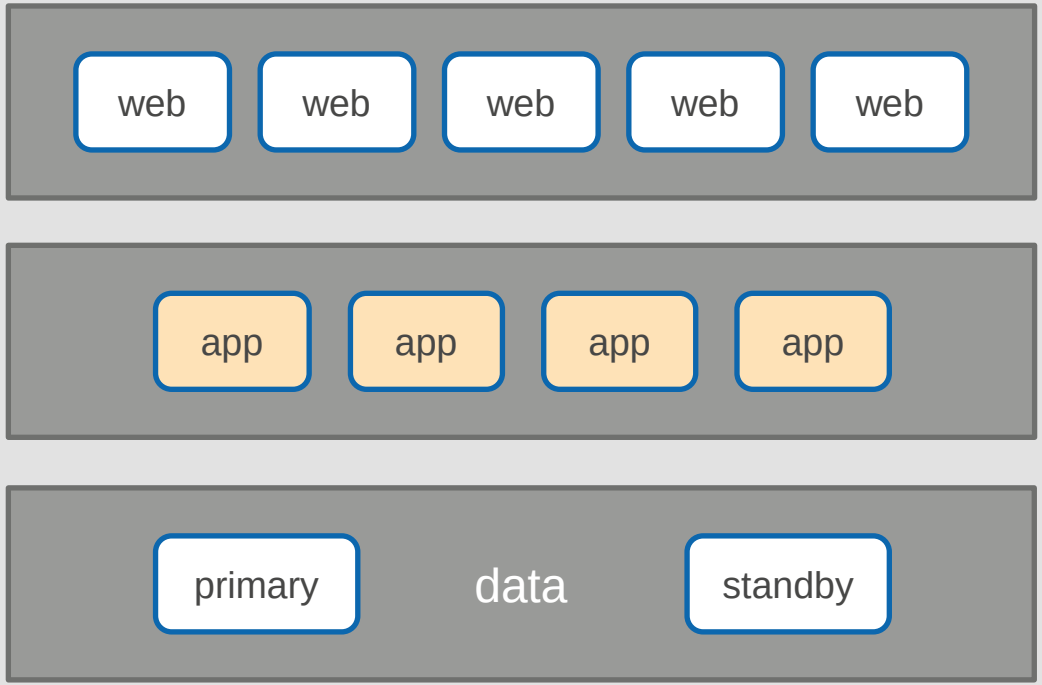
Elastic Container Service (ECS)

Chandra Lingam

Compute With Cloud Inc

An application requires many containers



Why ECS

- Replace unhealthy containers
- Scale number of containers
- Ensure sufficient resources are provisioned
- Release new version without downtime
- Simplify security

Lab – Run Containers on an EC2 Instance

Install docker stack in an EC2 instance

Run container

Docker installs required components

Query the web site

Failure simulation

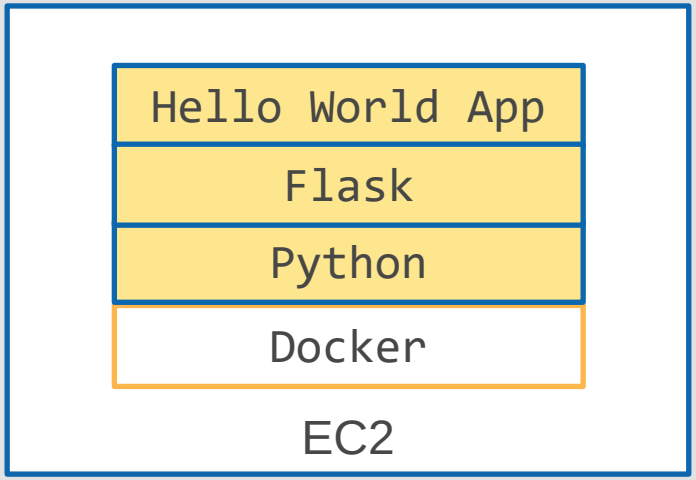


Image Location: https://hub.docker.com/repository/docker/demolearn/hello_world

Lab – ECS Cluster with EC2 Launch Type

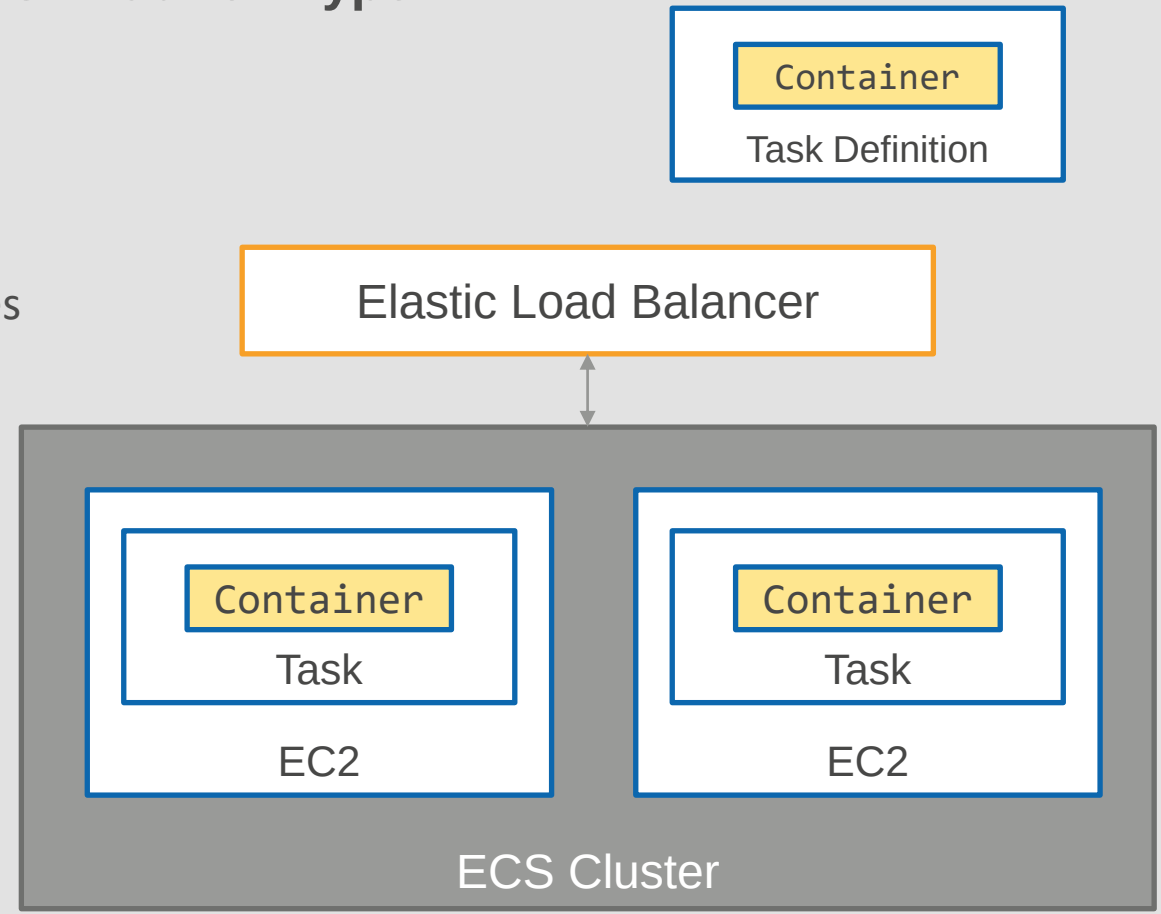
Create Task Definition

Create Cluster with EC2 instances

Run stand-alone Tasks

Run Tasks as a Service

Failure simulation



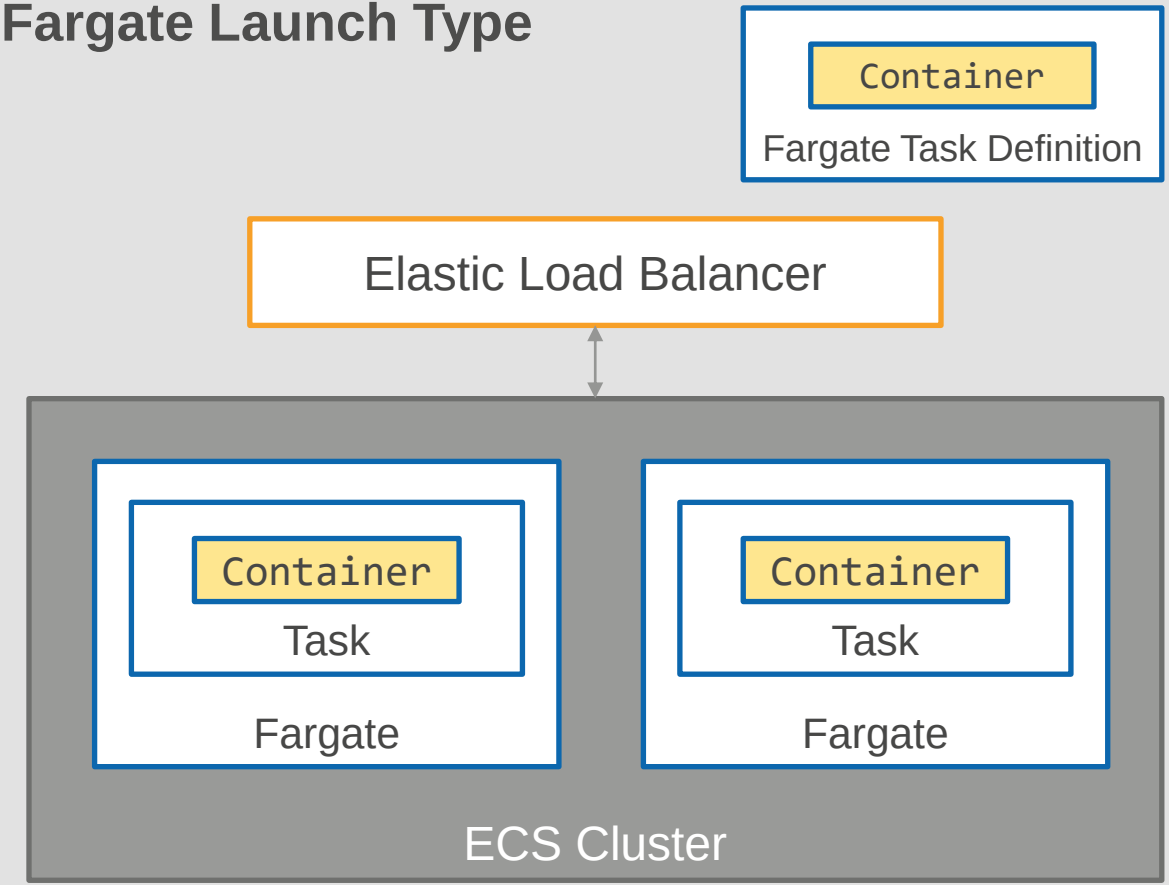
Lab – ECS Cluster with Fargate Launch Type

