AWS Certified Solutions Architect - Associate

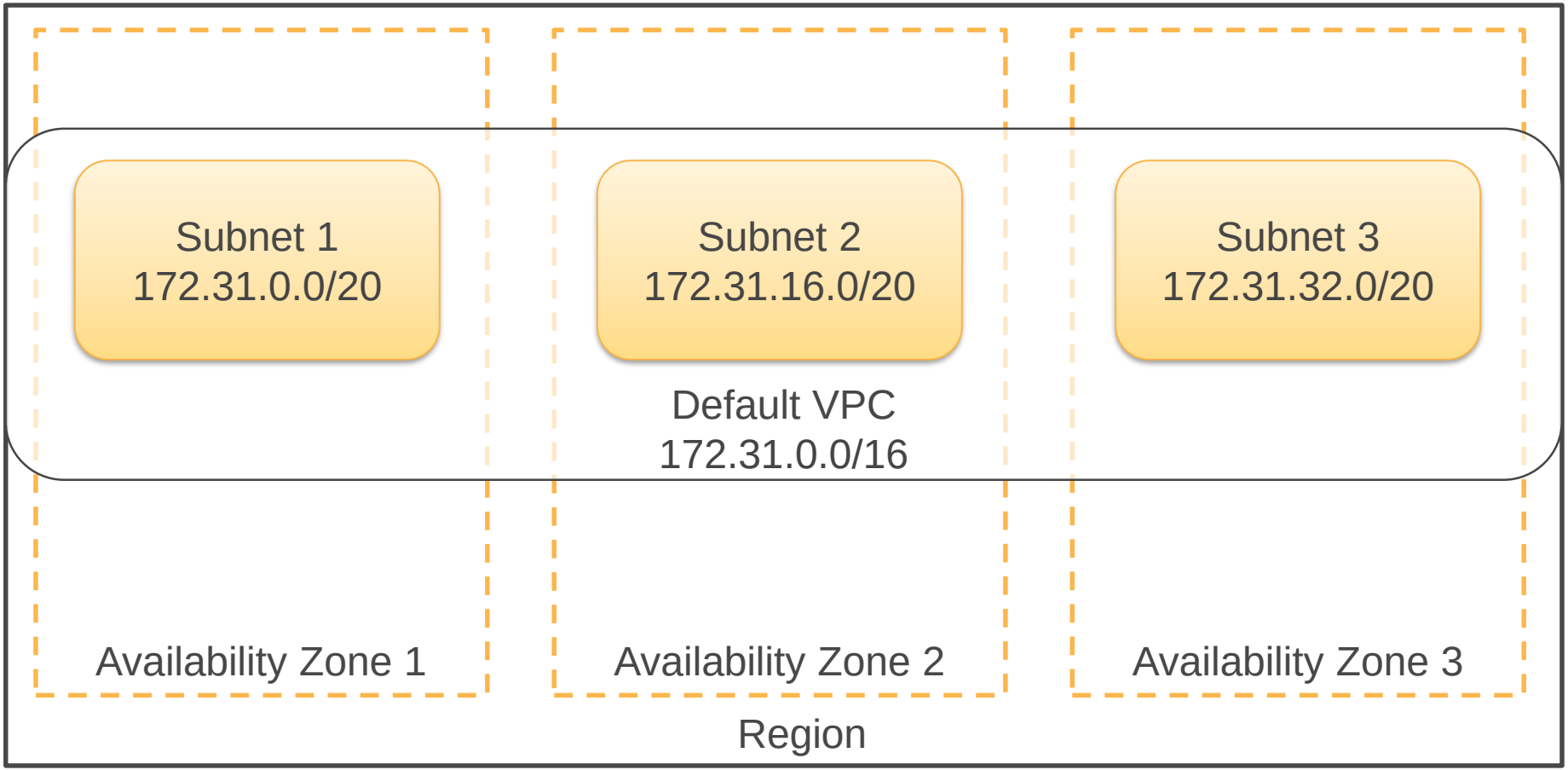
Practice Quiz + Video - Reference Material

