

AWS ECS & EventBridge

❖ Table of Content

- AWS ECS Fargate
 - AWS EventBridge
 - AWS CloudWatch
 - Dockerfile for Java Based Application
-

❖ AWS ECS Fargate

▪ Introduction to ECS

AWS ECS (Elastic Container Service) Fargate is a serverless compute engine for containers. It allows you to run containers without managing the underlying infrastructure. With ECS Fargate, you can focus on deploying and scaling your applications without worrying about provisioning, scaling, or managing servers.

We don't need to install or manage any packages. We can simply deploy our application and use it. ECS Fargate will take care for underlying servers capacity, patching etc.

▪ Component of ECS

- ✓ Cluster
- ✓ Task Definitions
- ✓ Tasks
- ✓ Services

▪ AWS Fargate vs AWS EC2 Instance

AWS Fargate	AWS EC2 Instance
AWS Fargate is a serverless compute engine which is managed by Fargate	AWS EC2 Instance is a standalone VM and it's completely managed by customers.
Not a complete control on Fargate container	Complete control on EC2 Instance
AWS Fargate suitable for short-running tasks	EC2 Instance suitable for long-running task/container
Charges is like pay-as-you-go model	Charges is like pay-as-you-go model
Supported OS – Linux, Windows	Supported OS – Linux, Windows, Mac etc.

▪ Costing

AWS Fargate pricing is calculated based on the vCPU, memory, Operating Systems, CPU Architecture¹, and storage resources used from the time you start to download your container image until the Amazon ECS Task terminates. Pay as you go model.

You may get additional charges if containers use other AWS services or transfers data.

For example, if your containers use Amazon CloudWatch Logs for application logging, you are billed for CloudWatch usage.

Linux/X86 Linux /ARM Windows/X86	
Region: Asia Pacific (Mumbai) ▾	
	Price
per vCPU per hour	\$0.04256
per GB per hour	\$0.004655

Ref: <https://aws.amazon.com/fargate/pricing/>

▪ Setup

- **Cluster:** Group of task and services to run ECS container
- **Task Definitions:** A task definition is a text file or JSON that describes a container configuration details like CPU, memory, port mapping and registry details. It's a container configuration.
- **Task:** A Task is created when you run a Task directly, which launches container(s) (defined in the task definition) until they are stopped or exit on their own, at which point they are *not replaced automatically*. Running Tasks directly is ideal for short-running jobs, perhaps as an example of things that were accomplished via CRON.
- **Service:** A Service is used to guarantee that you always have some number of Tasks *running at all times*. If a Task's container exits due to an error, or the underlying EC2 instance fails and is replaced, the ECS Service will replace the failed Task.
Services are typically used for long-running applications like web servers.

❖ AWS EventBridge

▪ Introduction to EventBridge

- Amazon EventBridge is a serverless, fully managed, and scalable event bus that enables integrations between AWS services, Software as a services (SaaS), and your applications.
- EventBridge was formerly called as AWS CloudWatch Events.
- EventBridge delivers a stream of real-time data from your applications, SaaS applications, and AWS services to targets such as AWS Lambda functions, HTTP invocation endpoints using API destinations, or event buses in other AWS accounts.

▪ How EventBridge Works?

- Scheduled Event
- Event Pattern

▪ Costing

<https://aws.amazon.com/eventbridge/pricing/>

❖ AWS CloudWatch

▪ Introduction to CloudWatch

▪ Costing

<https://aws.amazon.com/cloudwatch/pricing/>

- Metrics vs Logs
- Log Insights Query

```
fields @timestamp, @message, @logStream, @Log
| sort @timestamp desc
| filter @message like /ERROR/
| limit 200
```

- Sample Dockerfile to Run Java Application

```
# Use a base image with Java installed
FROM openjdk:11-jre-slim

# Set the working directory inside the container
WORKDIR /app

# Copy the JAR file from the host into the container
COPY target/my-application.jar /app/my-application.jar

# Expose the port on which the application will listen
EXPOSE 8080

# Specify the command to run the application
CMD ["java", "-jar", "my-application.jar"]
```

Some Useful Links:

For AWS ECS Farget:

<https://docs.aws.amazon.com/AmazonECS/latest/developerguide/Welcome.html>

<https://aws.amazon.com/ecs/faqs/>

For AWS EventBridge:

<https://docs.aws.amazon.com/eventbridge/latest/userguide/eb-what-is.html>

For CloudWatch:

<https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/WhatIsCloudWatch.html>