Create Task Definition

Create Cluster with Fargate

Run stand-alone Tasks

Run Tasks as a Service



ECS Best Practices

https://docs.aws.amazon.com/AmazonECS/latest/bestpracticesguide/intro.html

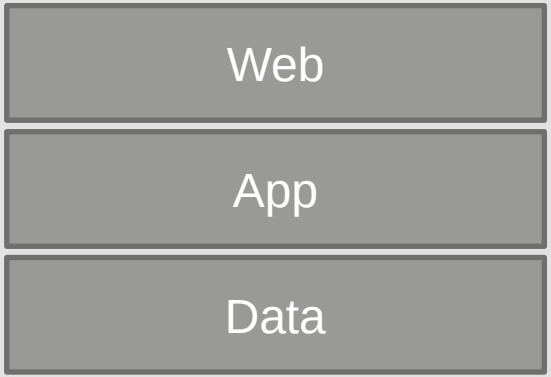
Task Definition

Application Layout

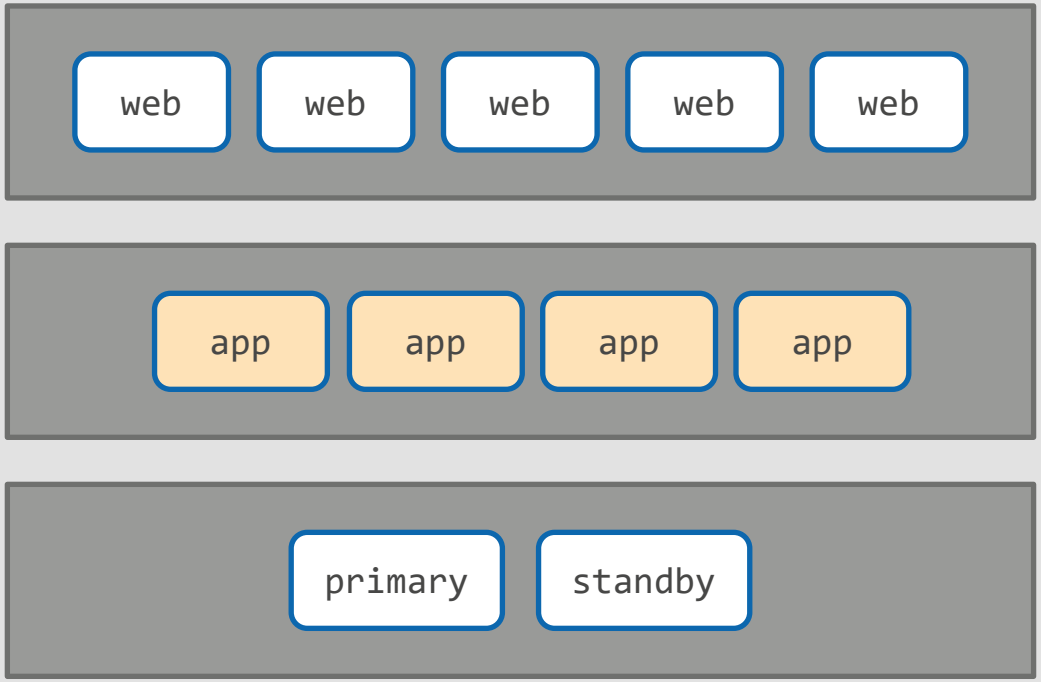
Container Image(s)