Content Prepared By: Chandra Lingam, Cotton Cola Designs LLC

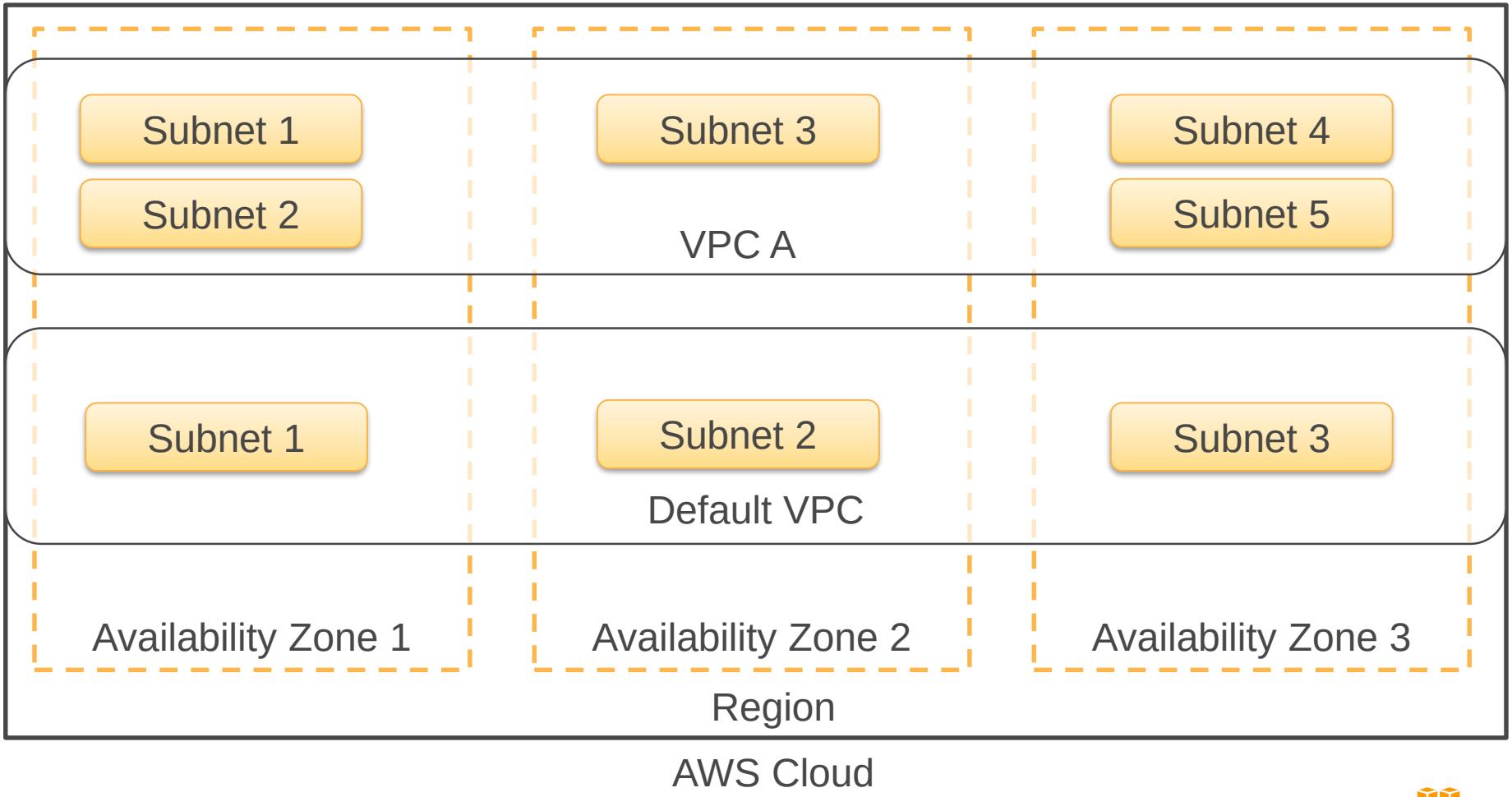
amazon web services

VPC

- Virtual Network Dedicated to your AWS Account
- Logically isolated from other virtual networks in the AWS Cloud
- Launch resources such as EC2 instances in your VPC
- Select your own IP Address range
- Create Subnets
- Configure route tables, network gateways
- Support for IPv4 and IPv6
- Simple to use



amazon web services



amazon web services

VPC Components

Component	Description
VPC	Isolated virtual network in AWS cloud
Subnet	Isolated segment of your VPC
Internet Gateway	VPC side of connection to internet
NAT Gateway	AWS managed Network Address Translation Service to make outbound internet connection from your private subnet (IPv4)
NAT Instance	Customer managed NAT (IPv4)
Egress-only Internet Gateway	IPv6 outbound internet access

