**Elastic Container Registry** Description

Amazon **Elastic Container Registry** (Amazon **ECR**) is an AWS managed container image registry service that is secure, scalable, and reliable. Amazon ECR supports private container image repositories with resource-based permissions using AWS IAM. This is so that specified users or Amazon EC2 instances can access your container repositories and images. You can use your preferred CLI to push, pull, and manage Docker images, Open Container Initiative (OCI) images, and OCI compatible artifacts.

**Amazon Elastic Container Registry Public** is a managed AWS container image registry service that is secure, scalable, and reliable. Amazon ECR supports public image repositories with resource-based permissions using AWS IAM so that specific users can access your public repositories to push images. Developers can use their preferred CLI to push and manage Docker images, Open Container Initiative (OCI) images, and OCI compatible artifacts. Your images are publicly available to pull, either anonymously or using an Amazon ECR Public authentication token.

## Service Overview

Amazon ECR contains the following components:

* An **Amazon ECR registry** is provided to each AWS account; you can create image repositories in your registry and store images in them. For more information, see [Amazon ECR private registries](https://docs.aws.amazon.com/AmazonECR/latest/userguide/Registries.html).
* Your client must authenticate to Amazon ECR registries as an AWS user with **Authorization token** before it can push and pull images. For more information, see [Private registry authentication](https://docs.aws.amazon.com/AmazonECR/latest/userguide/registry_auth.html).
* An **Amazon ECR image repository** contains your Docker images, Open Container Initiative (OCI) images, and OCI compatible artifacts. For more information, see [Amazon ECR private repositories](https://docs.aws.amazon.com/AmazonECR/latest/userguide/Repositories.html).
* You can control access to your repositories and the images within them with **repository policies**. For more information, see [Repository policies](https://docs.aws.amazon.com/AmazonECR/latest/userguide/repository-policies.html).
* You can push and pull container **images** to your repositories. You can use these images locally on your development system, or you can use them in Amazon ECS task definitions and Amazon EKS pod specifications. For more information, see [Using Amazon ECR images with Amazon ECS](https://docs.aws.amazon.com/AmazonECR/latest/userguide/ECR_on_ECS.html) and [Using Amazon ECR Images with Amazon EKS](https://docs.aws.amazon.com/AmazonECR/latest/userguide/ECR_on_EKS.html).

Amazon ECR Public contains the following components:

* The **Amazon ECR Public Gallery** is the public portal that lists all public repositories hosted on Amazon ECR Public. Visit the Amazon ECR Public Gallery at [https://gallery.ecr.aws](https://gallery.ecr.aws/). For more information, see [Using the Amazon ECR Public Gallery](https://docs.aws.amazon.com/AmazonECR/latest/public/public-gallery.html).
* A **public registry** is provided to each AWS account; you can create public image repositories in your public registry and store images in them. For more information, see [Amazon ECR public registries](https://docs.aws.amazon.com/AmazonECR/latest/public/public-registries.html).
* Your client must authenticate to a public registry as an AWS user before it can push images to a public repository. For image pulls, Amazon ECR Public accepts both anonymous pulls and pulls using an **authentication token**. For more information, see [Registry authentication](https://docs.aws.amazon.com/AmazonECR/latest/public/public-registries.html#public-registry-auth).
* Repository
* An Amazon ECR image repository contains your Docker images, Open Container Initiative (OCI) images, and OCI compatible artifacts. For more information, see [Amazon ECR public repositories](https://docs.aws.amazon.com/AmazonECR/latest/public/public-repositories.html).
* You can control access to your repositories and the images within them with **repository policies**. For more information, see [Public repository policies](https://docs.aws.amazon.com/AmazonECR/latest/public/public-repository-policies.html).
* You can push and pull container **images** to your repositories. You can use these images locally on your development system, or you can use them in Amazon ECS task definitions and Amazon EKS pod specifications.

## Use cases / Considerations

**Amazon ECR** provides the following features:

* **Lifecycle policies** help with managing the lifecycle of the images in your repositories. You define rules that result in the cleaning up of unused images. You can test rules before applying them to your repository. For more information, see [Lifecycle policies](https://docs.aws.amazon.com/AmazonECR/latest/userguide/LifecyclePolicies.html).
* Image scanning helps in identifying software vulnerabilities in your container images. Each repository can be configured to scan on push. This ensures that each new image pushed to the repository is scanned. You can then retrieve the results of the image scan. For more information, see [Image scanning](https://docs.aws.amazon.com/AmazonECR/latest/userguide/image-scanning.html).
* Cross-Region and cross-account replication makes it easier for you to have your images where you need them. This is configured as a registry setting and is on a per-Region basis. For more information, see [Private registry settings](https://docs.aws.amazon.com/AmazonECR/latest/userguide/registry-settings.html).

**Amazon ECR** is often used as source point for [ECS](https://docs.aws.amazon.com/AmazonECS/latest/developerguide/ecr-repositories.html) and [EKS](https://docs.aws.amazon.com/AmazonECR/latest/userguide/ECR_on_EKS.html) workflows.

It can perfectly suit as a backup solution for your on-premise docker registry.

## Governance

**Cross region replication** aimed to meet backup and disaster recovery requirements for your application. For more information, see [Private image replication.](https://docs.aws.amazon.com/AmazonECR/latest/userguide/replication.html)

For available **metrics** follow [Amazon ECR usage metrics](https://docs.aws.amazon.com/AmazonECR/latest/userguide/monitoring-usage.html) page.

## Cautions

Relevant quotas for **ECR** could be found on the [appropriate Amazon ECR service quotas page](https://docs.aws.amazon.com/AmazonECR/latest/userguide/service-quotas.html).  
**ECR Public** has separate [Amazon ECR Public service quotas page.](https://docs.aws.amazon.com/AmazonECR/latest/public/public-service-quotas.html)

## Pricing considerations

[*Amazon Elastic Container Registry pricing*](https://aws.amazon.com/ecr/pricing/) *(It has section regarding Public repositories as well)*

## More details

[*Containers Roadmap*](https://github.com/aws/containers-roadmap) [*Amazon Elastic Container Registry Public: A New Public Container Registry*](https://aws.amazon.com/blogs/aws/amazon-ecr-public-a-new-public-container-registry/)[*AWS on Air 2020: AWS What’s Next ft. Amazon Elastic Container Registry Public*](https://youtu.be/JMO50AoHHvY)

[*AWS re:Invent 2016: Amazon ECR Deep Dive on Image Optimization (CON401)*](https://youtu.be/Gk_7TR1QAss)

[*Managing Container Images with Amazon ECR - AWS Online Tech Talks*](https://youtu.be/JAlD2fNOPB4)