[Nagarro]-[Ecommerce]

DAR Document

Nagarro Software Pvt. Ltd.

Akshat Aggarwal

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision History** | | | |
| Version | Date | Author/Contributor | Comments |
| 1.0 | 09/02/2025 | Akshat Aggarwal | Initial Draft |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[1 Introduction 4](#_Toc421883713)

[1.1 Objective and scope of document 4](#_Toc421883714)

[2 Requirements at a Glance 5](#_Toc421883715)

[3 Available tools 6](#_Toc421883716)

[3.1 AWS Fargate with ECS 6](#_Toc421883717)

[3.1.1 Features 6](#_Toc421883718)

[3.1.2 Pricing 6](#_Toc421883719)

[3.2 AWS EKS 6](#_Toc421883720)

[3.2.1 Features 6](#_Toc421883721)

[3.2.2 Pricing 6](#_Toc421883722)

[4 Comparison Analysis 7](#_Toc421883723)

[4.1 Point Matrix 7](#_Toc421883724)

[4.2 <Comparison 1> 7](#_Toc421883725)

[4.3 <Comparison 2> 7](#_Toc421883726)

[5 Recommendation 8](#_Toc421883727)

[6 Assumptions 9](#_Toc421883728)

[7 Risks 10](#_Toc421883729)

[8 Appendix 11](#_Toc421883730)

[8.1 References 11](#_Toc421883731)

# Introduction

This document provides a detailed analysis of the deployment options for the e-commerce application services. The two deployment options under consideration are **AWS ECS with Fargate** and **AWS EKS**. The goal is to determine the most suitable deployment strategy based on features, cost, scalability, and ease of management.

## Objective and scope of document

The objective of this document is to compare **AWS ECS with Fargate** and **AWS EKS** for deploying a microservices-based e-commerce application. The scope includes evaluating the features, pricing, scalability, and management complexity of both options to make an informed decision.

# Requirements at a Glance

* **Scalability**: The application should scale seamlessly based on traffic demands.
* **Cost Efficiency**: The deployment option should be cost-effective, especially during low-traffic periods.
* **Ease of Management**: The deployment option should require minimal operational overhead.
* **Integration**: The solution should integrate well with other AWS services like RDS, S3, and CloudWatch.
* **High Availability**: The application should be highly available with minimal downtime.

# Available tools

* AWS ECS (Elastic Container Service) with Fargate
* AWS EKS (Elastic Kubernetes Service)

## AWS ECS

AWS Fargate is a serverless compute engine for containers that works with Amazon Elastic Container Service (ECS). It allows you to run containers without managing the underlying infrastructure.

### Features

* **Serverless**: No need to manage EC2 instances.
* **Autoscaling**: Automatically scales based on demand.
* **Integration with AWS Services**: Seamless integration with ALB, RDS, S3, and CloudWatch.
* **Security**: Built-in security with IAM roles and VPC support.
* **Ease of Use**: Simplified container management with ECS.

### Pricing

* **Fargate Pricing**: Pay for vCPU and memory resources used by your containers.
* **ECS Pricing**: No additional charge for ECS; you only pay for the AWS resources you create.

## AWS EKS

Amazon Elastic Kubernetes Service (EKS) is a managed Kubernetes service that allows you to run Kubernetes on AWS without needing to manage the control plane.

### Features

* **Kubernetes Management**: Full support for Kubernetes features like deployments, services, and ingress.
* **Scalability**: Automatically scales Kubernetes worker nodes.
* **Integration with AWS Services**: Works well with ALB, RDS, S3, and CloudWatch.
* **Security**: IAM integration, VPC support, and encryption.
* **Flexibility**: Supports advanced Kubernetes configurations and customizations.

### Pricing

* **EKS Pricing**: $0.10 per hour for each EKS cluster.
* **Worker Node Pricing**: Pay for EC2 instances or Fargate for worker nodes.

# Comparison Analysis

This section presents a comparative analysis of the features of AWS ECS and EKS based on key performance metrics relevant to the e-commerce platform.

## Weightage Matrix

|  |  |
| --- | --- |
| Feature | Points |
| Ease of Management | 5 |
| Cost Efficiency | 4 |
| Scalability | 5 |
| Flexibility | 4 |
| Security | 4 |

## Comparison 1: Ease of Maintenance, Scalability and Security

|  |  |  |
| --- | --- | --- |
| Feature | AWS ECS with Fargate | AWS EKS |
| Ease of Maintenance | **5** | **3** |
| Scalability | **5** | **4** |
| Security | **5** | **5** |

## Comparison 2: Cost Efficiency

|  |  |  |
| --- | --- | --- |
| Feature | AWS ECS with Fargate | AWS EKS |
| Cost Efficiency | **5** | **3** |
| Flexibility | **5** | **3** |

# Recommendation

Based on the analysis, **AWS ECS with Fargate** is recommended for the deployment of the e-commerce application. It offers better ease of management, cost efficiency, and seamless integration with AWS services, making it a more suitable choice for a microservices-based application that requires minimal operational overhead.

# Assumptions

1. The application will experience variable traffic, requiring auto-scaling capabilities.
2. The team has limited Kubernetes expertise, making EKS more complex to manage.
3. Cost optimization is a priority, especially during low-traffic periods.
4. High availability and security are critical requirements.

# Risks

1. **AWS Fargate**: Limited customization options compared to EKS.
2. **AWS EKS**: Higher operational complexity and potential for increased costs if not managed properly.
3. Both options require proper monitoring and logging to ensure optimal performance.

# Appendix

## References

1. AWS Fargate Documentation: <https://aws.amazon.com/fargate/>
2. AWS EKS Documentation: <https://aws.amazon.com/eks/>
3. AWS Pricing Calculator: <https://calculator.aws/>