VPC Components

Component	Description
Hardware VPN Connection	Secure connection between your datacenter and VPC
Virtual Private Gateway	Amazon VPC side of VPN connection
Customer Gateway	Customer side of VPN connection
Router	Routes traffic inside VPC
Peering Connection	Connect two VPCs and access resources with private IP address
VPC Endpoint	Access AWS resources like S3 without using NAT or Internet Gateway Control access to resources from specific VPCs

VPC Peering Connection

- VPC Peering connection allows you connect two VPCs together and route traffic using private IPv4 addresses or IPv6 addresses
- Address should not overlap between VPCs
- Instances communicate as if they are within same network
- Supported for VPCs across regions (Newly Supported!)
- VPCs can be part of one account or different accounts
- Owner of the peer VPC needs to accept the request

VPC Peering Connection

- Only one peering connection between two VPCs
- Multiple peering connections are supported from one VPC to multiple VPCs

Network

- Launch instances in your virtual private cloud (VPC)
  - Assign your own address range
- Keep instances in public subnet – for internet accessible systems
- Keep instances in private subnet – to restrict access and reduce footprint

Bastion Host

- Bastion Host is used to access your private resources from public internet
  - EC2 instances in private subnet allows SSH/RDP only from Bastion Host
  - Bastion Host on public subnet – allows access from specific IP address range for SSH/RDP access
- Reduce attack surface by controlling access points
- Harden to protect your resources
- Do not place your private key in bastion host – use SSH agent forwarding for connecting to private EC2 instances
- Windows Remote Desktop Gateway

Operating Systems

- Numerous Linux distributions
  - Amazon Linux, Red Hat, SUSE, Fedora, Ubuntu and more
- Microsoft Windows
- FreeBSD - marketplace

Amazon Machine Image (AMI)

- Amazon Machine Image provides information to launch an instance
- Template for root volume: OS, application server, applications
- Additional volumes that needs to be attached to the instance
- Permissions on who can launch an instance
- Several choices from Amazon, vendors and community
- Create your own, buy, share, and sell

Amazon Linux AMI

- Amazon provided and maintained Linux image
- Stable, secure, high-performance environment for EC2
- No additional charge
- Repository access to multiple versions of common packages
- Updated on regular basis include latest components
    - Can be used to update running instances through repository
- Includes AWS packages for integration – CLI, API, AMI tools, Boto library for python, ELB tools

