Attach policies directly**

Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

### User groups (1/1)

[Create group](#)

< 1 >

**Group name**

▲ | Users

▼ | Attached policies

▼ | Created

[adminaccess](#)

0

[AdministratorAccess](#)

2024-07-14 (16 days ago)

**► Set permissions boundary - optional**

Cancel

[Previous](#)

Next



Click "Create user"

## Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details		
User name mahi	Console password type None	Require password reset No

Permissions summary		
Name	Type	Used as
<a href="#">adminaccess</a>	Group	Permissions group

Tags - optional	
Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.	
No tags associated with the resource.	
<a href="#">Add new tag</a>	
You can add up to 50 more tags.	

[Cancel](#) [Previous](#) [Create user](#)

Go to the new user and create an Access key for AWS CLI

IAM > Users > mahi

mahi [Info](#) [Delete](#)

Summary		
ARN <a href="#">arn:aws:iam::851725583489:user/mahi</a>	Console access Disabled	Access key 1 <a href="#">Create access key</a>
Created July 31, 2024, 00:28 (UTC+01:00)	Last console sign-in -	
<a href="#">Permissions</a>	<a href="#">Groups (1)</a>	<a href="#">Tags</a> <a href="#">Security credentials</a> <a href="#">Access Advisor</a>

Console sign-in	
Console sign-in link <a href="https://851725583489.siginin.aws.amazon.com/console">https://851725583489.siginin.aws.amazon.com/console</a>	Console password Not enabled
<a href="#">Enable console access</a>	

Multi-factor authentication (MFA) (0)			
Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. <a href="#">Learn more</a>			
Type	Identifier	Certifications	Created on
No MFA devices. Assign an MFA device to improve the security of your AWS environment			
<a href="#">Assign MFA device</a>			

Access keys (0)		
Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. <a href="#">Learn more</a>		
<a href="#">Create access key</a>		



## Select "Command line interface (CLI)"

### Access key best practices & alternatives Info

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.

#### Use case

##### Command Line Interface (CLI)

You plan to use this access key to enable the AWS CLI to access your AWS account.

##### Local code

You plan to use this access key to enable application code in a local development environment to access your AWS account.

##### Application running on an AWS compute service

You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.

##### Third-party service

You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.

##### Application running outside AWS

You plan to use this access key to authenticate workloads running in your data center or other infrastructure outside of AWS that needs to access your AWS resources.

##### Other

Your use case is not listed here.



#### Alternatives recommended

- Use [AWS CloudShell](#), a browser-based CLI, to run commands. [Learn more](#)
- Use the [AWS CLI V2](#) and enable authentication through a user in IAM Identity Center. [Learn more](#)

#### Confirmation

I understand the above recommendation and want to proceed to create an access key.

[Cancel](#)

[Next](#)

Enter a description for the access key and click “create access key”

### Set description tag - *optional* Info

The description for this access key will be attached to this user as a tag and shown alongside the access key.

#### Description tag value

Describe the purpose of this access key and where it will be used. A good description will help you rotate this access key confidently later.

Maximum 256 characters. Allowed characters are letters, numbers, spaces representable in UTF-8, and: \_ . : / = + - @

[Cancel](#)

[Previous](#)

[Create access key](#)

Copy the access key and secret key, as the secret key value will not be visible again



## Retrieve access keys Info

### Access key

If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key	Secret access key
AKIA4MTWNASATFXBN3N	***** <a href="#">Show</a>

### Access key best practices

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the [best practices for managing AWS access keys](#).

[Download .csv file](#)[Done](#)

To verify if AWS CLI is installed and available on your system, you can run the following command in your terminal or command prompt. If AWS CLI is installed, this command will display the version information. If it is not installed, you'll receive an error message indicating that the command is not recognized

```
mahendranse...@Mahendrans-MBP ~ % which aws
/usr/local/bin/aws
mahendranse...@Mahendrans-MBP ~ % aws --version
aws-cli/2.17.20 Python/3.11.9 Darwin/23.2.0 exe/x86_64
mahendranse...@Mahendrans-MBP ~ %
```

To configure AWS CLI with your access credentials

```
mahendranse...@Mahendrans-MBP ~ % aws configure
AWS Access Key ID [None]: AKIA4MTWNASATFXBN3N
AWS Secret Access Key [None]: EN4VXD5xRVmkeMTAxsF0PxGJkhqr87gGzcsYNPZi
Default region name [None]: eu-north-1
Default output format [None]:
mahendranse...@Mahendrans-MBP ~ %
```

You'll be prompted to enter the following details:

- AWS Access Key ID: Enter your access key.
- AWS Secret Access Key: Enter your secret key.
- Default region name: Enter your preferred AWS region (e.g., us-west-2).



