Elastic Container Service

It is a container orchestrator and its supports only docker container. We have other orchestrators also available like Kubernetes service and which is open-source orchestrator provided by Google. If there are multiple containers and you have to scale individual containers independently things became really complicated. So far that we need container orchestrator. It’s an abstraction layer provided by AWS. What it will do is it will abstract all the scaling load, balancing and auto scaling for the end user. And for that we have to manage only orchestrator.

We can manage one or more containers by using ECS. So it’s a Container Management service.

AWS native CI CD – code commit, code pipeline and code build.

Also, we have AWS provided Kubernetes that is Elastic Kubernetes service. But ECS is AWS’s native container orchestrator.

ECS Fargate which is a server less cluster management by ECS.

ECS contains a few components and those components are the building blocks of ECS. They are all interdependent.

1. **ECS cluster** – It’s a pool of servers in the case of AWS its combination of one or more EC2 instances. We can scale using auto scaling so the entire cluster can be scaled. Individual containers can also be scaled. These are not the normal EC2 instances. When you launch ECS cluster, you have to launch it from container optimized or ECS optimized ami’s. These AMIs contain docker daemon by default. And then we have ECS agent which is also daemon running in each individual EC2 instances. These EC2 instances are called ECS container instances.
2. **Task Definition**: A Task definition contains multiple subcomponents or multiple definitions. That means you have container volume and network information. These information are provided in the docker run command. Whenever we ran docker we provided container images. We provided volume information, network information. Task definition contain one or more containers also. After task definition we run task.
3. **Task:** Task is a single instance of task definition and its logical group of one or more containers which are related to each other. You can have multiple task definitions and you can run multiple tasks also. Now how do you decide Which container to run which container instance. So this part will take care by ECS scheduler. ECS agent will talk to the ECS service if anyone of the task is down ECS service will take care of that.
4. **ECS Service:** ECS will talk to the ECR and task definitions also will talk to ECR because it provides container images in task definition file.

We have two types of launch types here.

1. EC2
2. Fargate
3. **Service :** Service is combination of one or more tasks. You can have one task or multiple tasks running under one service or multiple services.

**Registry:** This will provided to each AWS account. We can create image repositories in the registry and store images in them.

**Authentication Token:** This token required Docker Cli to authenticate to Amazon ECR

ECR login through AWS CLI commands:

aws ecr get-login –region us-east-1 –no-include-email

docker login -u AWS -p <paste the token from above command>

**Capacity Provider:**

Needed to scale infrastructure for tasks running in the cluster. A single cluster can have more than one capacity provider.

Auto scaling Group (for EC2 launch type)

ECS Fargate and Fargate spot – difference is similar to EC2 on demand and EC2 spot instances.