Dedicated and Shared Resource

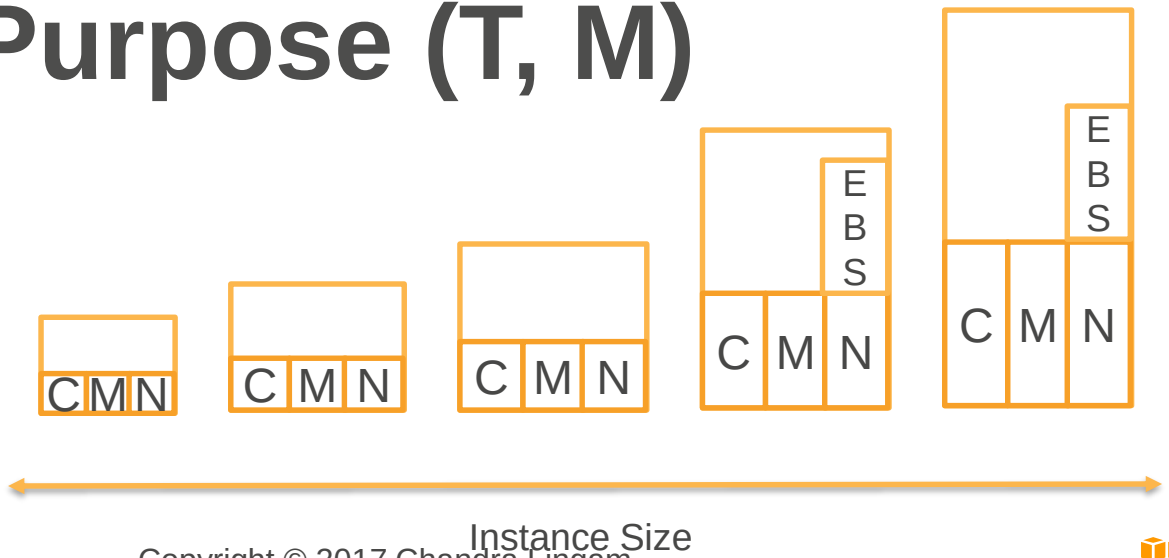
- EC2 dedicates some resources of host computer to each instance: CPU, memory, instance storage
- EC2 shares common resources like disk sub system and network
- When shared resource is underutilized - instance can consumer higher share
- When shared resource are in demand – each receives an equal share
    - High I/O performance instance types allocate larger portion of a shared resource
    - Greater or more consistent I/O performance

Instance Families

- General Purpose (T, M)
- Compute Optimized (C)
- Memory Optimized (X, R)
- Storage Optimized (I, D)
- Accelerated Computing (P, G, F)

Choice of CPU, Memory, Storage, Network, Hardware Acceleration for your needs. Determines the hardware of the host computer used

General Purpose (T, M)



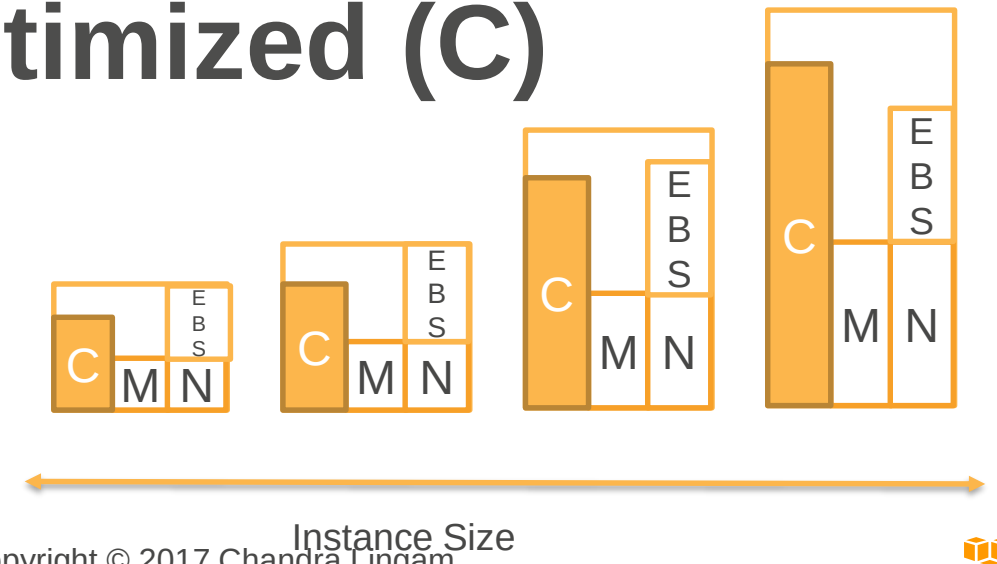
General Purpose – T2 instances

- Lowest cost general purpose instance type - Balance of compute, memory and network resources
- T2. micro eligible for free tier
- Baseline CPU performance with ability to burst
- Burst is governed by CPU credits - Accrue CPU credits when idle and use it when needed
- Good choice for workloads that doesn’t use full CPU but burst occasionally
- Suitable for Webserver, development environments and databases

