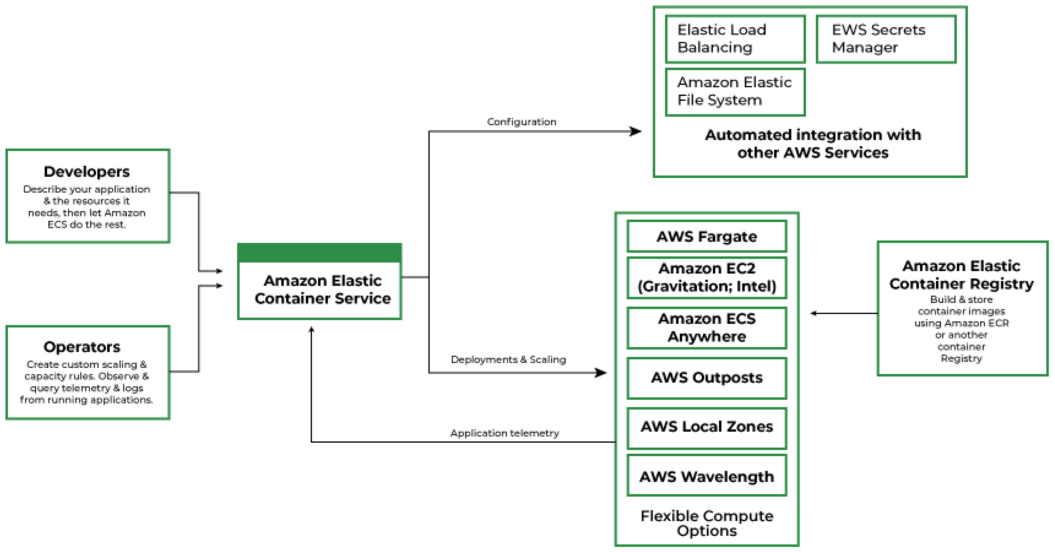
NAME- SHUBHAM KUMAR
ROLL NO.- 32

Introduction to Amazon Elastic Container Service (ECS)

Amazon Elastic Container Service (ECS), also known as Amazon EC2 Container Service, is a managed service that allows users to run Docker-based applications packaged as containers across a cluster of EC2 instances. Running simple containers on a single EC2 instance is simple but running these applications on a cluster of instances and managing the cluster is being administratively heavy process. With ECS, Fargate launch type, the load, and responsibility of managing the EC2 cluster is transferred over to the AWS and you can focus on application development rather than management of your cluster architecture.AWS Fargate is the AWS service that allows ECS to run containers without having to manage and provision the resources required for running these applications.

How Elastic Container Service Works?

Amazon elastic container service is a fully managed service which is provided by AWS it is mainly used to deploy containers that are docker based by which you scale up and down depending on the traffic you're going to get. The containers will run inside the Amazon elastic cloud (EC2) instance.


Developers
Describe your application & the resources it needs, then let Amazon ECS do the rest.
Operators
Create custom scaling & capacity rules. Observe & query telemetry & logs from running applications.
Amazon Elastic Container Service
Configuration
Deployments & Scaling
Application telemetry
Elastic Load Balancing
EWS Secrets Manager
Amazon Elastic File System
Automated integration with other AWS Services
AWS Fargate
Amazon EC2 (Gravitation; Intel)
Amazon ECS Anywhere
AWS Outposts
AWS Local Zones
AWS Wavelength
Flexible Compute Options
Amazon Elastic Container Registry
Build & store container images using Amazon ECR or another container Registry

Container: A container is a package that holds an application and everything dependencies, libraries, etc.) the application requires to run. Containers are independent of the underlying operating system and hence container applications are fairly portable, flexible, and scalable. This ensures the application will run always as expected irrespective of the system and environment in which a container is run.

Docker: Docker is software that facilitates and automates the installation and deployment of applications inside Linux containers.

Cluster: A logic group of EC2 instances running as a single application.

Container Instance: Each EC2 in an ECS Cluster is called a container instance.

