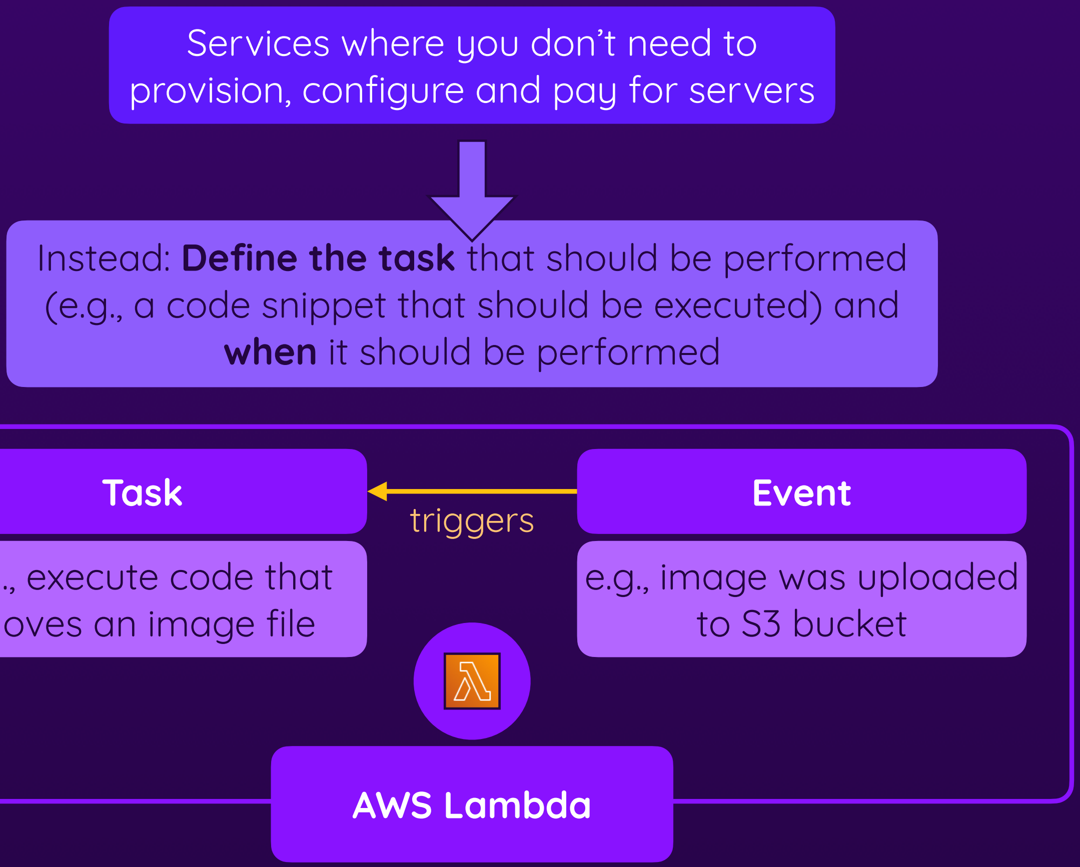
Beyond EC2: Serverless & Containers

From instances to Lambda functions & ECS clusters

- A Closer Look At "Serverless Services" & Containers
- Understanding AWS Lambda & ECS / EKS
- When To Use It

What Are "Serverless Services"?



Services where you don't need to provision, configure and pay for servers

Instead: **Define the task** that should be performed (e.g., a code snippet that should be executed) and **when** it should be performed

Task

e.g., execute code that moves an image file

triggers

Event

e.g., image was uploaded to S3 bucket

AWS Lambda

There Are Other Serverless Services!