Resource requirements

- Memory
- CPU



Example Application



Web Container
Horizontal Scaling

App Container
Horizontal Scaling

Data Container
Vertical Scaling

One Task Definition with all images

Containers are started together and co-located

Mixing functionality

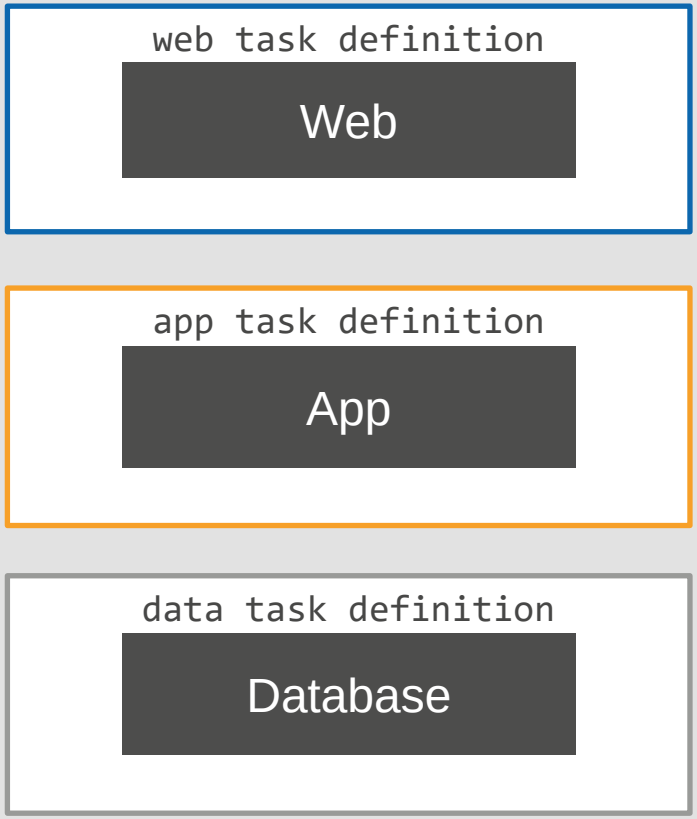
Cannot scale independently

Not a good practice

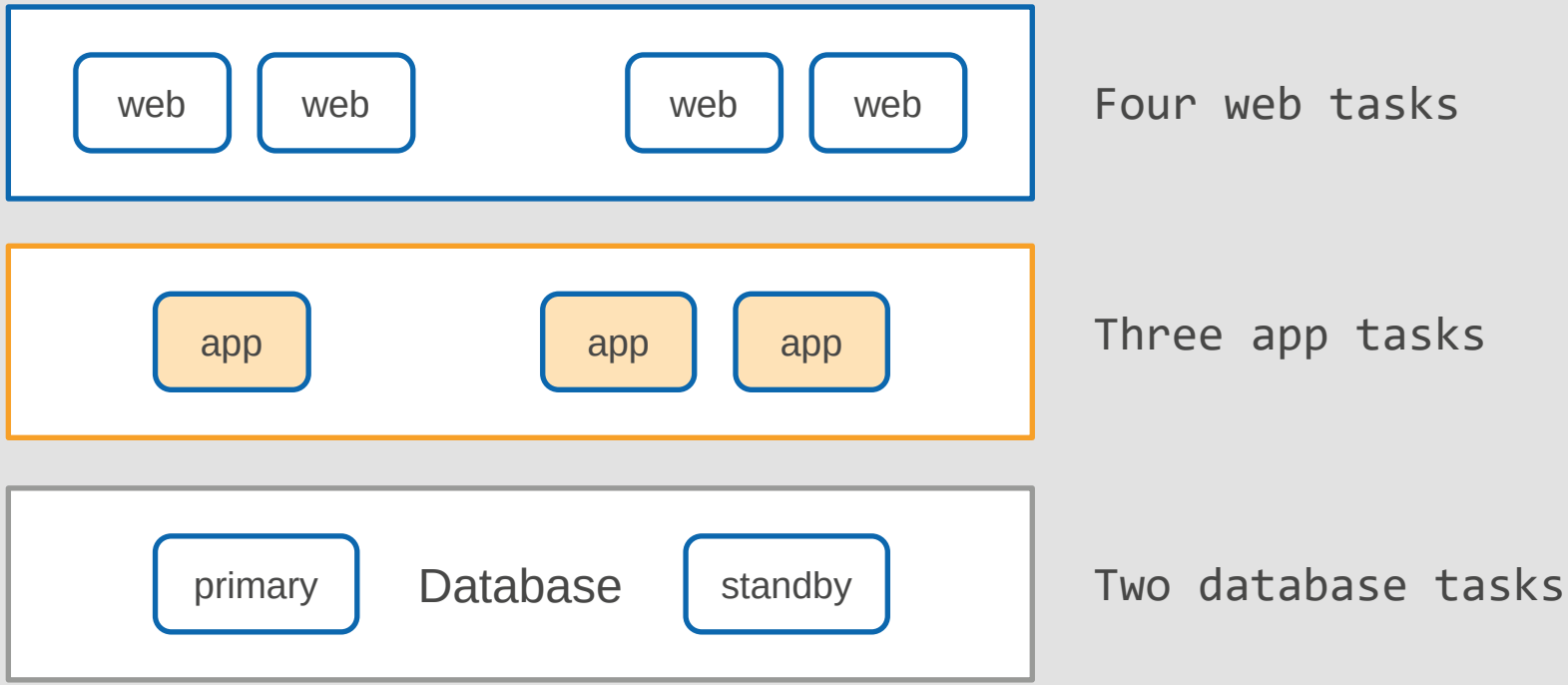
One Container per Task Definition

One container image per task definition

Additional support containers like side-cars may be included



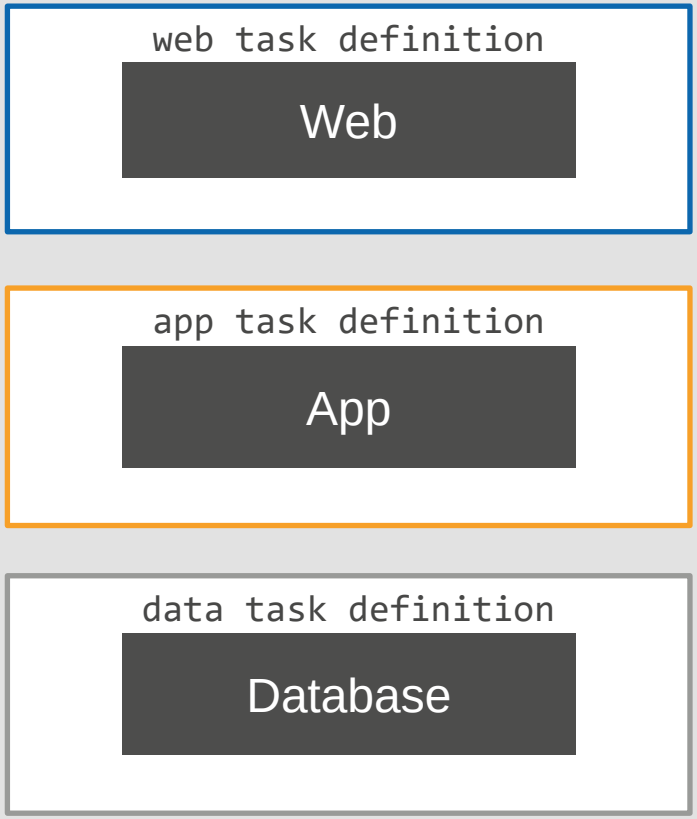
Each layer scaled independently



Summary - Task Definition

One container image per task definition

Additional support containers like side-cars may be included

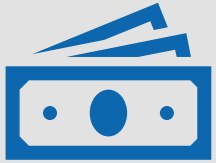


ECS Cluster

EC2 (server-based)

Fargate (serverless)

EC2 Launch Type



Cost effective

Workloads that require consistently high CPU and memory

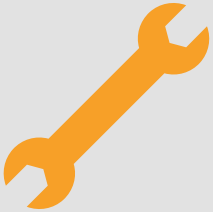
Security

Install additional security tools, host intrusion detection systems

Choice

Instance types (including GPU)

Fargate Launch Type



Maintenance

No need to manage infrastructure

Flexible

Suitable for many types of workloads

Batch

Perfect for batch workloads (compute needed only during specific times)

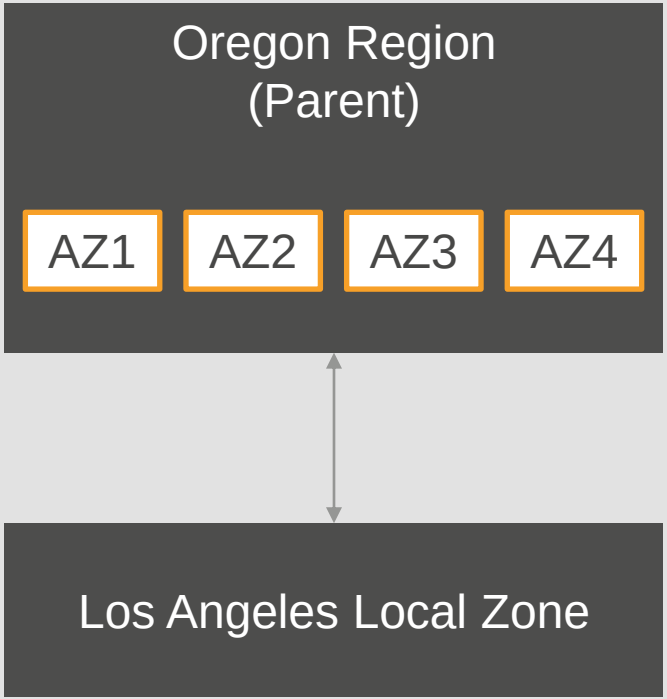
Additional Deployment Options

Location	Purpose
AWS Local Zones	• AWS location closer to large population centers and industries • Run latency sensitive applications in local zones
AWS Wavelength	• AWS managed edge infrastructure embedded within 5G communication providers • Run latency sensitive mobile services
AWS Outposts	• AWS managed on-premises infrastructure • Suitable when on-premises systems need low-latency access to AWS services
ECS Anywhere	• Run containers in customer-managed infrastructure • On-premises or other cloud providers

AWS Local Zones

AWS location closer to large population centers and industries

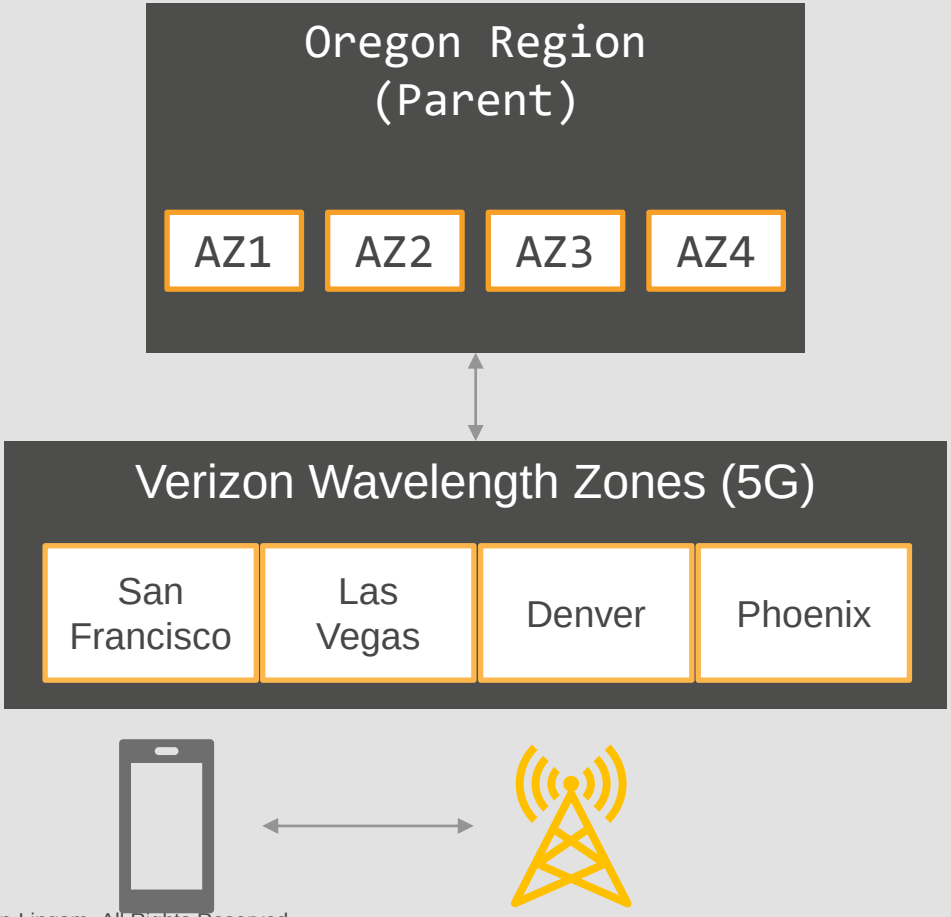
Run latency sensitive applications in local zones



AWS Wavelength

AWS managed edge infrastructure embedded within 5G communication providers

Run latency sensitive mobile services



AWS Outposts

Run AWS infrastructure and services on premises for a truly consistent hybrid experience

Get started with AWS Outposts | Contact Sales

FEATURED

Amazon Elasticsearch Service is now Amazon OpenSearch Service

Run and scale OpenSearch and Elasticsearch (versions 1.5 to 7.10) clusters.

