EC2 Container Service (ECS)

Container Management Service

Content Prepared By: Chandra Lingam, Cloud Wave LLC

amazon web services

EC2 Container Service

- Scale and Run Docker Based Applications
- Automate EC2 Cluster Management
- Automate deployment of containers across EC2 instances
- Failure Detection and Recovery
- Integrates with other AWS Services including VPC, ELB,Storage, CloudWatch and more
- IAM Security Controls

Docker Overview

- Docker Containers – What is it?
  - Package your application and dependencies
  - Eliminate version and environment incompatibilities
  - Portable to any machine running docker
  - Secure – Isolated from other applications and underlying infrastructure
  - Lightweight
- Container Vs Virtual Machines

Demo – Hello World

- Install Docker on EC2 instance
- Run Hello World
- Resource: Demo1.txt

Demo – Hello World Web Application

- Hello World Python Web App
- Define Container
- Build the image
- Port Mapping
- Run the app – Interactive
- Run the app - Background
- Run one more app on a different port – Background
- Static host port allocation

Demo – Docker Repository

- Demonstrate Portability of Docker Images
- Upload Hello World Image to Docker Hub Registry
- Access image from anywhere and run containers
- Launch another instance and run image

Containers in the wild!

EC2 Container Service

Orchestrate and Manage Containers

Terminology

Term	Description
Image	Lightweight, Stand-alone, Executable Package
Container	Runtime instance of an image
Cluster	Logical grouping of EC2 Container Instances
Container Instance	EC2 instance on which the task runs on and is part of ECS Cluster
Task	One or more containers that form your application. Containers in a task are run together in the same EC2 instance
Scheduler	Responsible for placing tasks on Container Instances
Container Agent	Runs on each EC2 Container Instance. Reports current tasks, resource utilization to ECS and Starts/Stops tasks whenever it receives requests from ECS

Terminology

Term	Description
ECS Instance Role	IAM Role for the EC2 Container Instance. Need instance level permissions including access to ECS from Container Agent
ECS Task Role	IAM Role for individual Tasks. Fine grained based on task specific access needs. Remember an ECS Instance can host several different types of tasks with different access needed. Task Role helps you accomplish this.
ECS Task Execution Role ***2018***	This takes the place of instance role for Fargate Tasks. Grants permissions to ECS to pull private images from ECR, publish logs to CloudWatch logs on your task behalf.

ECS Architecture

Different Components at Play:

Figure: Architecture

Figure: Scheduling

Figure: Container Agent

Fargate

Serverless offering for running Containerized applications

Package your application as containers

Configure CPU, Memory, Networking, IAM Policies

Scale very rapidly to supports tens of thousands of containers

Fargate Architecture

Tasks and Scheduling

Networking: awsvpc mode

- Each Task has same networking properties as EC2 instance
- Task gets an elastic network interface, Private IP address, Internal DNS name, optional Public IP address, security group

Fargate Pricing and Configuration

Pricing

- Per CPU $0.05/hr,
- Per GB Memory $0.0127/hr
- Per Second billing with 1 minute minimum
- Time starts from image download and ends when task is terminated
- Supported CPU and Memory Configuration

Task Invocation

Service – For long running tasks like microservices

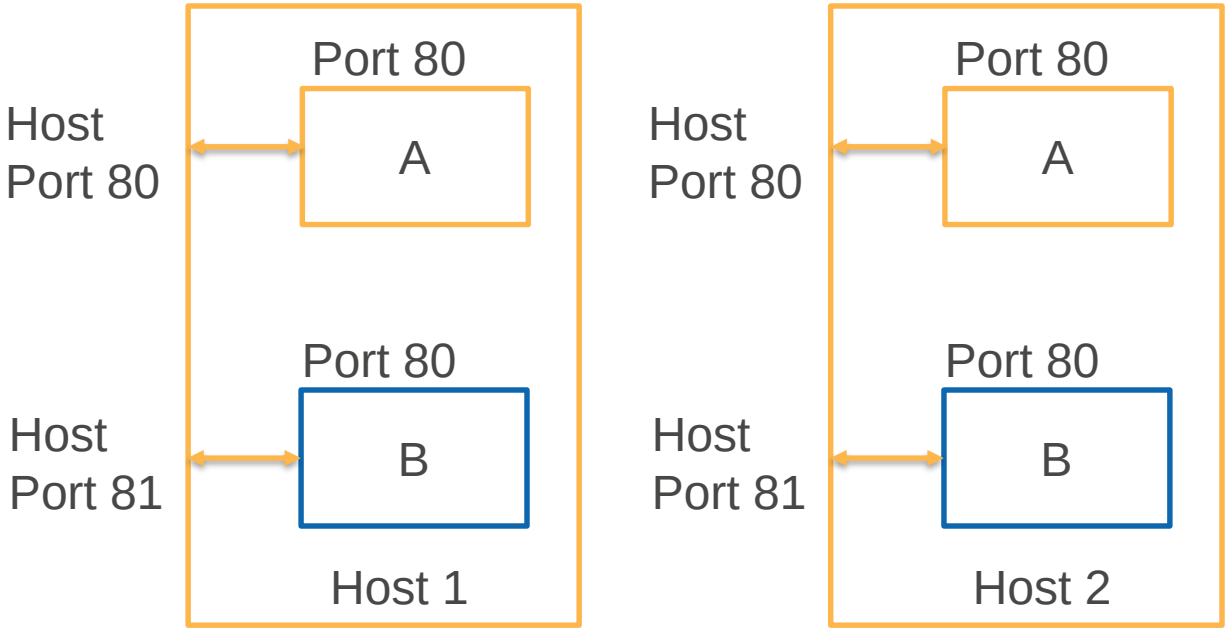
Scheduled – For time based invocation

Ad-hoc/On-demand – Invoke when needed

Demo – Setup ECS

- Configure ECS Cluster (1 EC2 instance) with Container Optimized AMI
- Configure Task Definition for HelloWorld Python Image
- Run Task with Static Port Mapping (manual)
- Run Multiple Tasks with ephemeral Port Mapping (manual)