- Default output format: Enter your preferred output format (e.g., Json, text, or yaml).

Verify the configuration

```
mahendranelvakumar@Mahendrans-MBP ~ % aws sts get-caller-identity
{
    "UserId": "AIDA4MTWNASA3FRJ52CGR",
    "Account": "851725583489",
    "Arn": "arn:aws:iam::851725583489:user/mahi"
}
mahendranelvakumar@Mahendrans-MBP ~ %
```

### 3. Build the Docker image and push it to ECR

Create a Dockerfile to build a Docker image

```
↳ Dockerfile > ...
1  # Use the official Ubuntu image as a base
2  FROM ubuntu:20.04
3
4  # Set the timezone environment variable (replace 'Area/City' with the desired timezone, e.g., 'America/New_York')
5  ENV TZ=Europe
6
7  # Set the timezone
8  RUN ln -snf /usr/share/zoneinfo/$TZ /etc/localtime && echo $TZ > /etc/timezone
9
10 # Install Apache and tzdata, then clean up unnecessary files
11 RUN apt update && \
12     apt install -y apache2 tzdata && \
13     apt clean && \
14     rm -rf /var/lib/apt/lists/*
15
16 # Set environment variables for Apache
17 ENV APACHE_RUN_USER=www-data
18 ENV APACHE_RUN_GROUP=www-data
19 ENV APACHE_LOG_DIR=/var/log/apache2
20 ENV APACHE_RUN_DIR=/var/www/html
21
22 # Add a custom index.html
23 RUN echo 'Hello, Devopstronaut Family!' > /var/www/html/index.html
24
25
26
27 # Expose port 80 for Apache
28 EXPOSE 80
29
30 # Add a health check to monitor the container's health
31 HEALTHCHECK --interval=30s --timeout=10s --start-period=5s --retries=3 \
32     CMD curl -f http://localhost/ || exit 1
33
34 # Start Apache in the foreground
35 ENTRYPOINT ["apache2ctl", "-D", "FOREGROUND"]
```

Build docker image using this command (**docker build -t devopstronaut .**)



```
mahendranelvakumar@Mahendrans-MBP Docker % docker build --platform linux/amd64 -t devopstronaut .
[+] Building 51.6s (8/8) FINISHED
=> [internal] load build definition from Dockerfile
=> transferring dockerfile: 1.05kB
=> [internal] load metadata for docker.io/library/ubuntu:20.04
=> [internal] load .dockerignore
=> transferring context: 2B
=> CACHED [1/4] FROM docker.io/library/ubuntu:20.04@sha256:0b897358ff6624825fb50d20ffb605ab0eaea77ced0adb8c6a4b756513dec6fc
=> [2/4] RUN ln -snf /usr/share/zoneinfo/Europe /etc/localtime && echo Europe > /etc/timezone
=> [3/4] RUN apt update && apt install -y apache2 tzdata && apt clean && rm -rf /var/lib/apt/lists/*
=> [4/4] RUN echo 'Hello, Devopstronaut Family' > /var/www/html/index.html
=> exporting to image
=> exporting layers
=> => writing image sha256:cb0f5c642be53d725333d9de36ba372715f904851c72bb045c237157dbe0a6b0
=> => naming to docker.io/library/devopstronaut

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/nb0ivgb5729tq9ceoutqzwz3

What's next:
  View a summary of image vulnerabilities and recommendations → docker scout quickview
```

Retrieve an authentication token and authenticate your Docker client to your registry. Use the AWS CLI

```
aws ecr get-login-password --region eu-north-1 | docker login --username AWS --password-stdin 851725583489.dkr.ecr.eu-north-1.amazonaws.com
```

```
mahendranelvakumar@Mahendrans-MBP Docker % aws ecr get-login-password --region eu-north-1 | docker login --username AWS --password-stdin 851725583489.dkr.ecr.eu-north-1.amazonaws.com
Login Succeeded
mahendranelvakumar@Mahendrans-MBP Docker %
```

Tag your image so you can push the image to this repository

```
docker tag devopstronaut:latest 851725583489.dkr.ecr.eu-north-1.amazonaws.com/devopstronaut:latest
```

```
mahendranelvakumar@Mahendrans-MBP Docker % docker tag devopstronaut:latest 851725583489.dkr.ecr.eu-north-1.amazonaws.com/devopstronaut:latest
mahendranelvakumar@Mahendrans-MBP Docker % docker push 851725583489.dkr.ecr.eu-north-1.amazonaws.com/devopstronaut:latest
The push refers to repository [851725583489.dkr.ecr.eu-north-1.amazonaws.com/devopstronaut]
2f61c1b9a272: Pushed
7dc8fa8ac06a: Pushing [=====>] 96.08MB/146.7MB
373fa0b680e0: Pushed
a8c68591d421: Pushing [=====] 15.76MB/65.66MB
```