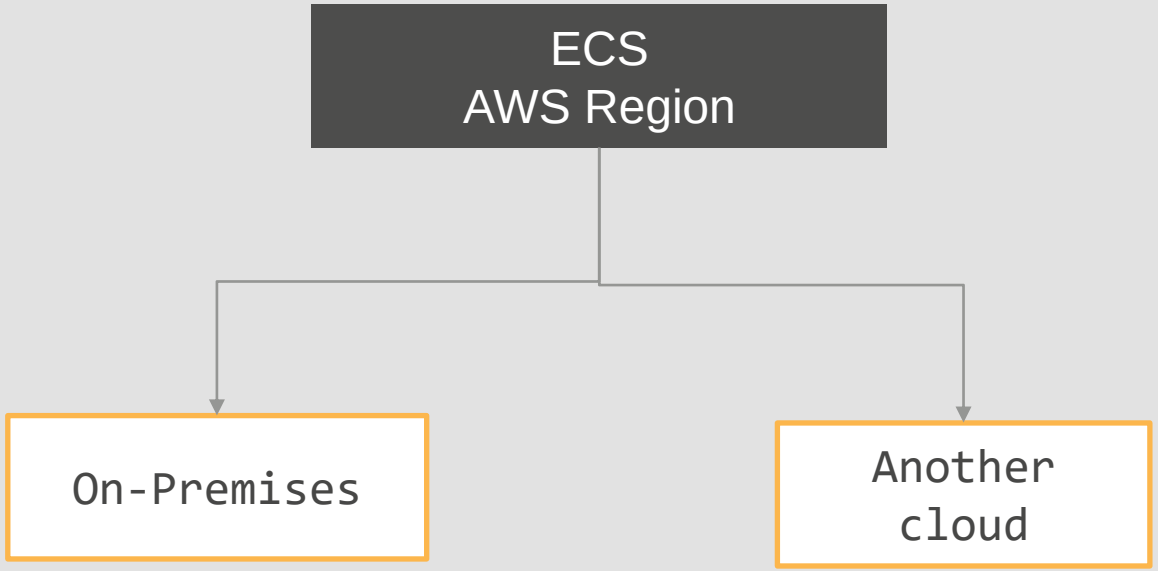
Learn more »

AWS Outposts is a fully managed service that offers the same AWS infrastructure, AWS services, APIs, and tools to virtually any datacenter, co-location space, or on-premises facility for a truly consistent hybrid

ECS Anywhere

Run containers in customer-managed infrastructure

On-premises or other cloud providers



Task – Three ways to run

Mode	Purpose
Batch	• Scheduled execution of tasks • Invoked through EventBridge rules
Service	• Suitable for continuously running tasks • Example, global ecommerce site need web, app and data tiers running 24x7
Run-task	• Manually start a task or configure run-task action in response to an event

Purchase Options

Container Workloads Purchase Options

On-demand

Compute Savings Plans (discounts for EC2 and Fargate)

Reservation and EC2 Savings Plans (discount for EC2)

Spot (discounts for EC2 and Fargate)

Container Registry

Elastic Container Registry

AWS provided service

IAM policy-based access control

Privately share images with accounts and applications

Faster task startup time – store images in the same region as your application

Docker Hub and Third-party registries

Accessed over the internet

Task launch may incur additional delay and costs

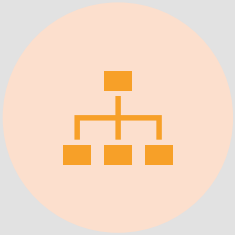
Public repository directly accessible

Private repository

Need to provide credentials
Store credentials in AWS Secrets Manager

Permission Management

Permissions Management



CLUSTER MANAGEMENT

CONTAINER IMAGE ACCESS

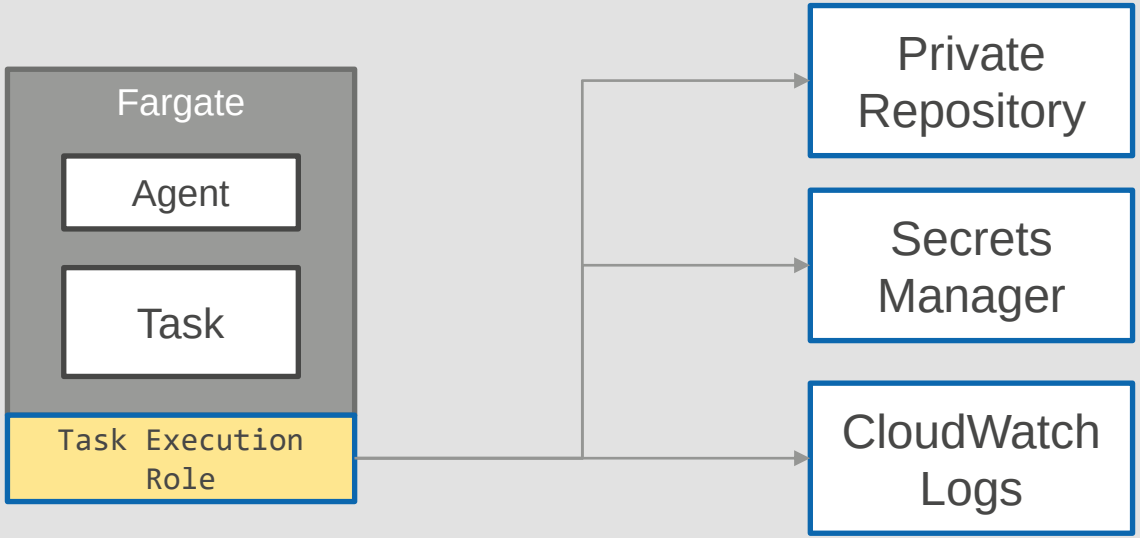
LOGGING

APPLICATION INTERACTION WITH OTHER AWS SERVICES

Task Execution Role

Fargate Agent uses this role to

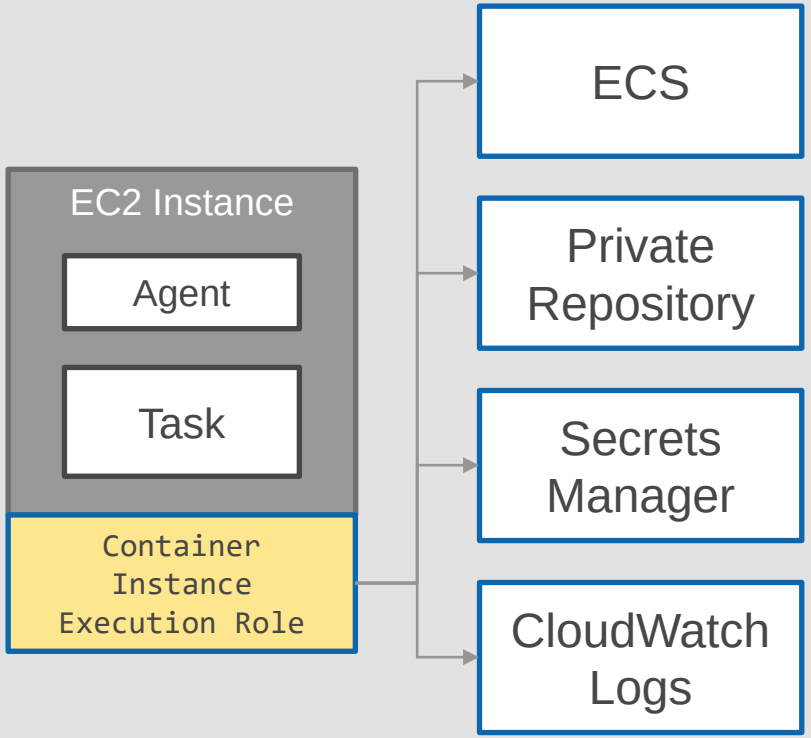
- Download images from private repository
- Retrieve third-party repository credentials stored in Secrets Manager
- Publish Container generated Logs to CloudWatch Logs



Container Instance Role

Agent (EC2) uses this role to

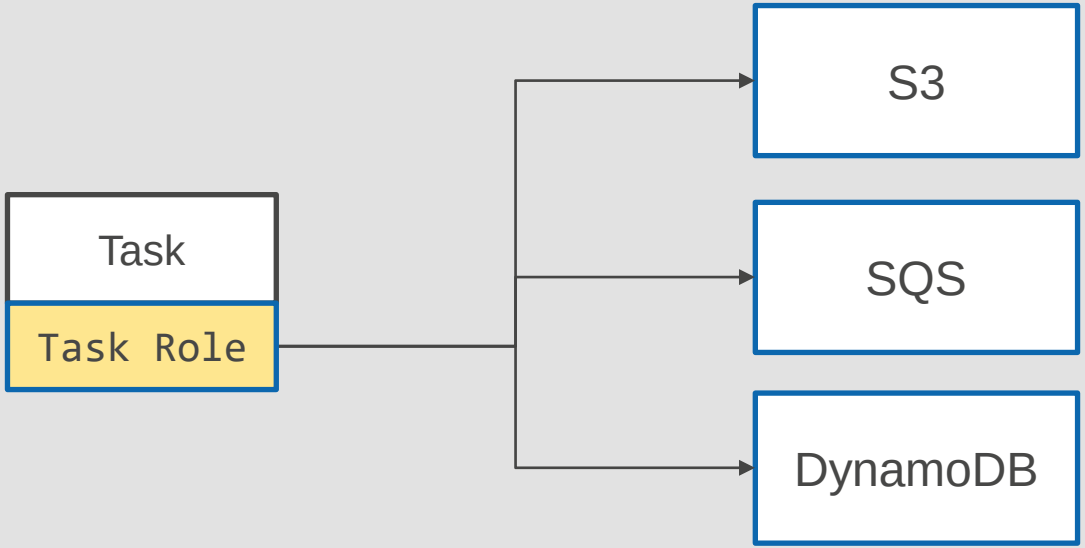
- Communicate with ECS service
  - EC2 Instance registration, deregistration
  - Launch and terminate tasks
  - Reports status of tasks and server to Cluster
- Download images from private repository
- Retrieve third-party repository credentials stored in Secrets Manager
- Publish Container generated logs to CloudWatch Logs



