

## Problem Statement:

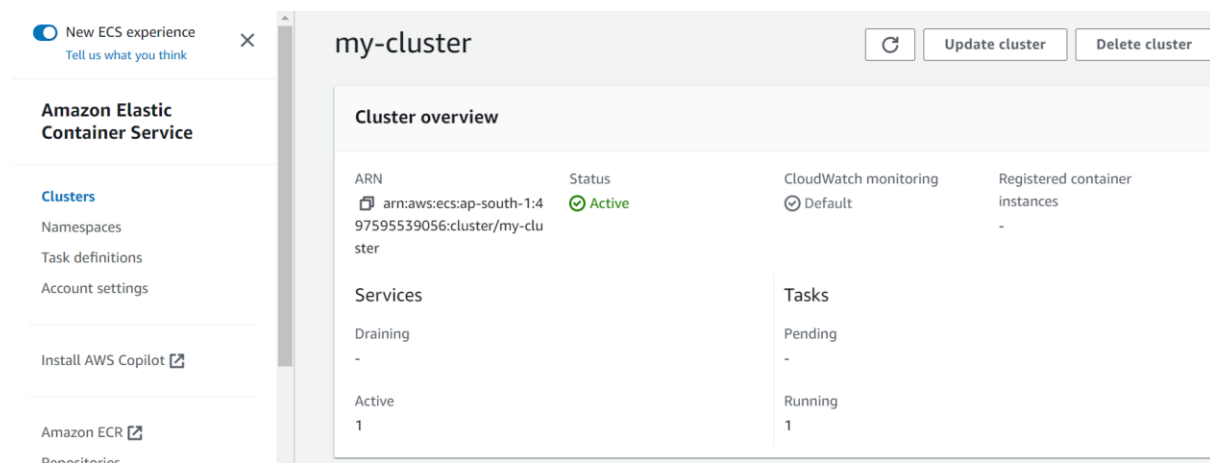
1. Deploy a simple HTTP Server hosted in a Docker image to AWS ECS (Fargate deployment) fronted by an Application Load Balancer.
2. Describe the further steps to map a Custom Domain / Sub-domain name for this application.

\*\*\*\*\*I have used Python Boto3 for aws-sdk and Nginx for HTTP Server\*\*\*\*\*

Importing Boto3: The script starts by importing the Boto3 library, which provides access to AWS services.

Function Definitions:

**create\_cluster():** Creates an ECS cluster named 'my-cluster'.



The screenshot displays the Amazon ECS console interface. On the left is a navigation sidebar with the 'Amazon Elastic Container Service' header and links for 'Clusters', 'Namespaces', 'Task definitions', 'Account settings', 'Install AWS Copilot', and 'Amazon ECR Repositories'. The main panel is titled 'my-cluster' and includes buttons for 'Update cluster' and 'Delete cluster'. Below the title is a 'Cluster overview' section with a table of cluster details.

ARN	Status	CloudWatch monitoring	Registered container instances
arn:aws:ecs:ap-south-1:497595539056:cluster/my-cluster	Active	Default	-

Services	Tasks
Draining -	Pending -
Active 1	Running 1

**register\_task\_definition():** Registers a task definition for the ECS service. It defines a container (Nginx in this case) with specific configurations like CPU, memory, networking, and execution roles.

New ECS experience  
Tell us what you think

Amazon Elastic Container Service

Clusters  
Namespaces  
**Task definitions**  
Account settings

Install AWS Copilot  
Amazon ECR

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

taskdef:1

Deploy Actions Create new revision

**Overview** Info

ARN arn:aws:ecs:ap-south-1:497595539056:task-definition/taskdef:1	Status ACTIVE	Time created 2023-11-30T03:36:04.106Z	App environment FARGATE
Task role -	Task execution role ecsTaskExecutionRole	Operating system/Architecture -	Network mode awsvpc

**create\_load\_balancer():** Creates an application load balancer (ALB) named 'alb-test' in an internet-facing scheme.

EC2 Dashboard  
EC2 Global View  
Events

▼ Instances  
Instances  
Instance Types  
Launch Templates  
Spot Requests  
Savings Plans  
Reserved Instances  
Dedicated Hosts  
Capacity Reservations  
New