Run the following command to push this image to your newly created AWS repository

```
docker push 851725583489.dkr.ecr.eu-north-1.amazonaws.com/devopstronaut:latest
```

```
mahendranelvakumar@Mahendrans-MBP Docker % docker push 851725583489.dkr.ecr.eu-north-1.amazonaws.com/devopstronaut:latest
The push refers to repository [851725583489.dkr.ecr.eu-north-1.amazonaws.com/devopstronaut]
2f61c1b9a272: Pushed
7dc8fa8ac06a: Pushed
373fa0b680e0: Pushed
a8c68591d421: Pushed
latest: digest: sha256:3c300b8612412605a64ffae9d9c359355072ea99313704a600524c89b4af6917 size: 1155
mahendranelvakumar@Mahendrans-MBP Docker %
```

Verify that the docker image has been pushed



Amazon ECR > Private registry > Repositories > devopstronaut

## devopstronaut

[View push commands](#)

**Images (1)**

<input type="checkbox"/>	Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest
<input type="checkbox"/>	latest	Image	31 July 2024, 01:46:41 (UTC+01)	88.53	<a href="#">Copy URI</a>	<a href="#">sha256:3c300b86124126...</a>

## 4.Create ECS Cluster

Go to AWS ECS and click “Create Cluster”

Amazon Elastic Container Service > Clusters

## Clusters (0) [Info](#)

[Create cluster](#)

<input type="checkbox"/>	Cluster	Services	Tasks	Container instances	CloudWatch monitoring
No clusters No clusters to display					

Enter a cluster name and choose "AWS Fargate" so that the cluster is automatically configured for AWS Fargate



## Cluster configuration

Cluster name  
 Cluster name must be 1 to 255 characters. Valid characters are a-z, A-Z, 0-9, 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.  
 X

### ▼ Infrastructure Info Serverless

Your cluster is automatically configured for AWS Fargate (serverless) with two capacity providers. Add Amazon EC2 instances.

**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.

**i** External instances using **ECS Anywhere** can be registered after cluster creation is complete.

### ► Monitoring - *optional* Info

Container Insights is turned off by default. To change the default behaviour, use the CloudWatch Container Insights account setting. When you use Container Insights, there is a cost associated with it.

### ► Encryption - *new, optional*

Choose the KMS keys used by tasks running in this cluster to encrypt your storage.

### ► Tags - *optional* Info

Tags help you to identify and organise your clusters.

Cancel Create

The Cluster has been created successfully

Clusters (1) <span style="color: #0070C0; font-size: small;">Info</span>						
<span style="border: 1px solid #ccc; padding: 2px 10px; margin-right: 10px;">View cluster</span> <span style="border: 1px solid #ccc; border-radius: 50%; padding: 2px 5px; margin-right: 10px;">X</span>						
Cluster	Services	Tasks	Container instances	CloudWatch monitoring	Capacity provider strategy	
devopstronautcluster	0	No tasks running	0 EC2	Default	No default found	<span style="border: 1px solid #ccc; border-radius: 50%; padding: 2px 5px; margin-right: 10px;">Edit</span> <span style="background-color: #FFA500; color: white; border: 1px solid #FFA500; padding: 2px 10px; border-radius: 5px;">Delete</span>



## Create Task definition

Specify the task definition family and set the launch type to "AWS Fargate"

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.  
  
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.

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

**Network mode** Info

**Task size** Info  
Specify the amount of CPU and memory to reserve for your task.  
**CPU**   
**Memory**

**▼ Task roles - conditional**

**Task role** Info  
A task IAM role allows containers in the task to make API requests to AWS services. You can create a task IAM role from the [IAM console](#).

Enter the container name, provide the ECR image URI, and specify the port name



▼ 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.

<b>Name</b> <input type="text" value="mycontainer"/>	<b>Image URI</b> <input type="text" value="851725583489.dkr.ecr.eu-north-1.amazonaws.com"/>	<b>Essential container</b> <input type="checkbox"/> Yes
Up to 255 letters (uppercase and lowercase), numbers, hyphens, underscores, colons, periods, forward slashes, and number signs are allowed.		