AWS Lambda is the main serverless compute service

But compute isn’t everything!

e.g., you can think of S3 as a serverless storage service

A Closer Look At AWS Lambda

Code

- You upload or define the code that should be executed
- Write code in console, upload ZIP file or Docker container
- Choose a supported programming language
- Advanced setup (e.g., environment variables)

Event

- Choose a supported event source
- e.g., a file was uploaded to S3
- Advanced setup (e.g., event filtering)

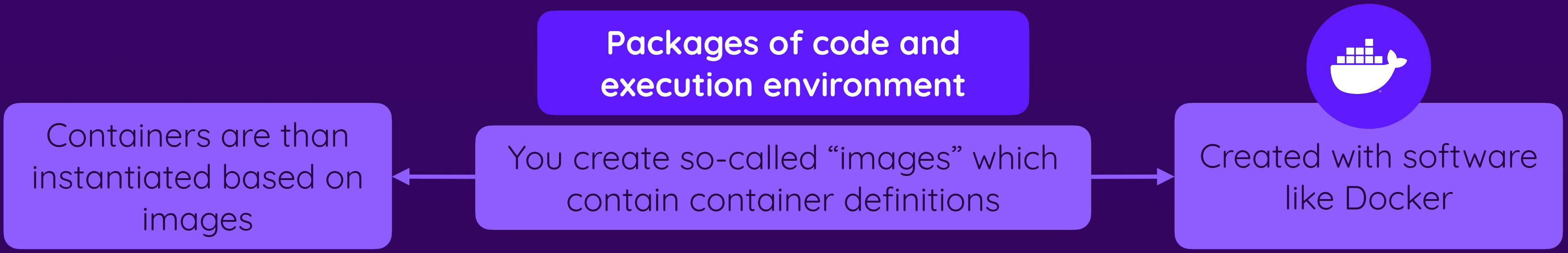
Configuration

- Timeout, underlying architecture, file systems, …
- Attach an execution role for permissions
- Connect to a VPC (it’s NOT placed in there though)

EC2 vs Lambda

EC2	Lambda
Spin up instances, install software & run your code	Upload your code & define execution events
You can install & run any software	You can only executed code (can’t install software)
e.g., run web server, run databases, …	e.g., no easy way of running web servers, no databases, …
Extremely versatile & configurable	Focused on event-driven code execution
Does requires lots of manual setup work & pay for uptime	Almost no manual setup work required & only pay for usage

What Are Containers?