General Purpose – M4 instances

- Latest generation and provides a balance of compute, memory, network resources
- Good choice for many applications
- EBS optimized at no additional cost
- Support for enhanced networking
- M3 Instance - SSD Based instance storage for fast I/O performance
- Suitable for small-mid sized databases, data processing, cluster compute, sharepoint

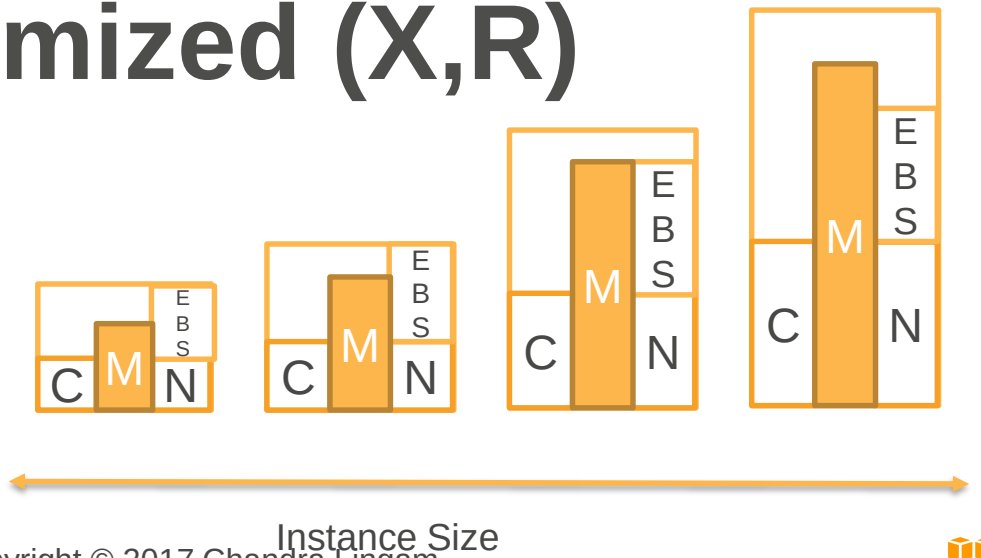
Compute Optimized (C)



Compute Optimized – C4

- Latency gen, highest performing processors
- Lowest price per compute performance in EC2
- EBS optimized at no additional cost
- Support for enhanced networking and clustering
- Ability to control processor C-state and P-state configuration on large instances
- C3 – SSD based instance storage
- MMO gaming, Video encoding, Distributed analytics, batch processing, science and engineering use

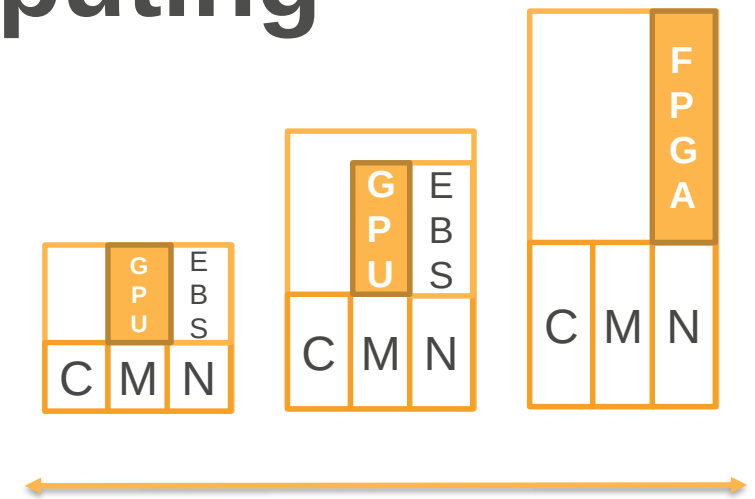
Memory Optimized (X,R)



Memory Optimized – X1

- Optimized for large scale in-memory applications
- Lowest price per GiB of RAM among EC2 instances
- Upto 1,952 GiB of instance memory
- SSD Instance storage
- EBS Optimized at no additional cost
- Ability to control processor C-state and P-state configuration
- Certified -SAP HANA, Apache Spark, Presto, HPC apps
- Smaller R4 and R3 instances available

Accelerated Computing (P,G,F)