**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. For port name, a default will be assigned if left blank.

Container port	Protocol	Port name	App protocol
80	TCP	httpd	HTTP
<a href="#">Add port mapping</a>			

**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](#)  
Container-level CPU, GPU and memory limits are different from task-level values. They define how many resources are allocated for the container. If the container attempts to exceed the memory specified by the hard limit, the container is terminated.

<b>CPU</b> <input type="text" value="1"/> in vCPU	<b>GPU</b> <input type="text" value="1"/>	<b>Memory hard limit</b> <input type="text" value="3"/> in GB	<b>Memory soft limit</b> <input type="text" value="1"/> in GB
---	--	---	---

▼ Environment variables - optional

**Environment variables** [Info](#)

**Add individually**  
Add a key-value pair to specify an environment variable.

You can verify the created task by checking the container name



Amazon Elastic Container Service > Task definitions > devopstask > Revision 1 > Containers

### devopstask:1

Deploy Actions Create new revision

**Overview Info**

ARN arn:aws:ecs:eu-north-1:851725583489:task-definition/devopstask:1	Status <span style="color: green;">ACTIVE</span>	Time created 31 July 2024 at 02:16 (UTC+1:00)	App environment FARGATE
Task role -	Task execution role ecsTaskExecutionRole	Operating system/Architecture Linux/X86_64	Network mode awsvpc

Containers JSON Task placement Volumes (0) Requires attributes Tags

**Task size**

Task CPU 1024 units (1 vCPU)	Task memory 3072 MiB (3 GB)
Task CPU maximum allocation for containers	
CPU (unit) 0 100 200 300 400 500 600 700 800 900 1000	Memory (MiB) 0 200 400 600 800 1000 1200 1400 1600 1800 2000 2200 2400 2600 2800 3000
<span style="color: blue;">█</span> mycontainer Shared task CPU	<span style="color: blue;">█</span> mycontainer Shared task memory

**Containers Info**

Container name	Image	Private registry	Essential	CPU	Memory hard/soft limit
mycontainer	851725583489.dkr.ecr.eu-north-1.amazonaws.com	-	Yes	0	-/-

## Create ECS Service

Go back to the "Deploy" section and select "Create service"

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

View task definition

Amazon Elastic Container Service > Task definitions > devopstask

### devopstask (1/1) Info

Deploy Actions Create new revision

Filter status Active

Task definition: revision devopstask:1 ACTIVE

Choose the existing cluster and select compute option as “capacity provider”



## Create Info

### Environment

AWS Fargate

Existing cluster

devopstronautcluster

▼ Compute configuration (advanced)

Compute options Info

To ensure task distribution across your compute types, use appropriate compute options.

Capacity provider strategy  
Specify a launch strategy to distribute your tasks across one or more capacity providers.

Launch type  
Launch tasks directly without the use of a capacity provider strategy.

**Capacity provider strategy Info**

Select either your cluster default capacity provider strategy or select the customised option to configure a different strategy.

Use cluster default  
No default capacity provider strategy configured for this cluster.

Use custom (Advanced)

Capacity provider	Base <small>Info</small>	Weight <small>Info</small>
FARGATE <input type="button" value="▼"/>	0 <input type="text"/>	1 <input type="text"/>

Platform version Info

Specify the platform version on which to run your service.

LATEST

Provide the Service name and specify the desired tasks to “1”



## Deployment configuration

### Application type [Info](#)

Specify what type of application you want to run.

#### Service

Launch a group of tasks handling a long-running computing work that can be stopped and restarted. For example, a web application.

#### Task

Launch a standalone task that runs and terminates. For example, a batch job.

Task application type is not applicable when launching a service.

### Task definition

Select an existing task definition. To create a new task definition, go to [Task definitions](#).

#### Specify the revision manually

Manually input the revision instead of choosing from the 100 most recent revisions for the selected task definition family.

#### Family

#### Revision

devopstask

1

### Service name

Assign a unique name for this service.

devopsservice

### Service type [Info](#)

Specify the service type that the service scheduler will follow.

#### Replica

Place and maintain a desired number of tasks across your cluster.

#### Daemon

Place and maintain one copy of your task on each container instance.

### Desired tasks

Specify the number of tasks to launch.

1

### ► Deployment options

### ► Deployment failure detection [Info](#)

In the Networking section, choose the VPC, subnet, and security group



## ▼ Networking

### VPC | Info

Choose the Virtual Private Cloud to use.

vpc-02aa3080a2b7a0b77

default



### Subnets

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

Choose subnets



Clear current selection

subnet-0f4e64ce0a813bf21 X

eu-north-1c 172.31.0.0/20

subnet-071184dc52011c698 X

eu-north-1b 172.31.32.0/20

subnet-0c7727afb8ff0c730 X

eu-north-1a 172.31.16.0/20

### 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.

Choose security groups



sg-086bbc46373ed734e X

default

### Public IP | Info

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

Turned on

It will take more time to create the service

devopsservice deployment is in progress. It takes a few minutes.