Task Role

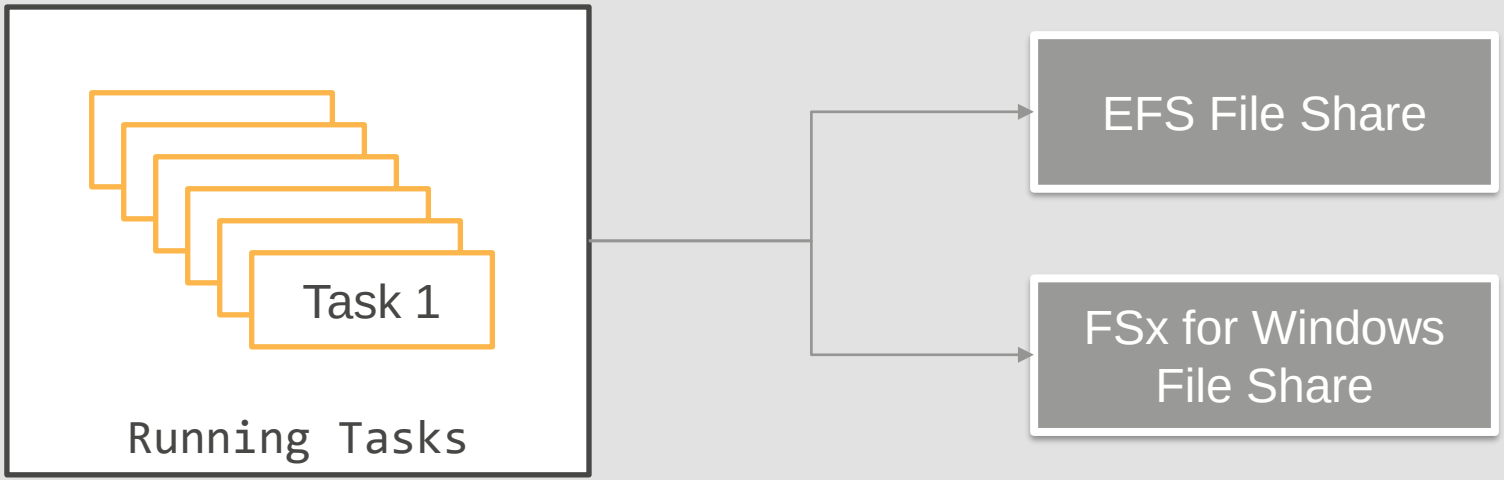
Application uses this role to

- Talk to other AWS Services such as S3, SQS, SNS, DynamoDB, etc.



Data Volumes for Containers

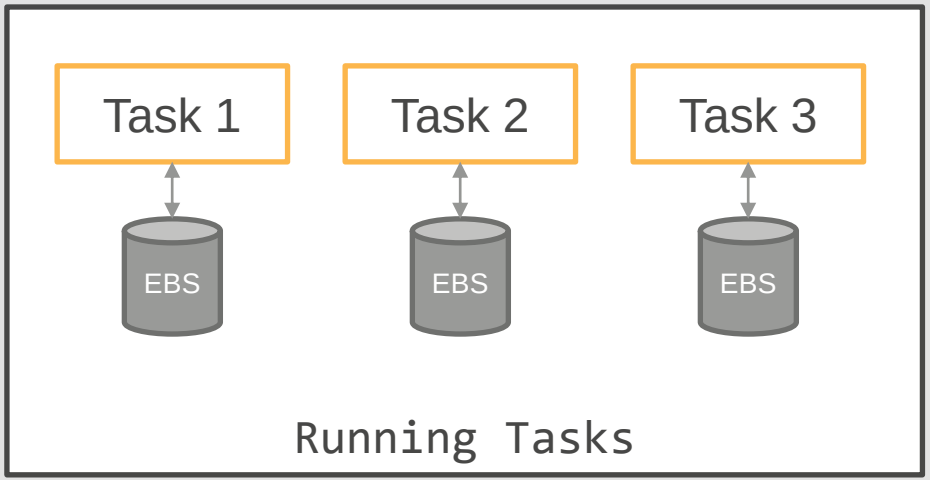
File Shares



All tasks (from a task definition) access the same storage

Example: ML distributed training, maintain data in a shared location

EBS Volumes



Private task level storage

Task requires persistent storage
Example: Databases

Task requiring temporary storage
Example: Video transcoding

Container Data Volumes

Storage	Purpose
EFS FSx for Windows	• File share • All tasks access the same storage • Example: ML distributed training, maintain data in a shared location
EBS Volume	• Private task level storage • Task requires persistent storage (Example: Databases) • Task requiring temporary storage (Example: Video transcoding)

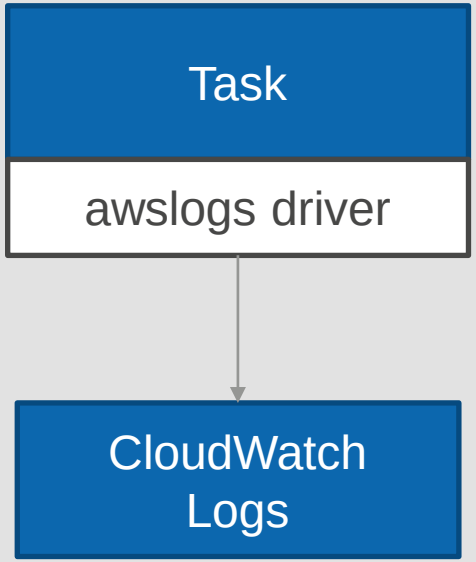
Monitoring and Logging

ECS automatically replaces unhealthy tasks and instances

CloudWatch - Aggregate metrics for running tasks by

- Cluster
- Service
- Task-definition

Logging

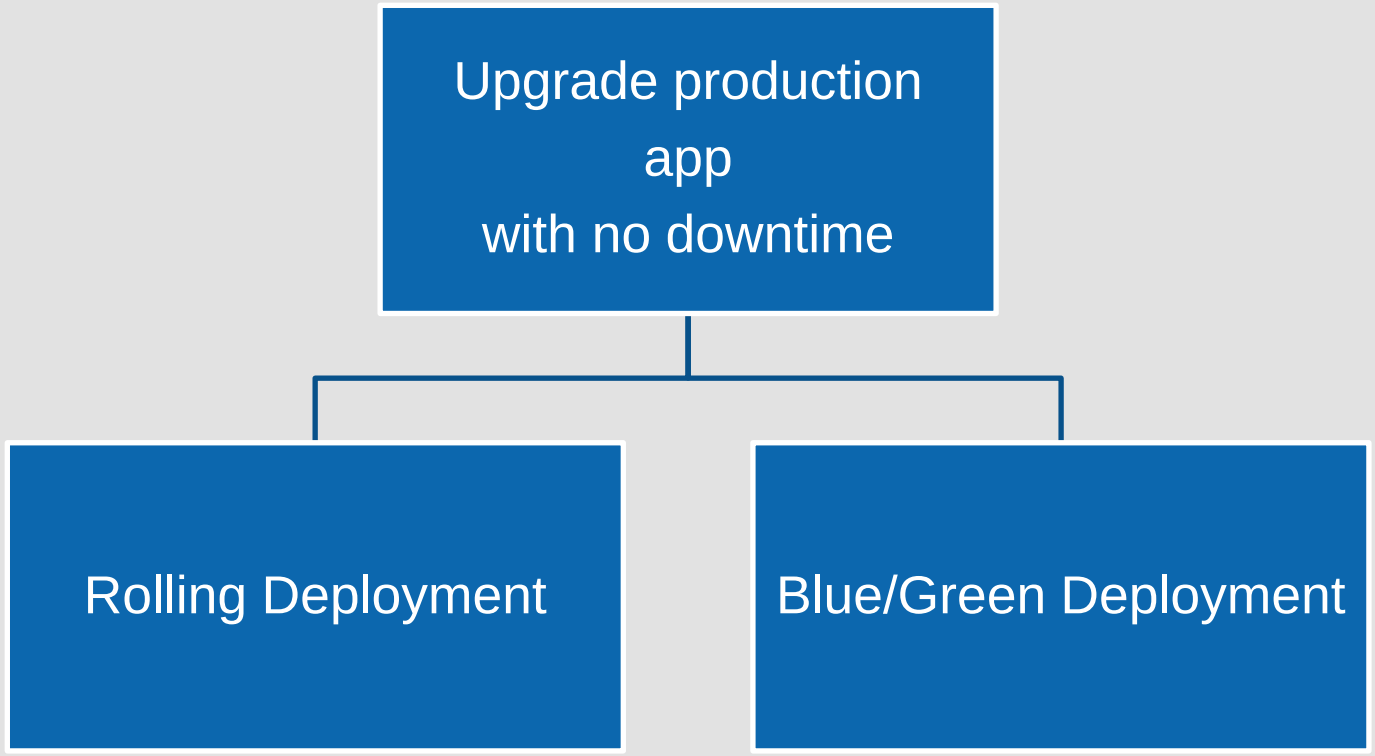


Publish logs to CloudWatch Logs using awslogs driver

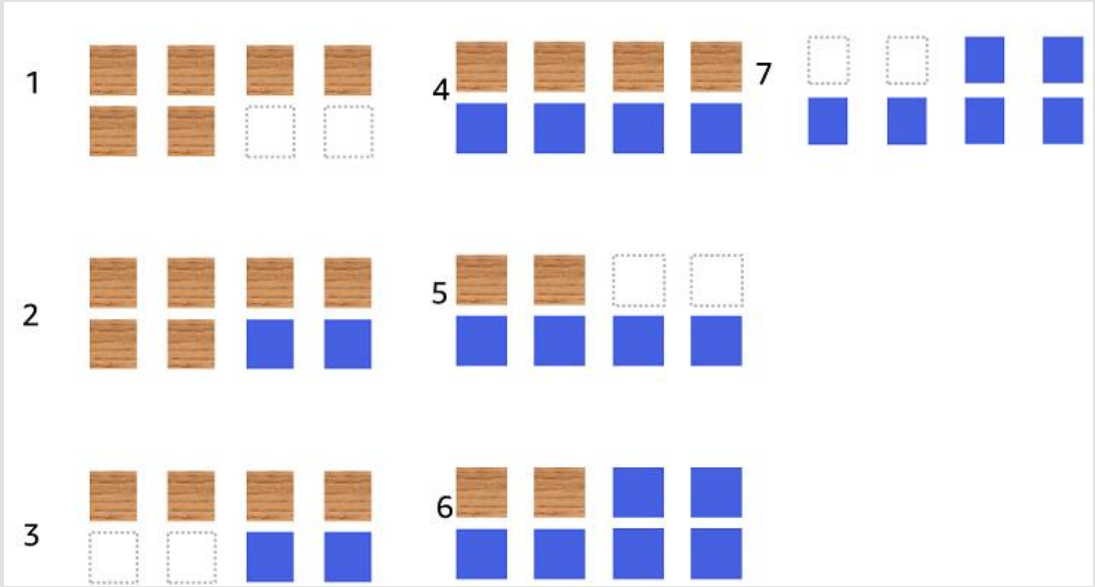
Captures messages in STDOUT and STDERR streams

