ECS & ECR & EKS

Introduction to Container Orchestration

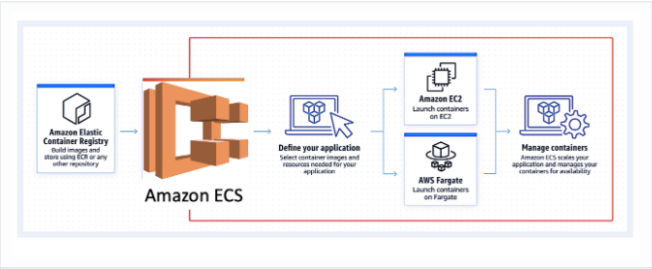
Container Orchestration

In the container environment, you may have many containers to run, manage, and scale. We called this process as **container orchestration**. Container orchestration is an automated process for handling or managing the work of individual containers for applications operating on several cluster microservices.

There are many options for orchestration. One of the most popular container-orchestration system is **Kubernetes**. It is an open-source container orchestration system for automating application deployment, scaling, and management. And also, you can choose **Docker Swarm** of Docker, Inc which is another popular option for the orchestration process.

As for AWS, AWS offers **AWS Amazon Elastic Container Service (Amazon ECS)** as a solution of container orchestration

Amazon Elastic Container Service (Amazon ECS)



Amazon ECS

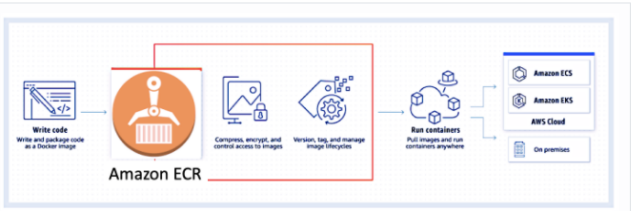
Amazon Elastic Container Service (Amazon ECS) is a fully managed container orchestration service. It allows you to easily run, scale, and secure **Docker** container applications on AWS.

Shortly, it is a service that handles the orchestration and provisioning of Docker containers.

Amazon ECS enables you to grow from a single container to thousands of containers across hundreds of instances without creating additional complexity

As you see in the picture above there are two options while creating the Amazon ECS. With Amazon ECS, you can use **AWS Fargate** to fully manage your infrastructure and you no longer have to select Amazon EC2 instance types, provision and scale clusters, or patch and update each server. Or, you can choose **EC2 launch type**, then you create manually the clusters that are a group of container instances you manage.

Amazon Elastic Container Registry (ECR)



Amazon ECR

Before we cover ECR, we need to understand the differences between **Registry and Repository**.

A registry is a list of an image with pointers for where to find the items, like an index. Container registries store multiple repositories of container images, as well as storing API paths and access control rules just like Dockerhub.

But, as for repository, it is where you can store one or more versions of a specific image. A repository is a place for you to publish and access your Docker images. It stores actual images.

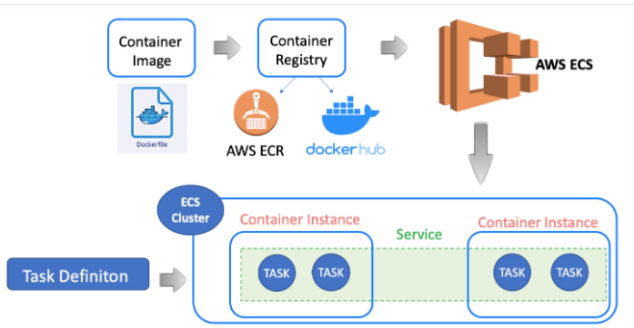
You may assume that the registry consists of repositories and the repository contains multiple copies of the same image that are uniquely versioned with tags. Docker Hub and other third party repository hosting services are called "registries".

Amazon Elastic Container Registry (ECR) service is a container registry solution of AWS. It is a collection of repositories made to store Docker container images.

It is a fully-managed Docker container registry that helps you to store, manage, and deploy Docker container images and it is integrated with Amazon Elastic Container Service (ECS).

An Amazon ECR registry is provided to each AWS account; you can create image repositories in your registry and store images in them.

Component and Terms of ECS



ECS Components

While creating an ECS we use a variety of components in AWS as you see in the ECS architecture above. To become familiar with these components, let's take a look at these terms below;

- **Containers and Images:** To deploy applications on Amazon ECS, your application components must be architected to run in containers. Containers are created from a read-only template called an image.
- **A Container Registry:** It is a collection of repositories made to store Docker container images. You may use the Dockerhub or **Amazon Elastic Container Registry (ECR)**.
- **Task Definition:** This is a guide that describes how to launch a docker container. It is a kind of blueprint. It contains settings like port, docker image, CPU shares, memory requirement, a command to run, etc.
- **Task:** This is a running container that has the settings specified in the task description. It can be thought of as an "instance" in the definition of a task. Shortly, it is a product of task definition.
- **Service:** It is a group of a container using the same Task Definition.
- **Cluster:** A logic group of EC2 instances. When you run tasks using Amazon ECS, you place them on a cluster, which is a logical grouping of resources.
- **Container Instance:** Container Instance is an EC2 instance located in ECS Cluster.

Amazon Elastic Kubernetes Service (EKS)

As you know, Kubernetes is an open-source container orchestration system. It was originally founded by Google and is now maintained by the Cloud Native Computing Foundation.

AWS ensures that the customer can use the worldwide popular program, Kubernetes, in its own environment via Amazon EKS.

Amazon Elastic Kubernetes Service (EKS) is a managed Kubernetes service that makes it easy for you to run Kubernetes on AWS without needing to install, operate, and maintain your own Kubernetes control plane. Amazon EKS is a certified Kubernetes conformant, so existing applications running on upstream Kubernetes are compatible with Amazon EKS.

Amazon EKS automatically manages the availability and scalability of the Kubernetes control plane nodes that are responsible for starting and stopping containers, scheduling containers on virtual machines, storing cluster data, and other tasks. Amazon EKS automatically detects and replaces unhealthy control plane nodes for each cluster.

Complementary Lesson about AWS ECS