Accelerated Computing – P2

- General purpose GPU compute applications
- High performance NVIDIA K80 GPUs
- GPUDirect support for GPU-GPU peer communication
- Enhanced networking upto 20Gbps
- EBS optimized at no additional cost
- Machine Learning, High performance databases, computational fluid dynamics, seismic analysis, rendering, genomics workloads

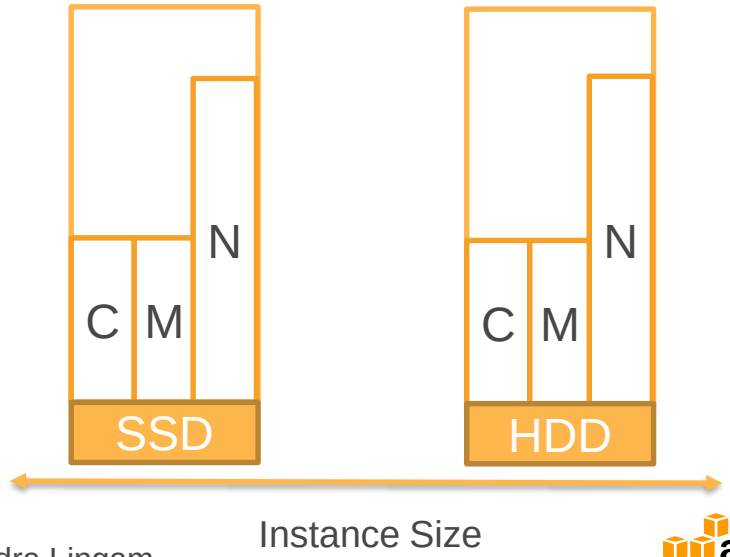
Accelerated Computing – G2

- Optimized for Graphics intensive applications
- High performance NVIDIA GPUs
- On-board hardware decoder for multiple real-time HD streaming
- Low latency frame capture and encoding – High quality interactive streaming experience
- 3D application streaming, video encoding, server side graphic workload

Accelerated Computing – F1

- Customized hardware acceleration with field programmable arrays (FPGA) -Xilinx
- NVMe SSD storage
- Support for Enhanced networking
- Genomics research, financial analytics, real-time video processing, security, big data search and analysis

Storage Optimized (I,D)



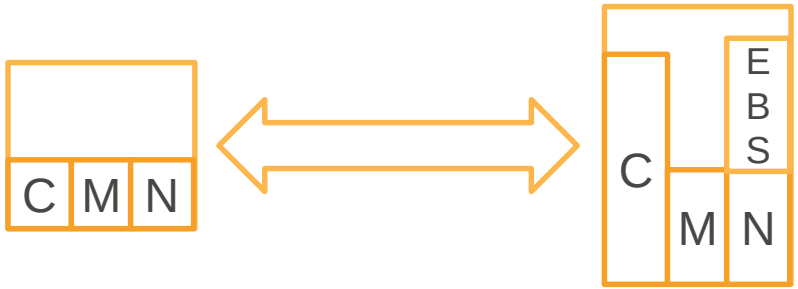
Storage Optimized – I2

- High storage instances with SSD backed instance storage
- Very high random I/O performance
- High IOPS at low cost
- Support for enhanced networking
- NoSQL databases Cassandra, MongoDB, scale out transactional databases, cluster filesystems, data warehousing, hadoop

Storage Optimized – D2

- Dense storage instances – 48TB of HDD local instance storage
- High disk throughput
- Lowest price per disk throughput
- Massively parallel data warehousing, Hadoop Map Reduce, Distributed file systems, network file systems, log or data processing applications

Resizing Instances



Resizing Instances

- Resize an existing instance based on your usage – over or under utilization
- Stop instance, update to new instance type, restart
- Only supported for Instances with EBS root device volume. Not supported on Instance store root device volumes
- Target instance type must be compatible
    - Virtualization Type. HVM <-> PV not allowed
    - 32 bit <-> 64 bit not allowed
    - Some instances are restricted to VPC. You cannot use in EC2-Classic

