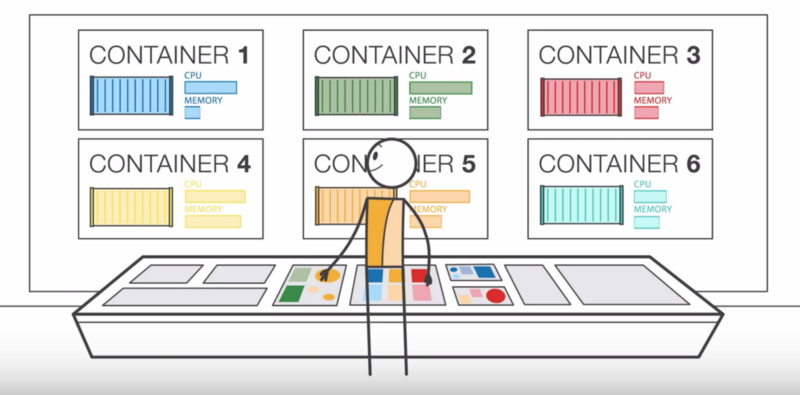
Overview

Amazon Elastic Container Service (ECS)

**Amazon Elastic Container Service (Amazon ECS)**, as defined by AWS, is a highly scalable container management service that easily runs, stops, or manages docker containers in a cluster. You can host a serverless infrastructure by running a service or task using the Fargate launch type or using the EC2 launch type to run EC2 instances. **Amazon ECS** compared to Kubernetes, Docker Swarm and Azure Container Instances.

\

Amazon ECS runs containers in a cluster of multiple Amazon EC2 instances with Docker pre-installed. This service handles container installation, scaling, monitoring, and management of these instances (launch/stop) through both the API and the AWS Management Console.

Amazon Elastic Container Service allows you to simplify the view of EC2 instances into a pool of resources, such as CPU and memory.

You can use Amazon ECS to install containers across the cluster based on the resources you need, policy independence, or scalability. With Amazon ECS, you don’t have to run your cluster management and configuration management systems or worry about expanding your management infrastructure.

Amazon ECS is a region-based service that simplifies running application containers across multiple AZs in the same Region. You can create an ECS cluster inside a new or old VPC. After a cluster is up and running, you can define tasks and services that specify the Docker container image to run through the clusters.

Task Definition

This is a text file (json format). It will describe one or more containers (up to 10) that make up your application. The task definition will specify some parameters for the application, such as which container should be used, the image to be used, the launch type to be used, the configuration of the container (CPU and memory), the open port, and what data volume. will be created with the container in the tasks…

The parameter in the task definition depends on what launch type is being used. The following example:

{

"containerDefinitions": [

{

"command": [

"/bin/sh -c \"echo '<html> <head> <title>Amazon ECS Sample App</title> <style>body {margin-top: 40px; background-color: #333;} </style> </head><body> <div style=color:white;text-align:center> <h1>Amazon ECS Sample App</h1> <h2>Congratulations!</h2> <p>Your application is now running on a container in Amazon ECS.</p> </div></body></html>' > /usr/local/apache2/htdocs/index.html && httpd-foreground\""

],

"entryPoint": [

"sh",

"-c"

],

"essential": true,

"image": "httpd:2.4",

"logConfiguration": {

"logDriver": "awslogs",

"options": {

"awslogs-group" : "/ecs/fargate-task-definition",

"awslogs-region": "us-east-1",

"awslogs-stream-prefix": "ecs"

}

},

"name": "sample-fargate-app",

"portMappings": [

{

"containerPort": 80,

"hostPort": 80,

"protocol": "tcp"

}

]

}

],

"cpu": "256",

"executionRoleArn": "arn:aws:iam::012345678910:role/ecsTaskExecutionRole",

"family": "fargate-task-definition",

"memory": "512",

"networkMode": "awsvpc",

"runtimePlatform": {

"operatingSystemFamily": "LINUX"