View in CloudFormation

Amazon Elastic Container Service > Clusters > devopstronautcluster > Services

devopstronautcluster

Cluster overview			
ARN arn:aws:ecs:eu-north-1:851725583489:cluster/devopstronautcluster	Status Active	CloudWatch monitoring <input checked="" type="radio"/> Default	Registered container instances -
Services	Draining -	Tasks Pending -	Running -
Encryption	Managed storage -	Fargate ephemeral storage	

Verify the task status after the service has been created



Amazon Elastic Container Service > Clusters > devopstronautcluster > Services > devops > Tasks

devops [Info](#)

[Health and metrics](#) [Tasks](#) [Logs](#) [Deployments](#) [Events](#) [Configuration and networking](#) [Tags](#)

Tasks (1/1)

Filter desired status: Running | Filter launch type: Any launch type

Task	Last status	Desired st...	Task defin...	Health status	Started at	Container instan...	Launch type	Pla...
<a href="#">22c29c033ac64087ad573c6911e7c9e7</a>	<span>Running</span>	<span>Running</span>	<a href="#">devopstask:4</a>	<span>Unknown</span>	1 minute ago	-	FARGATE	1.4

Containers for task 22c29c033ac64087ad573c6911e7c9e7

Containers (1)

Container name	Container runtime ID	Image URI	Image Digest	Status	Health status
mycontainer	<a href="#">22c29c033ac6408...</a>	<a href="#">851725583489.dkr.ecr.eu-north-1.a...</a>	<a href="#">sha256:7a29aba7e182e7884...</a>	<span>Running</span>	<span>Unknown</span>

Go to Tasks configuration and Copy the Public IP Address from Task

Amazon Elastic Container Service > Clusters > devopstronautcluster > Services > devops > Tasks > [22c29c033ac64087ad573c6911e7c9e7](#) > Configuration

22c29c033ac64087ad573c6911e7c9e7

[Configuration](#) [Logs](#) [Networking](#) [Volumes \(0\)](#) [Tags](#)

**Task overview**

ARN <a href="#">arn:aws:ecs:eu-north-1:851725583489:task/devopstronautcluster/22c29c033ac64087ad573c6911e7c9e7</a>	Last status <span>Running</span>	Desired status <span>Running</span>	Started/created at 31 July 2024 at 03:54 (UTC+1:00) 31 July 2024 at 03:54 (UTC+1:00)
---	-------------------------------------	--	--

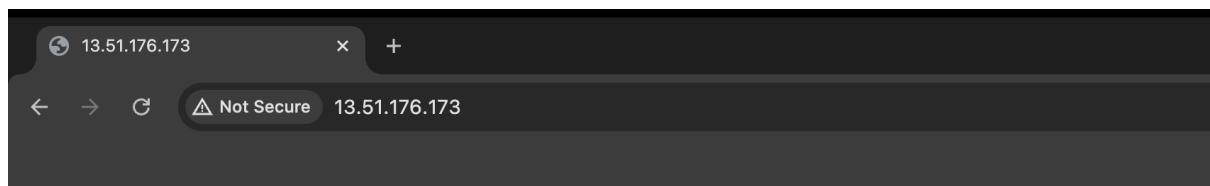
Fargate ephemeral storage

Encryption <a href="#">Info</a> Default AWS Fargate encryption	Size (GiB) 20
---	------------------

**Configuration**

Operating system/Architecture Linux/X86_64	Capacity provider FARGATE	ENI ID <a href="#">eni-0be63908f6928024e</a>	Public IP <a href="#">13.51.176.173</a>   <a href="#">open address</a>
CPU   Memory 1 vCPU   3 GB	Launch type FARGATE	Network mode awsvpc	Private IP <a href="#">172.31.16.96</a>
Platform version 1.4.0	Container instance IDs: -	Subnet ID <a href="#">subnet-0c7727afb8ff0c730</a>	MAC address <a href="#">06:ad:d6:79:f4:d1</a>
	Task definition: revision <a href="#">devopstask:4</a>		
	Task group service:devops		

Paste the IP address to the browser and now you should be able to access the Apache website from AWS ECS.



Hello, Devopstronaut Family

Keep Learning, Keep Automating!!!

Feel free to reach out to me, if you have any other queries or suggestions

Stay connected on Linkedin <https://www.linkedin.com/in/mahendran-selvakumar-36444a77/>