Awslogs driver captures these logs to CloudWatch Log

Deployment Options



Rolling Deployment – Example 1

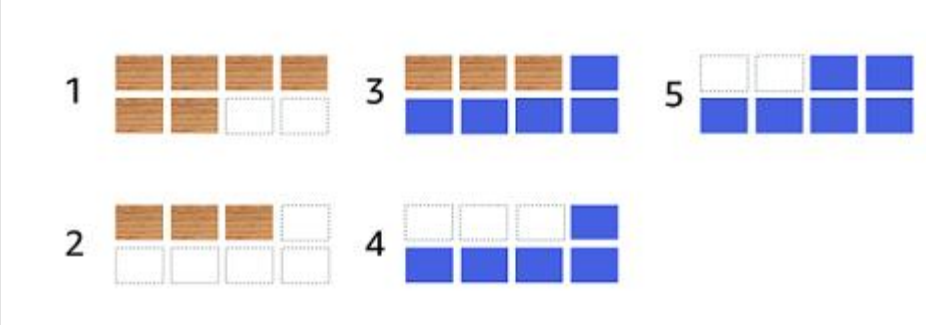


Desired: 6
Minimum: 100% Maximum: 200%

- Small number of new tasks are launched
- Wait until health check passes
- Remove old tasks
- Repeat until all old tasks are replaced with new

Reference: https://docs.aws.amazon.com/AmazonECS/latest/bestpracticesguide/service-options.html

Rolling Deployment – Example 2

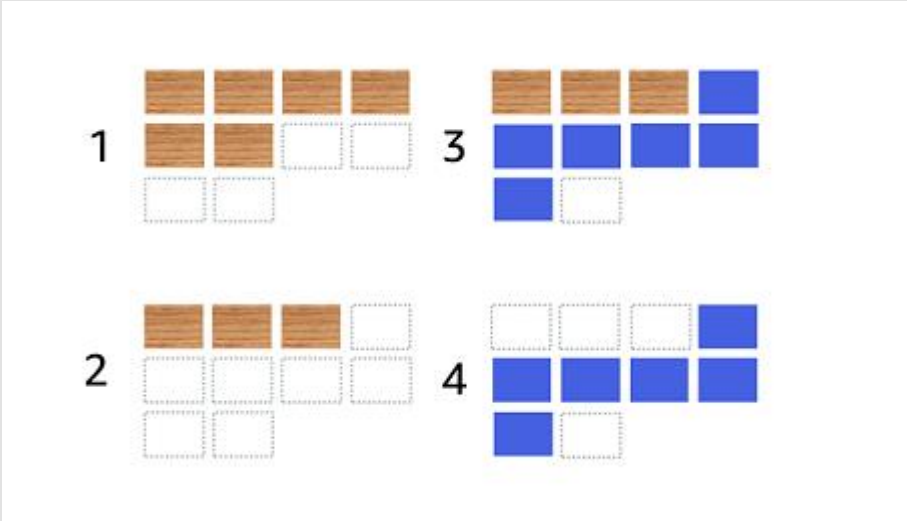


Desired: 6
Minimum: 50% Maximum: 200%

- We need to maintain a minimum of 3 healthy tasks
- Speed up deployment by adjusting min and max capacity

Reference: https://docs.aws.amazon.com/AmazonECS/latest/bestpracticesguide/service-options.html

Rolling Deployment – Example 3



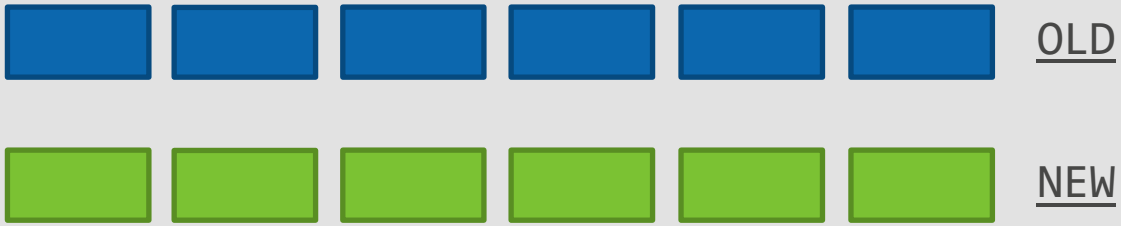
Desired: 6
Minimum: 50% Maximum: 200%

- Capacity increased to run up to 10 tasks
- We need to maintain a minimum of 3 healthy tasks always running
- Speed up deployment by adjusting min and max capacity

Reference: https://docs.aws.amazon.com/AmazonECS/latest/bestpracticesguide/service-options.html

Blue/Green Deployment

- Old and new version of software running side-by-side
- Rapidly respond to issues with new version
- More expensive



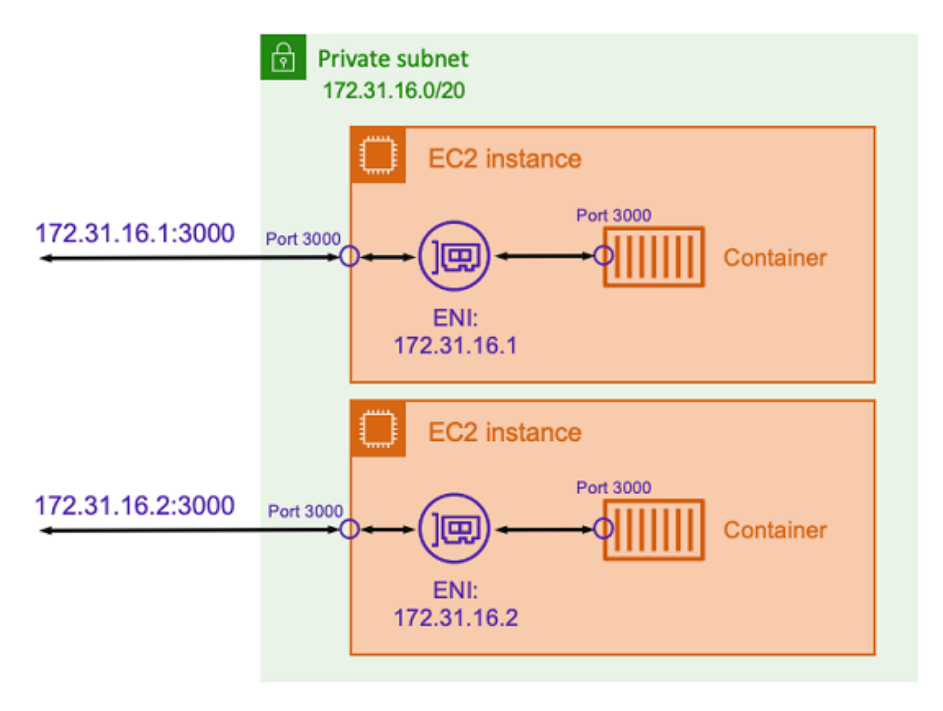
Networking Modes

Host

Bridge

AWSVPC

Host Mode



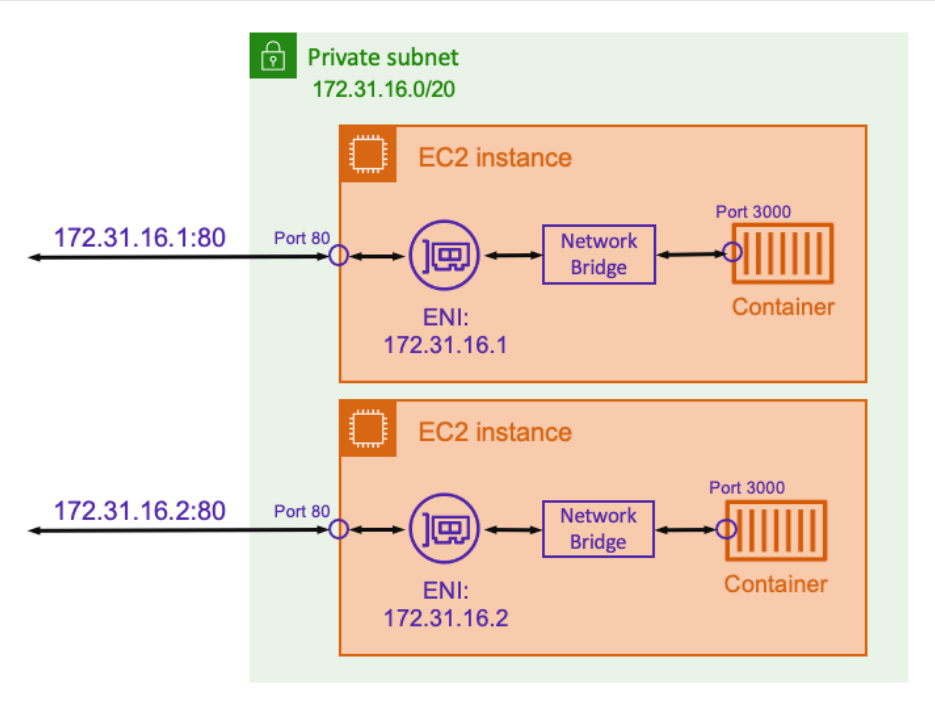
Container is directly connected to host networking

Drawbacks:

- Only one instance of task in each host
- Static port binding
- Security issues – container can access other services running in the host
- Supported only in EC2 launch type

Reference: https://docs.aws.amazon.com/AmazonECS/latest/bestpracticesguide/networking-networkmode-host.html

Bridge Mode (static port)



Virtual network bridge between host and container

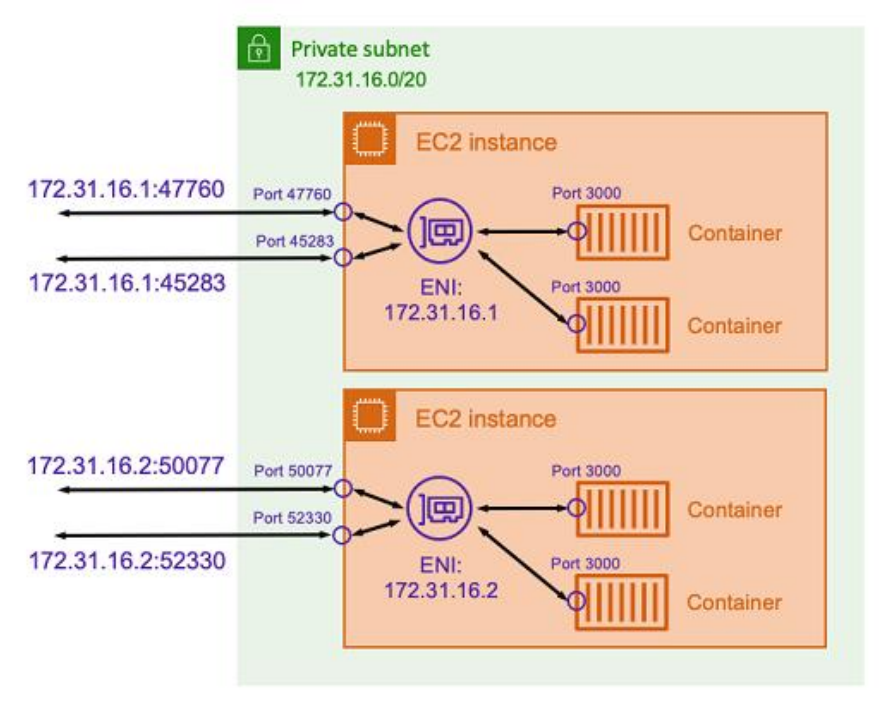
Remap Host port to container port

Static port Drawbacks:

- Only one instance of task in each host

Reference: https://docs.aws.amazon.com/AmazonECS/latest/bestpracticesguide/networking-networkmode-bridge.html

Bridge Mode (Dynamic port)



Virtual network bridge between host and container

Docker assigns a random host port for container

Hide dynamic ports from caller

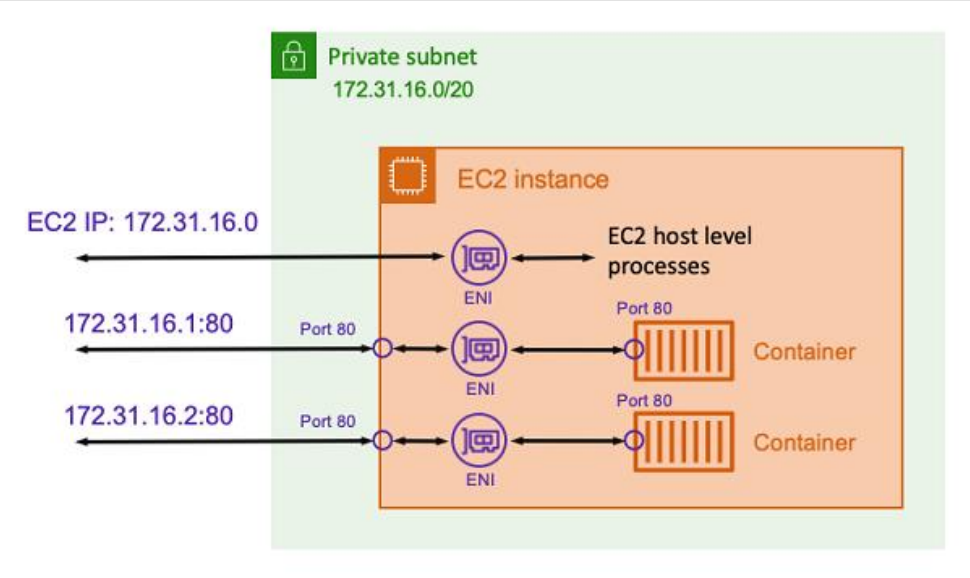
- ELB Target Groups tracks host IP, port to container mapping
- Cloud Map and App Mesh

Reference: https://docs.aws.amazon.com/AmazonECS/latest/bestpracticesguide/networking-networkmode-bridge.html

Bridge Mode Drawbacks

- Security Group and NACL Firewall - Need to allow traffic on a broad range of dynamic ports in the host
- Difficult to monitor traffic on these dynamic ports as we don’t know what ports are assigned by docker
- Bridge mode is supported only for EC2-launch types

AWSVPC Mode



Each Task is assigned a network interface and private IP

Containers can bind to static port

Any traffic to task IP address and port is routed to container

No need to use dynamic ports

Reference: https://docs.aws.amazon.com/AmazonECS/latest/bestpracticesguide/networking-networkmode-bridge.html

AWSVPC Advantages

- Specify Security group at Task level
- Monitor network traffic at task network interface
- VPC Flow to capture task specific traffic
- Fargate uses AWSVPC mode
- EC2 launch type also support this mode

AWSVPC Drawbacks

- EC2 limit on number of network interfaces
- For example, c5.large supports three network interfaces.
    - One is used for host and two are available for ECS
    - We can run only two tasks
- Enable ENI Trunking
    - C5.large can support up to 10 tasks with trunking
- Startup delay – every task requires a network ENI