Packages of code and execution environment

Containers are than instantiated based on images

You create so-called “images” which contain container definitions

Created with software like Docker

Single Image Application

One container contains all the parts that make up the application

e.g., web server & database in one single image

Multiple containers may be started (based on same image) for scaling

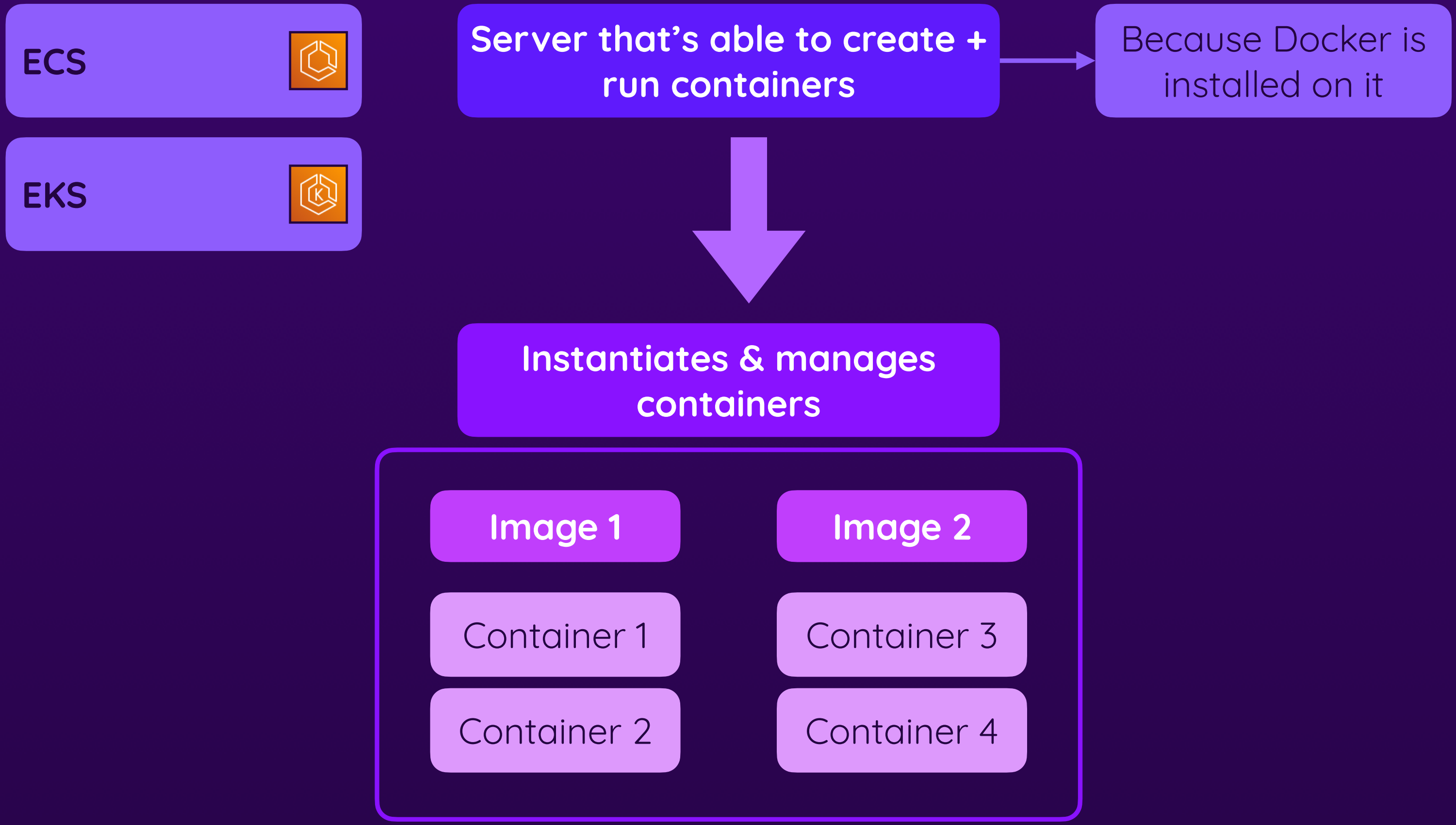
Multi Image Application

Multiple containers contain the parts that make up the application

e.g., web server & database in two separate images

Multiple containers for multiple images (+ potential scaling containers)