},

"requiresCompatibilities": [

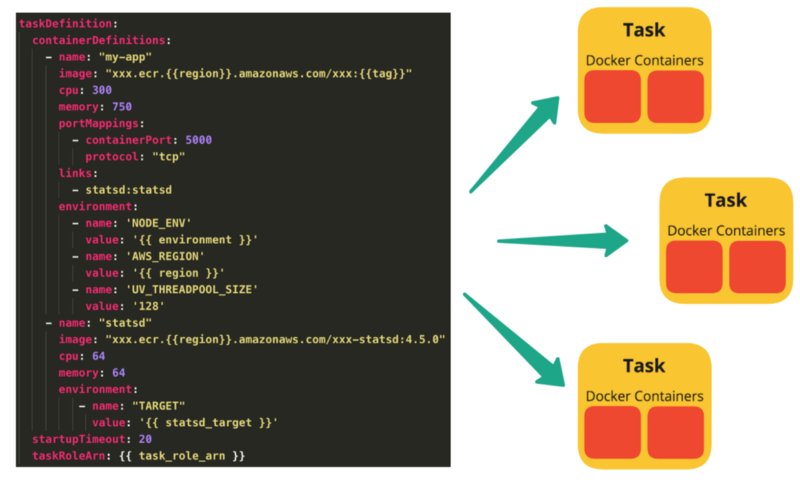
"FARGATE"

]

}

Task and Schedule

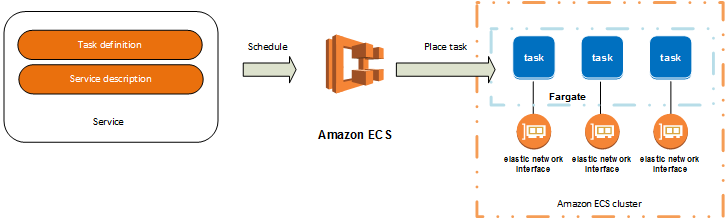
A task is the instantiation of a task definition within the cluster. Many different Tasks can be created by a Task Definition, depending on the need, but specifying a certain number of tasks. However, these tasks can be the same.



Each task that uses the Fargate launch type has a separate boundary and does not share kernel, CPU resources, memory, or elastic network interface with other tasks.

The Amazon ECS task scheduler is responsible for replacing tasks within the cluster. There are a few different ways to schedule tasks.

* Service Schedule
* Manually running task
* Running tasks on a cron-like schedule
* Custom scheduler



Service

Specifies the min and max of one or more Tasks from a Task Definition running at any given time. This is a scaling and load balancing feature. Now that the Service is in place, the Service’s Tasks need to be run somewhere to be accessible. It needs to be located on a Cluster and the container management service will handle it when running on one or more ECS Container instances (s).

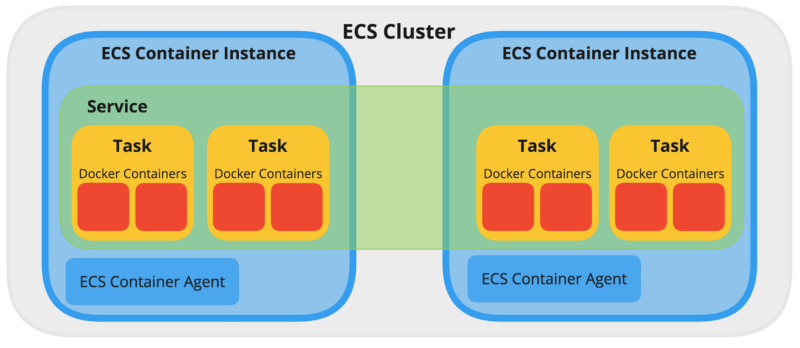
Amazon ECS Container Instances and Amazon ECS Container Agents

An ECS Container Instance can run many of the same or different Tasks, from the same or different Services. Agents are used to supporting exchange connections between ECS and instances, provide information about running containers, and manage newly created containers…

Cluster

A cluster is a group of ECS Container Instances. Amazon ECS handles the logic of scheduling, maintaining, and handling scaling requests for these instances. Tasks running on ECS are always in the cluster.

When Tasks are run on Fargate, the resources of the cluster are managed by Fargate. When using the EC2 launch type, the clusters are groups of ECS Container Instances (run on EC2 instances).



Amazon ECS loads your container images from the registry you previously set up and then runs these images in your cluster:

* Cluster is Region-specific.

The cluster can contain multiple tasks using both Fargate launch type and EC2 launch type.

When tasks use the EC2 launch type, clusters can contain many different ECS Container Instances. At the same time, an ECS Container Instance belongs to only one cluster.

You can create a custom IAM policy for the cluster to allow or limit user access to clusters.