Autoscaling With AWS ECS

Autoscaling is an AWS Compute service that allows your application to scale up/down according to EC2 instance CPU usage or some other criteria (Autoscaling policies) set by the user. For example: For a web application, you can set an Autoscaling Policy like, when CPU usage exceeds 80% for five minutes add another EC2 instance with the same configurations. This will add another instance behind the Amazon ELB and allow requests divided into EC2 instances now. Earlier, the Autoscaling feature was only provided with AWS EC2 service, in the year 2016, it started to support ECS clusters as well.

Autoscaling for ECS Clusters can be set up from AWS-CLI, AWS-Console and AWS SDKs as well.

You can choose the number of starting tasks for the application, and attach one or more policies to the Autoscaling Group and AWS handles the rest.

Autoscaling for ECS also manages the Multi-AZ presence of your ECS cluster. That is, you can attach policies to have a minimum of one or more instances in each Availability Zone to make your application highly available.

Features of ECS

Some of the features of ECS are listed below:

Removes the need for users to manage their CPU type clusterthe management system by interacting with AWS Fargate.

Allows seamless deployment of container-based applications. This can be scheduled or done by simple API calls.

AWS ECS takes care of the management and monitoring of the application cluster.

Amazon ECS is *region specific.* This means that a cluster can only scale up/down (start-up or shut-down container instances) in a single region.

Clusters are dynamically scalable.

Launch types of AWS ECS

ECS can be launched in two following modes:

Fargate launch: As discussed above, the *fargate launch type*, takes most of the responsibility from the users and takes in only the basic inputs like the CPU type, memory, and IAM policies from the user to run the application cluster.

EC2 Launch: This is a more customizable launch type. Users are responsible for the number of instances in the cluster, scaling their cluster, and more. This allows you to be more in control of your clusters which may be required for security reasons.

Irrespective of the launch type AWS Cloudwatch is enabled for ECS and can be used for monitoring or to create alarms and events as required.

Advantages Of ECS

Here are some advantages of ECS:

Scalability: ECS automatically scales your applications based on demand, allowing you to easily handle changes in traffic or workload.

High availability: ECS provides built-in availability and fault tolerance, ensuring that your applications are always up and running.

Cost-effective: ECS enables you to optimize your infrastructure costs by scaling resources based on demand and only paying for what you use.

Integration: ECS integrates with other AWS services such as Amazon ECR, AWS Fargate, Amazon CloudWatch, and AWS IAM.

Security: ECS provides a secure environment to run your applications, with features such as IAM roles for tasks, VPC isolation, and encryption at rest.

ECS has some limitations to consider. For example, it requires a certain level of expertise to set up and configure the service. Additionally, it may not be suitable for applications that require a high level of customization or specialized networking requirements.

Amazon ECS Capacity

The Amazonwhich ECS capacity can be managed in no.of different ways some of which are mentioned below.

Amazon Cluster: Amazon will allow clustersclusters you to have one are more clusters depending upon the capacity needed for you. Amazon cluster will isolate or group the resources like Amazon EC2 or forget which will allow runrun you to run your containers.

Fargate: Containers can be run with the help of AWS Fargatethe also which will help you focus more on your code rather than worrying about underlying infrastructure. It will help you to deploy the application in the form of containers without managing the EC2-instance.

Integrate: Amazon ECS will allow you to integrate the on-premises server or virtual machine with the Amazon ECS Cluster.

AutoScaling: Amazon ECS supports autoscaling which the containers will scale up and down depending on the incoming traffic. If the traffic is high then Amazon ECS will scale up the containers and if there is minimal traffic then Amazon ECS will scale down the containers.

Amazon ECS Provisioning

You can provision the AMzon ECS differently as some of them mentioned following.

AWS Management Console: It like graphical user interface(GUI) which can be accessed from the internet and can perform the tasks according to your requirements.

AWS Command Line Interface(AWS CLI): By using commands you can manage the AWS services.

AWS SDKs: Offers APIs tailored to individual languages and handles most connection-related issues.