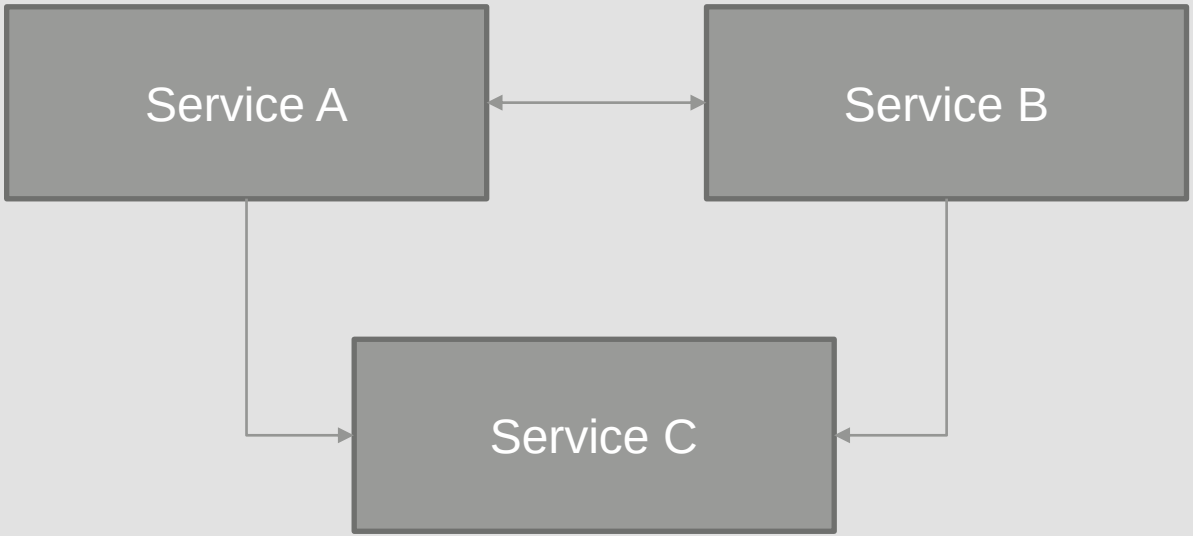
Service Discovery

ELB

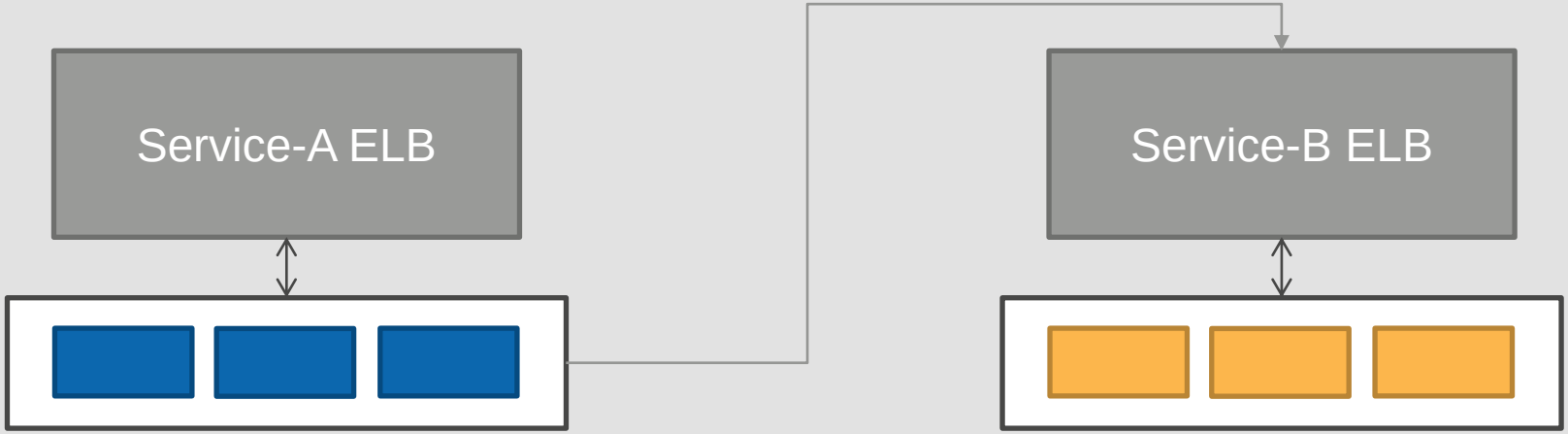
Cloud Map

App Mesh

Service to Service Communication

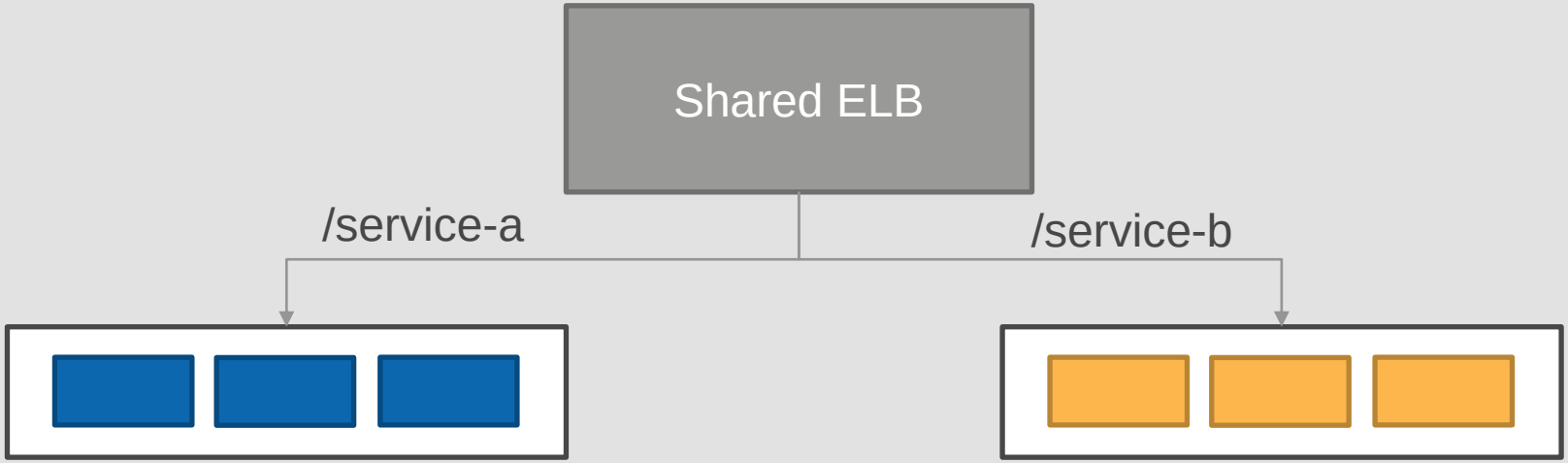


Use Internal Elastic Load Balancer



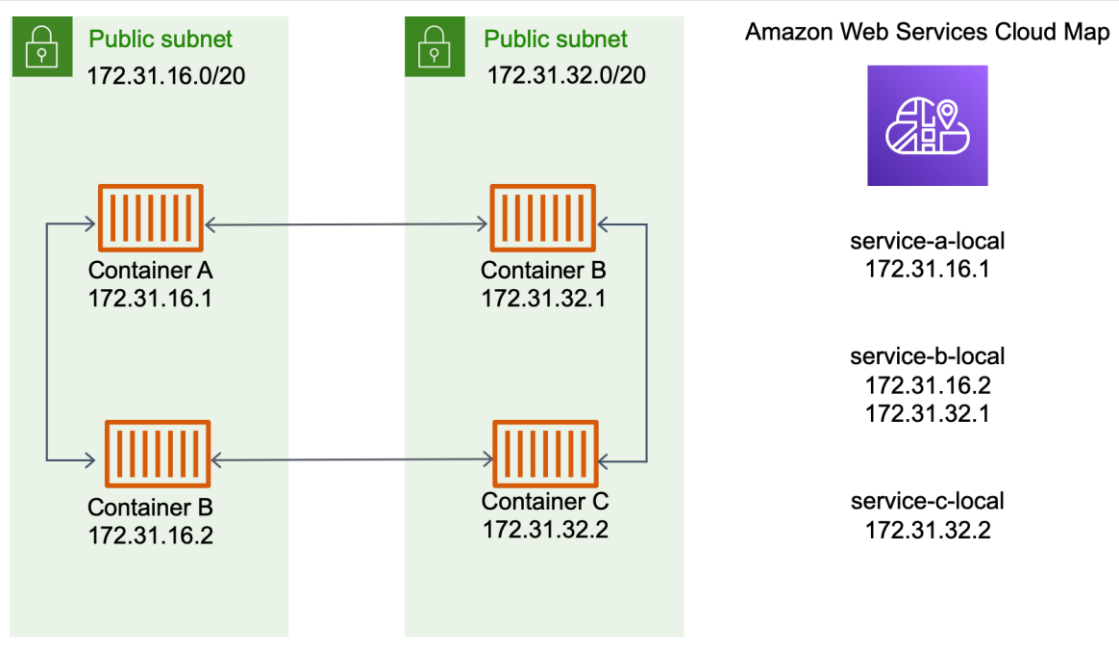
Advantages: Simple, Health Checks, Easy to scale
Drawbacks: Increased Cost

Shared Internal Elastic Load Balancer



- Single Load Balancer shared with many services
- Lower cost

Cloud Map



Maintain service domain names in Cloud Map

ECS automatically updates entries

Services use domain name to communicate

Low-latency, direct communication

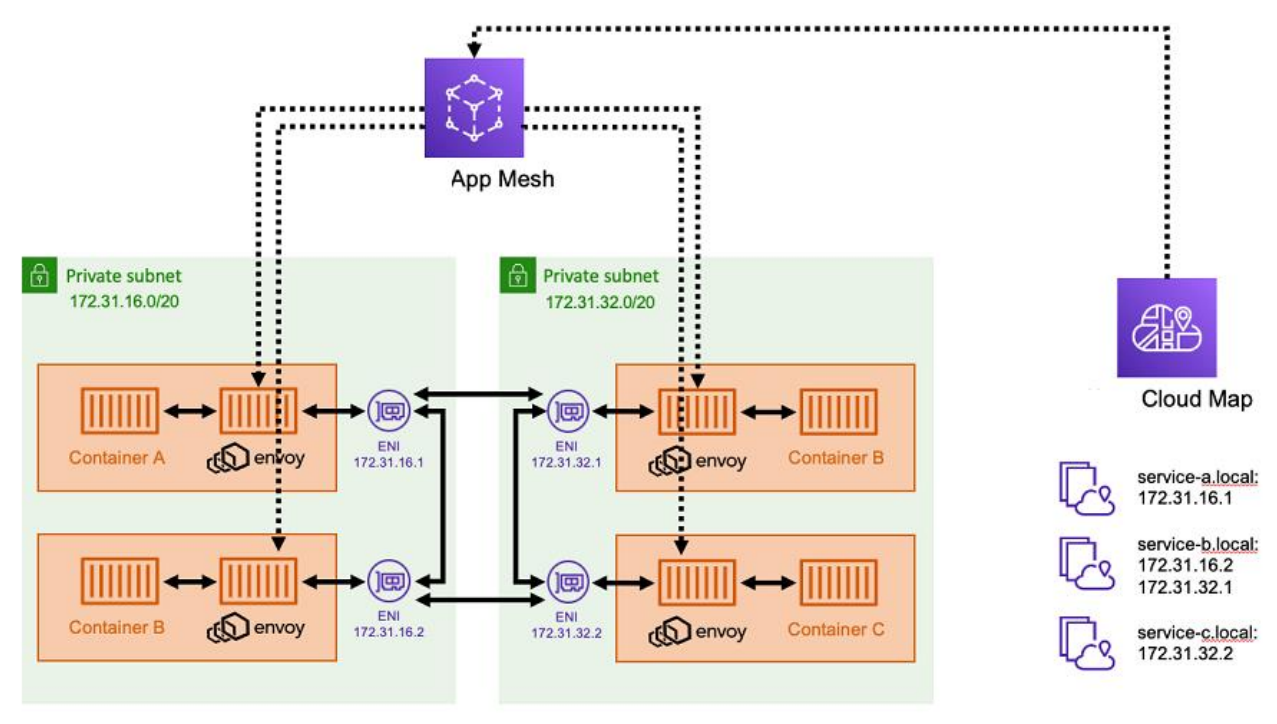
Works with AWSVPC Mode (each task assigned an IP)

Reference: https://docs.aws.amazon.com/AmazonECS/latest/bestpracticesguide/networking-connecting-services.html

Cloud Map Disadvantages

- Application needs to handle connection failure and retries
- Task list is maintained as DNS entries
- DNS entries are cached at client and may point to a non-existent task
- Application must handle these scenarios and redirect to a different task

App Mesh



Reference: https://docs.aws.amazon.com/AmazonECS/latest/bestpracticesguide/networking-connecting-services.html

App Mesh Advantages

- Service discovery with ease of managed load balancer
- Envoy proxy handles load balancing
- Envoy Proxy can detect failures and retry failed requests
- Automatically encrypt traffic using mTLS (mutual TLS). Verify destination and encrypt data in-transit
- Extremely low-latency communication

App Mesh Drawback

- Envoy proxy requires CPU and memory resources
- Need to maintain Envoy proxy

Chandra Lingam

75,000+ Students

Instructor, Course Developer

7X AWS Certified

For a list of courses, visit
https://www.cloudwavetraining.com/

Connect with me on LinkedIn
https://www.linkedin.com/in/chandralingam/

Cloud Wave LLC