▼ Images  
AMIs  
AMI Catalog

EC2 > Load balancers > alb-test

alb-test

Refresh Actions

▼ Details

Load balancer type Application	Status Provisioning	VPC vpc-0fbcaed020cc53d30	IP address type IPv4
Scheme Internet-facing	Hosted zone ZP97RAFLXTNZK	Availability Zones subnet-0150e21d640382ae9 ap-south-1b (aps1-az3) subnet-09a00c70dac535df0 ap-south-1a (aps1-az1)	Date created November 29, 2023, 19:36 (UTC-08:00)
Load balancer ARN arn:aws:elasticloadbalancing:ap-south-1:497595539056:loadbalancer/app/alb-test/948838fb13538da0		DNS name alb-test-738277539.ap-south-1.elb.amazonaws.com (A Record)	

**create\_target\_group():** Creates a target group 'tg-test' associated with the ALB for routing traffic.

AMI Catalog

▼ Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

▼ Network & Security

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

▼ Load Balancing

Load Balancers

Target Groups

Trust Stores [New](#)

tg-test

Actions ▼

Details

arn:aws:elasticloadbalancing:ap-south-1:497595539056:targetgroup/tg-test/d2474ec48c6d90dc

Target type	Protocol : Port	Protocol version	VPC
IP	HTTP: 80	HTTP1	<a href="#">vpc-0fbcaed020cc53d30</a>
IP address type	Load balancer		
IPv4	<a href="#">alb-test</a>		

1	1	0	0	0	0
Total targets	Healthy	Unhealthy	Unused	Initial	Draining
	0 Anomalous				

► Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.

**create\_listener():** Creates a listener on the ALB to forward traffic to the target group.

Listeners and rules (1) [Info](#)

↻

Manage rules ▼

Manage listener ▼

Add listener

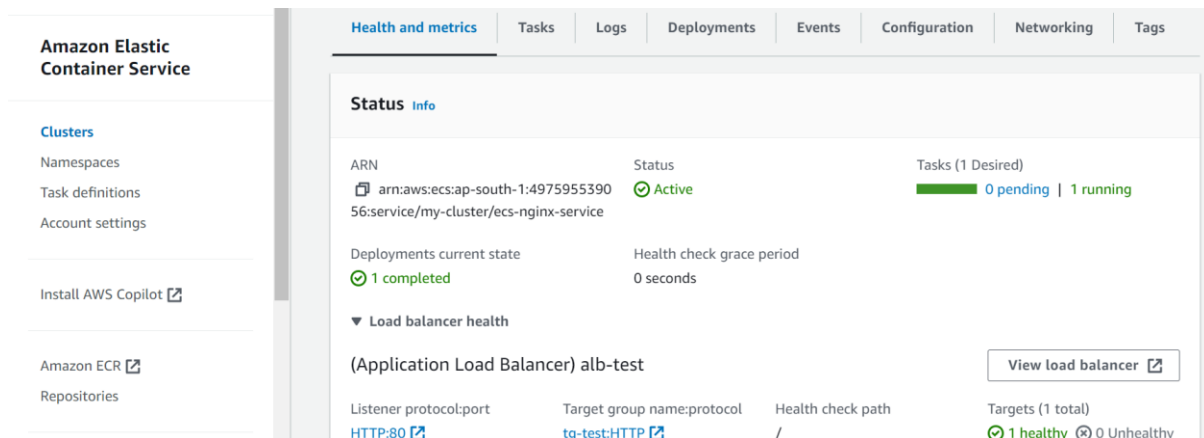
A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

🔍 Filter listeners

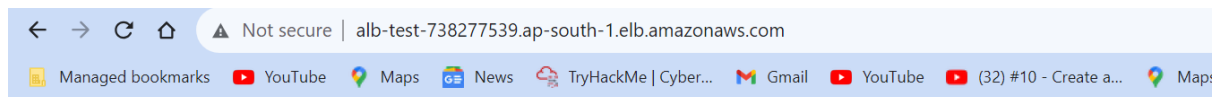
< 1 > ⚙️

<input type="checkbox"/>	Protocol:Port ▼	Default action ▼	Rules ▼	ARN ▼	Security
<input type="checkbox"/>	<a href="#">HTTP:80</a>	<div>Forward to target group<ul style="list-style-type: none"><li><a href="#">tg-test</a>: 1 (100%)</li><li>Group-level stickiness: Off</li></ul></div>	<a href="#">1 rule</a>	<div><div>📄</div>ARN</div>	Not appli

**create\_service():** Creates an ECS service named 'ecs-nginx-service' with the defined task definition and configurations. It associates the service with the ALB and the target group



After everything has been created we can access the Http server by using ALB DNS Name.



## 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](https://nginx.org).  
Commercial support is available at [nginx.com](https://nginx.com).

*Thank you for using nginx.*