AWS Command Line Interface(AWS CLI)

AWS provides a set of commands that can be run on AWS-CLI (AWS Command Line Interface) to manage your services. Much like you'd manage from your AWS Console. Following is a list of commands that can be used for managing the AWS ECS service.

create-capacity-provider: Used to create a new capacity provider. Capacity providers comprise a name, an Auto Scaling group along with termination and scaling settings.

create-cluster: Creates a new AWS ECS Cluster.

create-service: Runs and maintains a desired number of tasks as specified by the given task definition.

create-task-set: Create a the task-set in AWS ECS cluster or a Service.

delete-account-setting: Disables an account setting for an IAM user, role, the or the root user of the account.

delete-attributes: Deletes one or more custom attributeof thehe a ECS cluster.

delete-cluster: Deletes an entire specified cluster.

delete-service: Deletes the specified service.

delete-task-set: Deletes the specified task-set.

deploy: Deploys new task definition to the specified ECS service.

deregister-container-instance: Deregisters the specified container instance from its ECS cluster.

deregister-task-definition: Deregisters a task definition from ECS Service.

describe-capacity-providers: Describes one or more capacity providers.

describe-clusters: Describes one or more of ECS Clusters.

describe-container-instances: Describes ECS container instances. Gives metadata about remaining resources on each of these instances.

describe-services: Describes Services runningin aa resources specified cluster.

describe-task-definition: Describe task definition.

describe-task-sets: Describes task sets of the specified ECS service or cluster.

describe-tasks: Describes specified task(s).

discover-poll-endpoint: Gives endpoint of AWS ECS agent to poll for updates.

list-account-settings: Gives account settings for the specified principal.

list-attributes: List attributes for specified ECSresourcesa e.

list-clusters: Gives a list of existing clusters.

list-container-instances: Gives a list of container instances in a a specified cluster.

list-services: Gives a list of services runningina the resources specified cluster.

list-tags-for-resource: Gives tags associated with specifiedresourcestask-definitione.

list-task-definition-families: List all the task-definition, families registered to your account.

list-task-definitions: List task definitions registered to your account.

list-tasks: Gives tasks running in the specified cluster.

put-account-setting: attributesattributes,attributes resources or the root user.

put-account-setting-default: Used to modify an account setting for all IAM users on an account for whom no individual account setting has been specified.

put-attributes: Create/Update attributes resourcesresources for specified ECS resources.

put-cluster-capacity-providers: Modify capacity providers for a cluster.

register-container-instance: Registers container instances into the specified cluster.

register-task-definition: Registers task definition from the specified family and container definitions.

run-task: Starts a new task from a task definition.

start-task: Starts a new task from the task definition on the specified container instance or instances.

stop-task: Stop specified task. (Note that any tags associated with the task are deleted.)

submit-attachment-state-changes: Sent to acknowledge that a container changed states.

submit-container-state-change: Sent to acknowledge that an attachment changed states.

submit-task-state-change: Sent to acknowledge that a task changed states.

tag-resource: Adds specified tags to the resource whose RN is supplied.

untag-resource: Removes specified tags from a resource.

update-cluster-settings: Modify the cluster settings.

update-container-agent: Updates the Amazon ECS container agent on a specified container instance.

update-container-instances-state: Modifies the status of an Amazon ECS container instance.

update-service: Modifies the parameters of a service.

update-service-primary-task-set: Modifies which task-set in service is the primary task-set. Any parameters that are updated on the primary task set in a service will transition to the service.

update-task-set: Modifies a task set.

wait: Wait until a particular condition is satisfied. Each sub-command polls an API until the listed requirement is met.

FAQs On Amazon ECS

1.What Is The Difference Between EC2 And ECS?

Amazon EC2 is an service offered by the amazon web services used to deploy the containers and EC2 is an service which provide the resizable compute capacity in the form of virtual machines.

2. Is Amazon ECS The Same As Docker?

Amazon ECS uses Docker image to deploy the containers on the Amazon ECS.