Static Port Mapping



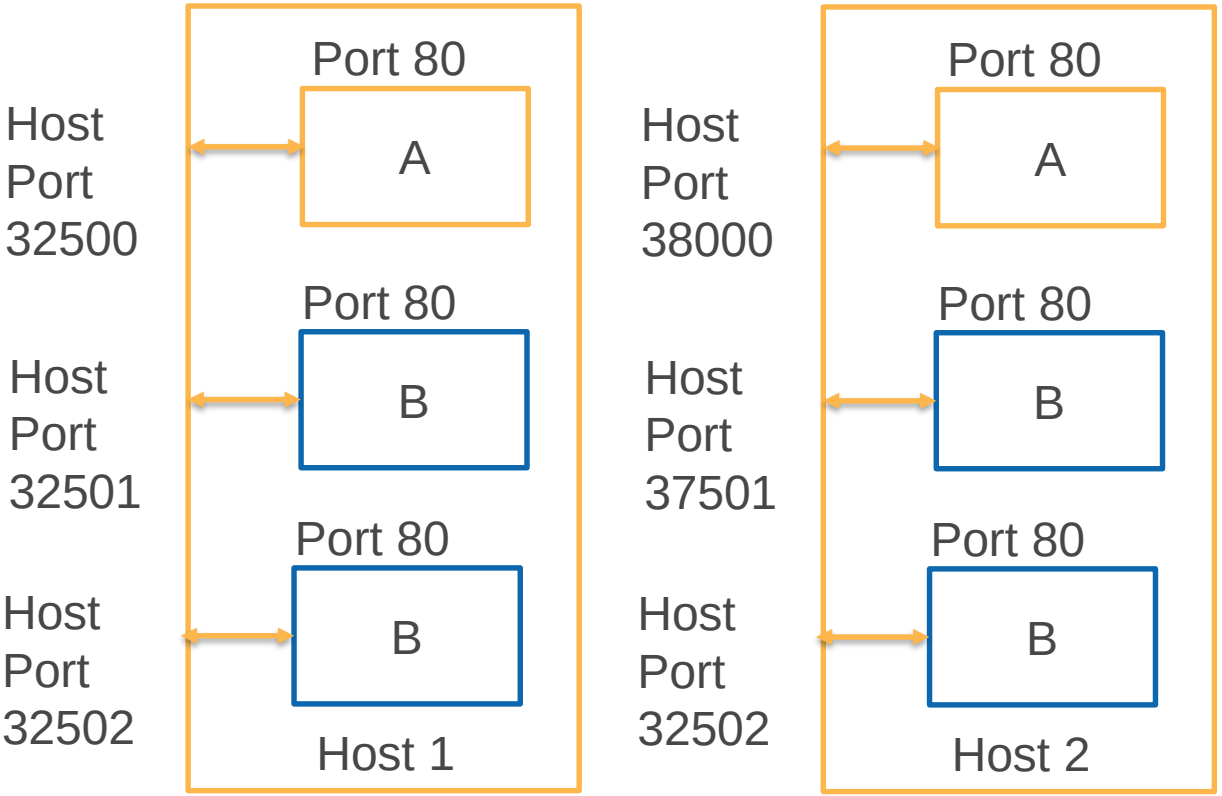
Host Port to Container Port mapping is hardcoded

Requires coordination between container teams

Limits flexibility on how many containers can be deployed

Only one container of a particular image can be deployed in a host

Dynamic Port Mapping



Host Port to Container Port mapping is dynamic

Auto assigned. A container port can map to completely different ports across EC2 Instance

Multiple containers of a particular image can be deployed in a host

Application Load Balancer hides all this complexity from end users

amazon web services

Application Load Balancer – Dynamic Mapping

- Containers are part of Target Group
- ALB maintains Target Group mapping. For every container, it tracks (Instance ID, Port)
- New Containers that are part of target group will automatically start receiving traffic

Demo – Setup Application Load Balancer

- Publish your container service from well defined end point
- Automatic discovery and registration of new containers to application load balancer
- Task Ephemeral port mapping maintained by ALB (Instance ID : Port)

Demo – Scaling Demo

- Add additional instances to ECS Cluster
- Deploy Additional Tasks to ECS Cluster
- Access with same ELB end point

Demo – Fargate Task

- Use docker hub image created in previous demos
- Fargate Task Definition
- Fargate Cluster
- Run Tasks

Demo – Fargate with Network Load Balancer

- Setup a Network Load Balancer (NLB)
- Configure NLB Target Group for Containers
- Configure Fargate Service to process Web requests
- Use NLB to load balance traffic to the containers

ECS Pricing

ECS Pricing

- No Additional charge for using ECS
- You pay only for AWS resources created to run your application (EC2 instances, Storage Volumes, ALB and so forth)

ECS CloudWatch Metrics

CloudWatch Metrics

AWS ECS Scheduler

AWS re:Invent 2016: Advanced Task Scheduling with Amazon ECS and Blox (CON307)

- First 23 minutes gives an excellent overview of scheduling options and Event Stream (must watch)

AWS re:Invent 2016: Introduction to Container Management on AWS (CON303)

- First 19 minutes general framework discussion
- AWS ECS from 40th minute onwards