**(Elastic Container Service) ECS – Part 2 – ECS**

**About ECS:**

* ECS is a regional service that you can use in one or more AZs across a new or existing, VPC to schedule the placement of containers across your cluster based on your resource needs, isolation policies, and availability requirements.
* Amazon ECS eliminates the need for you to operate your own cluster management and configuration management systems, or to worry about scaling your management infrastructure.
* ECS can also be used to create a consistent deployment and build experience, manage and scale batch and TEL workloads, and build experience, manage and scale batch and ETL workloads, and build sophisticated application architectures on a microservice model.

**What is a Docker Image?**

* An image is a read-only template with instructions for creating a Docker container. It contains:
  + An ordered collection of root filesystem changes and the corresponding execution parameters for use within a container runtime.
* An image is created from a DockerFile, a plain text file that specifies the components that are to be included in the container.
* Images are stored in a registry, such as DockerHub or AWS ECR.

**Container Registries:**

* Amazon EC2 Container Registry (Amazon ECR) is a managed AWS Docker registry that is secure, scalable, and reliable. Amazon ECR supports private Docker repositories with resource-based permissions using AWS IAM so that specific users or Amazon EC2 instances can access repositories and images. Developers can use the Docker CLI to push, pull, and manage images.

**ECS Task Definitions:**

* A Task Definition is required to run Docker containers in Amazon ECS.
* Task Definitions are text files in JSON format that describe one or more containers that form your application.
* Some of the parameters you can specify in a task definition include:
  + Which Docker images to use with the containers in your task.
  + How much CPU and memory to use with each container.
  + Whether containers are linked together in a task.
  + The Docker networking mode to use for the containers in your task.
  + What (if any) ports from the container are mapped to the host container instance.
  + Whether the task should continue to run if the container finishes or fails.
  + The command the container should run when it is started.
  + What (if any) environment variables should be passed to the container when it starts.
  + Any data volumes that should be used with the containers in the task.
  + What (if any) IAM role your tasks should use for permissions.

**ECS Services:**

* An Amazon ECS service allows you to run and maintain a specific number (or, the “desired count”) of instances of a task definition simultaneously in an ECS cluster.
* Think of services like auto-scaling groups for ECS.
* If a task should fail or stop, the Amazon ECS service scheduler launches another instance of your task definition to replace it and maintain the desired count of tasks in the service.

**ECS Clusters:**

* An Amazon ECS cluster is a logical grouping of container instances that you can place tasks on. When you first use the Amazon ECS service, a default cluster is created for you, but you can create multiple clusters in an account to keep your resources separate.
* Concepts:
  + Clusters can contain multiple different container instance types.
  + Clusters are region-specific.
  + Container instances can only be part of one cluster at a time.
  + You can create IAM policies for your clusters to allow or restrict users’ access to specific clusters.

**ECS Scheduling:**

* Service Scheduler:
  + Ensures that the specified number of tasks are constantly running and reschedules tasks when a task fails (for example, if the underlying container instance fails for some reason).
  + Can ensure tasks are registered against an ELB.
* Custom scheduler:
  + You can create your own schedulers that meet your business needs.
  + Leverage third-party schedulers, such as Blox.
* The Amazon ECS schedulers leverage the same cluster state information provided by the Amazon ECS API to make appropriate placement decisions.

**ECS Container Agent:**

* The Amazon ECS container agent allows container instances to connect to your cluster. The Amazon ECS container agent is included in the Amazon ECS-optimized AMI, but you can also install it on any EC2 instance that supports the Amazon ECS specification. The Amazon ECS container agent if only supported on EC2 instances.
  + Pre-installed on special ECS AMIs.
  + Linux-based:
    - Works with Amazon Linux, Ubuntu, Red Hat, CentOS, etc.
    - Will not work with windows.

**ECS Security:**

* IAM Roles:
  + EC2 instances use an IAM role to access ECS.
  + ECS tasks use an IAM role to access services and resources.
* Security Groups attach at the instance-level (i.e. the host not the task or container).
* You can access and configure the OS of the EC2 instances in your ECS cluster.

**ECS Limits:**

* Soft limits:
  + Clusters per region (default = 1000)
  + Instances per cluster (default = 1000)
  + Services per cluster (default = 500)
* Hard limits:
  + One load balancer per service.
  + 1000 tasks per service (the “desired count”)
  + Max. 10 containers per task definition.
  + Max. 10 tasks per instance.

**ECS Exam Tips:**

* ECS – Amazon’s managed EC2 container service. Allows you to manage Docker containers on a cluster of EC2 instance.
* Containers are a method of operating system virtualization that allow you to run an application and its dependencies in resource-isolated processes.
* Containers are created from a read-only template called an image.
* An image is a read-only template with instructions for creating a Docker container.
* Images are stored in a registry (Amazon ECR) is a managed AWS Docker registry service.
* A task definition is required to run Docker containers in Amazon ECS.
* Task definitions are text files in JSON format that describe one or more containers that form your application.
* Think of a task definition as a cloud formation template but for Docker. Configure things such as the amount of CPU, RAM, etc.
* An Amazon ECS service allows you to run and maintain a specified number (or the “desired count”) of instances of a task definition simultaneously in an ECS cluster.
* Think of services like auto-scaling groups for ECS.
* An Amazon ECS cluster is a logical grouping of container instances that you can place tasks on.
* Clusters can contain multiple different container instance types.
* Clusters are region-specific.
* Container instances can only be part of one cluster at a time.
* You can create IAM policies for your clusters to allow or restrict users’ access to specific clusters.
* You can schedule ECS in two ways:
  + Service Scheduler.
  + Customer Scheduler.
* ECS agent to connect EC2 instances to your ECS cluster LINUX ONLY.
* IAM with ECS to restrict access.
* Security groups operate at the instance level, not at the task or container level.