Running Containers



ECS

EKS

Server that’s able to create + run containers

Because Docker is installed on it

Instantiates & manages containers

Image 1

Container 1

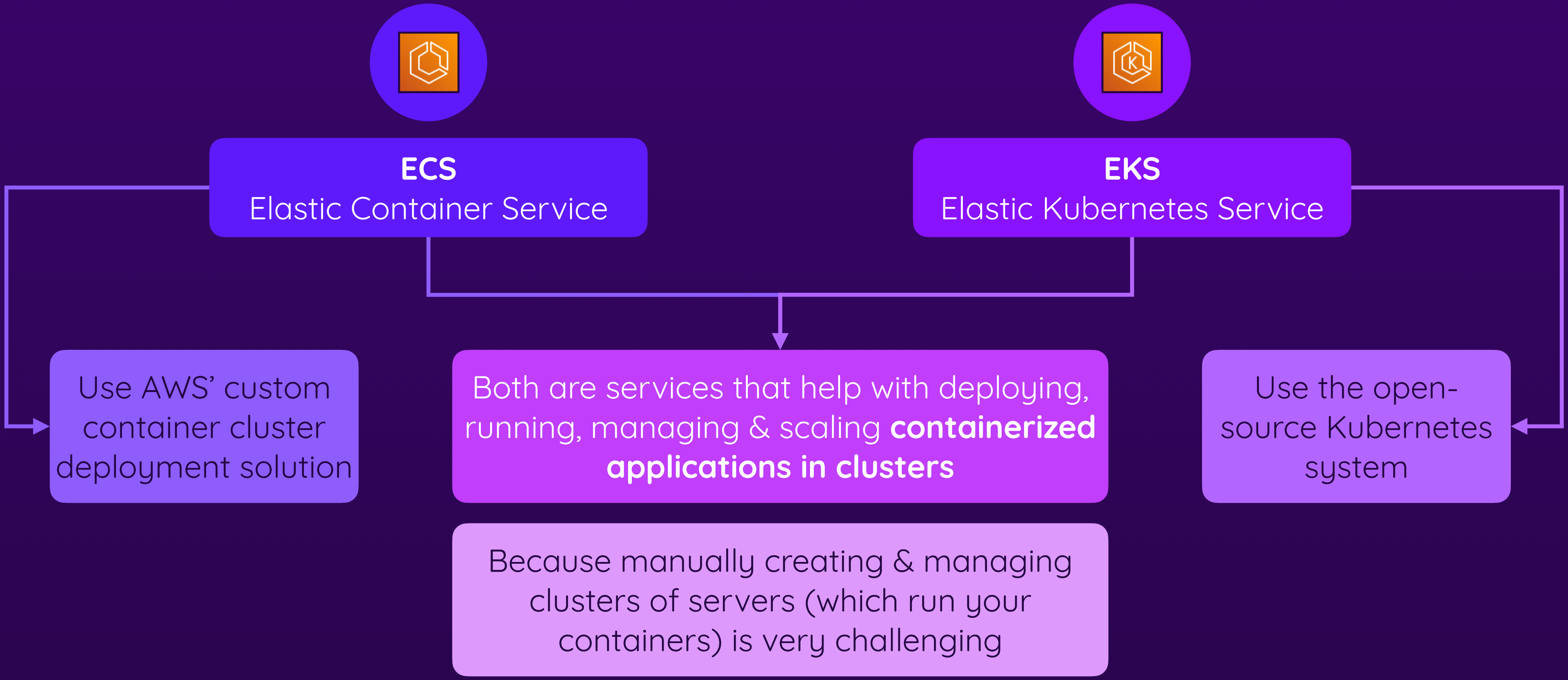
Container 2

Image 2

Container 3

Container 4

ECS & EKS



ECS
Elastic Container Service

EKS
Elastic Kubernetes Service

Use AWS' custom container cluster deployment solution

Both are services that help with deploying, running, managing & scaling **containerized applications in clusters**

Use the open-source Kubernetes system

Because manually creating & managing clusters of servers (which run your containers) is very challenging

Understanding ECS & EKS

Managed Container Clusters

Services that help with launching, scaling & managing containers

Define Cluster Structure

- Define tasks: images & image configurations
- Choose EC2 instances or **Fargate** as container host
- Configure default network & security settings

Operate & Scale Containers

- Define service- / task-specific settings
- Monitor containers
- Start or stop when needed

The Need For Container Image Repositories

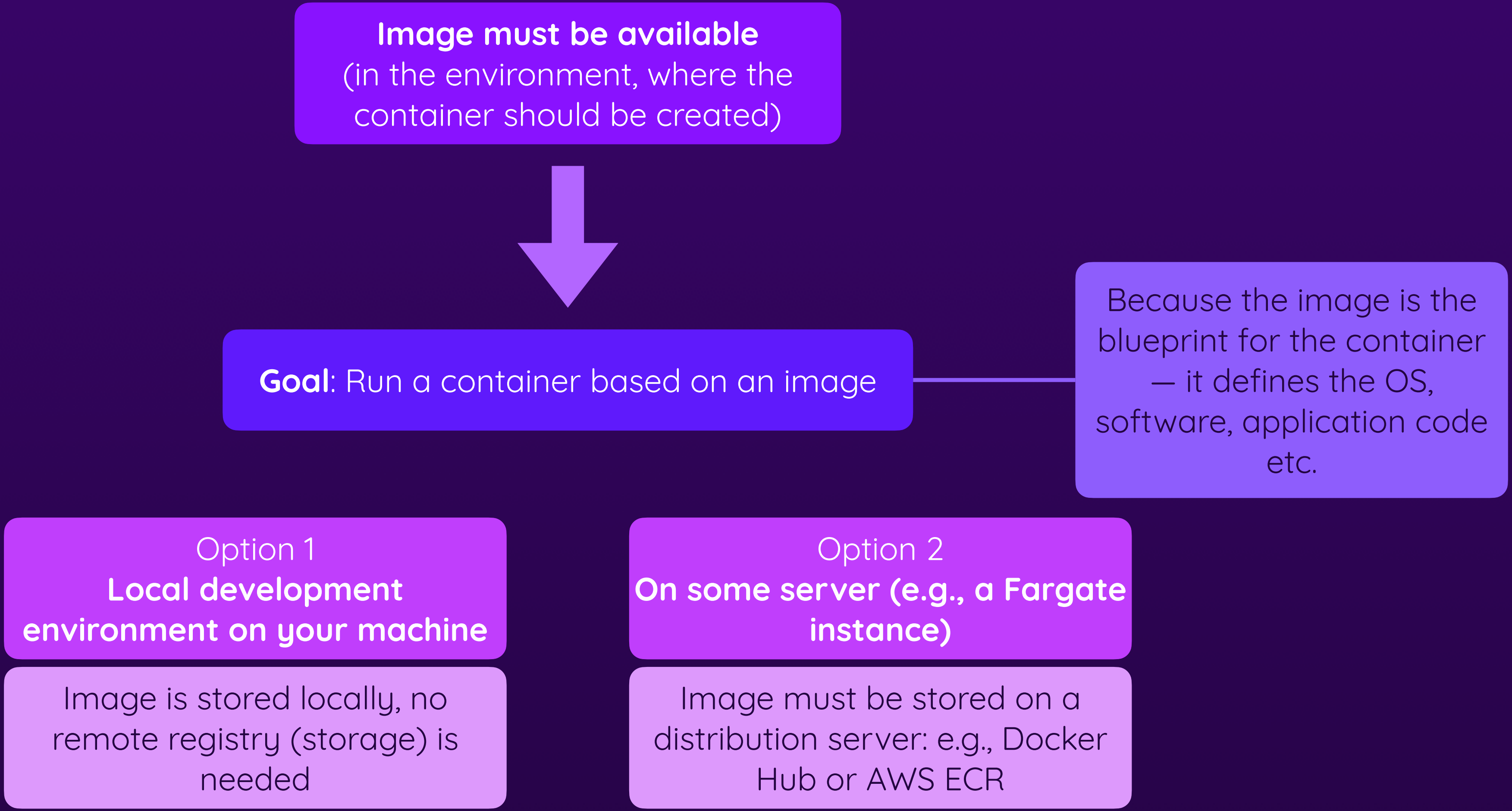


Image must be available
(in the environment, where the container should be created)

Goal: Run a container based on an image

Because the image is the blueprint for the container — it defines the OS, software, application code etc.

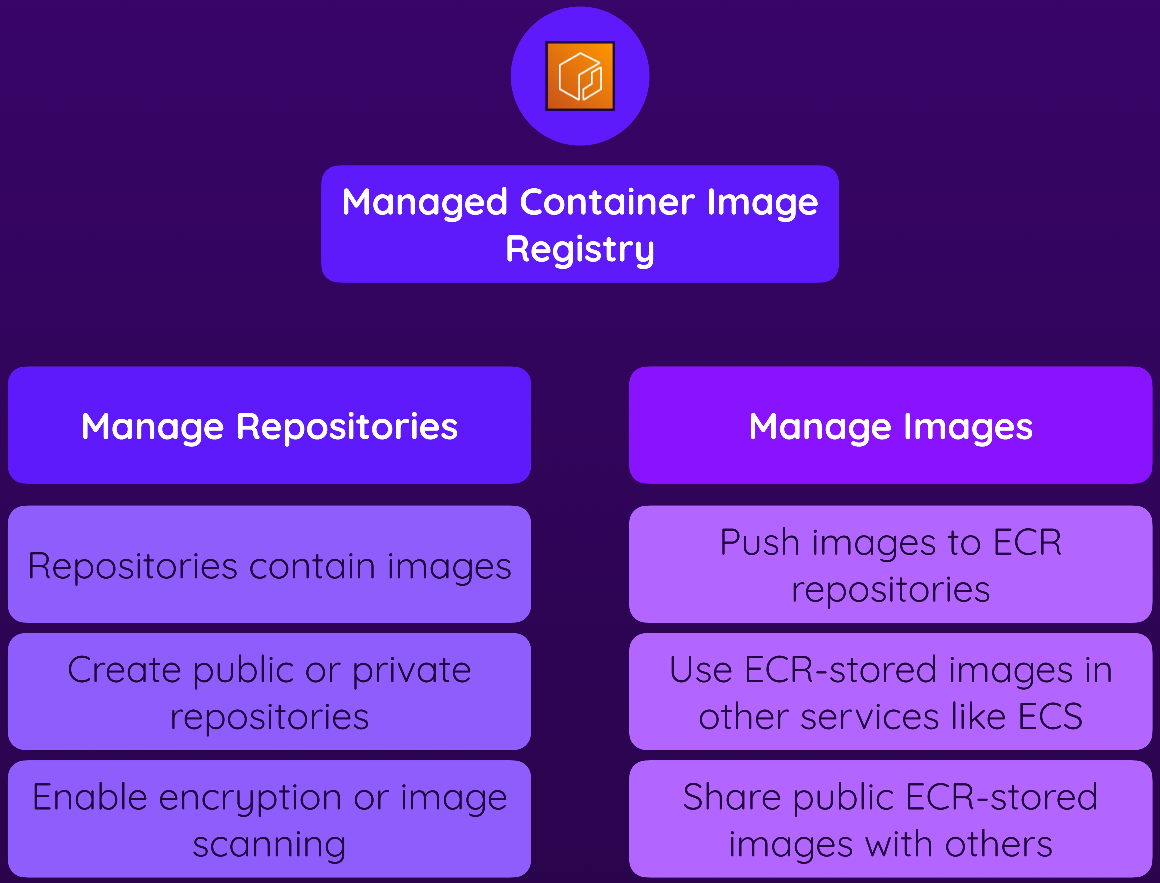
Option 1
Local development environment on your machine

Image is stored locally, no remote registry (storage) is needed

Option 2
On some server (e.g., a Fargate instance)

Image must be stored on a distribution server: e.g., Docker Hub or AWS ECR

Managing Images with ECR



Managed Container Image Registry

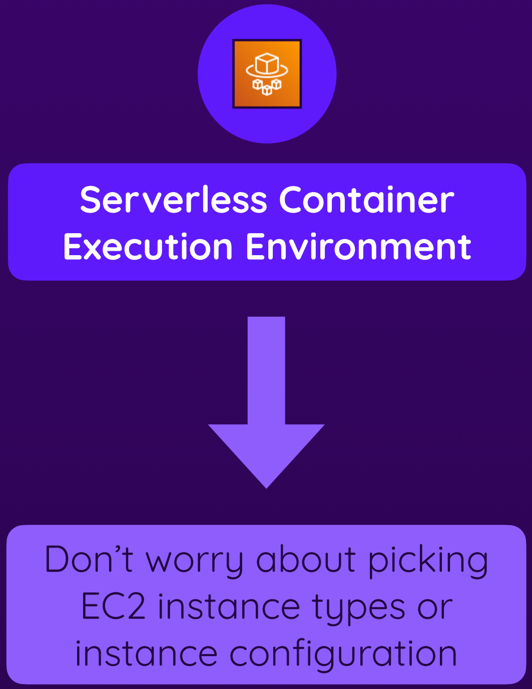
Manage Repositories

- Repositories contain images
- Create public or private repositories
- Enable encryption or image scanning

Manage Images

- Push images to ECR repositories
- Use ECR-stored images in other services like ECS
- Share public ECR-stored images with others

Understanding Fargate



Serverless Container Execution Environment

Don’t worry about picking EC2 instance types or instance configuration

Summary

Serverless & Containers

- Alternative to EC2 (where you rent entire servers)
- **Serverless:** On-demand code execution (with a timeout)
- **Containers:** Packages of code + required execution environment
- Different problems benefit from different solutions

AWS Lambda

- Serverless, event-driven code execution
- Provide code + define event triggers + execution configuration
- Many supported event types (e.g., S3 file changes, …)
- Assign execution role for permissions

ECS, EKS, ECR

- Managed container clusters, help with running containers
- Provide images & environment configuration
- Run on top of EC2 instances or Fargate (serverless)
- Manage & distribute images with ECR