Firewall

- Security Groups – Mandatory firewall for EC2 instances
  - Applies to all Inbound and outbound traffic at Instance level
  - Stateful filters
- Network Access Control Lists (ACL)
  - Applies to all inbound and outbound traffic from a subnet in VPC
  - Stateless traffic filters

Storage Options

- Amazon EC2 Instance Store
- Amazon Elastic Block Store (EBS)
- Amazon Elastic File System (EFS)
- Amazon Simple Storage Service (S3)
- Figure: Storage

Cross Zone Load Balancing

- For fault tolerance, EC2 instances should be distributed across two or more availability zones
- Cross Zone Load Balancing controls how traffic is distributed across Availability Zones and Instances in each Availability Zone

Load Balancer Types

- Classic Load Balancer
  - Simple load balancing across EC2 instances
  - Supports HTTP, HTTPS, TCP, SSL (Secure TCP)
  - OSI Layer 4 (Transport), 7 (Application) Load Balancer
- Application Load Balancer
  - Path based routing
  - Route traffic to multiple services
  - Route traffic to different ports on the same EC2 instance
  - Ideal for microservices and container based architectures
  - Supports HTTP, HTTPS, HTTP/2, WebSocket
  - OSI Layer 7 (Application) Load Balancer

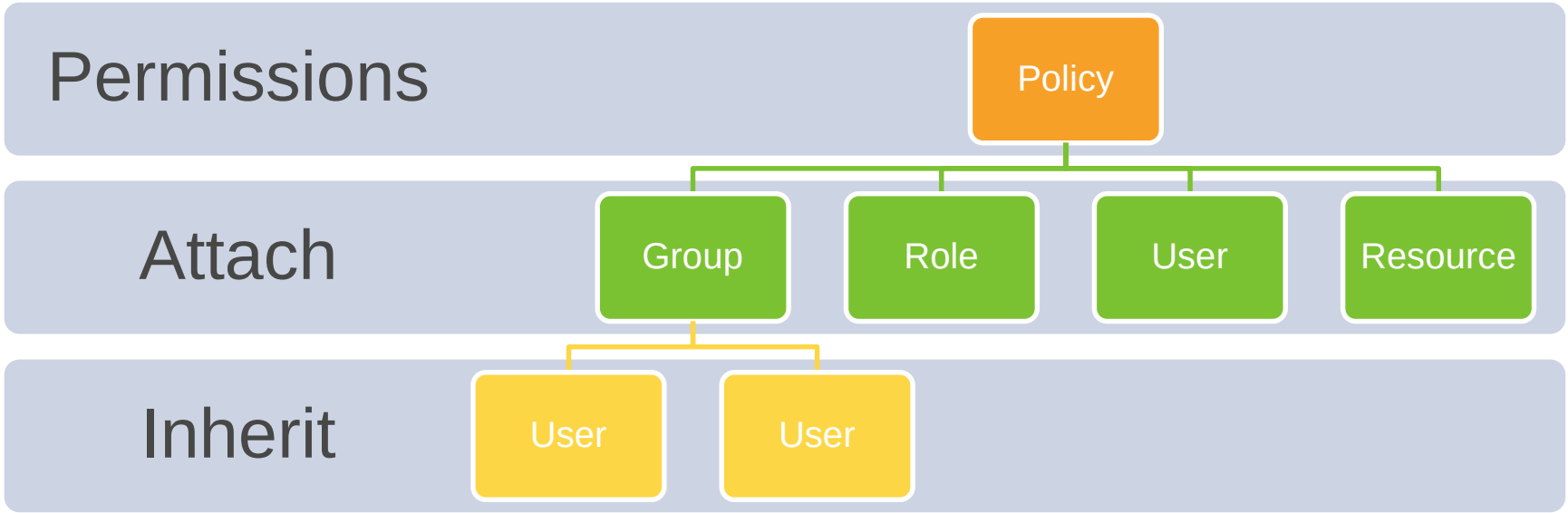
Disabled - Cross Zone Load Balancing

- Distribute traffic evenly across Availability Zones
  - Happens when Cross Zone load balancing is “Disabled”
  - Two availability zones ‘A’ and ‘B’ would each receive 50% of the traffic irrespective of number of EC2 instances in each Availability Zone
  - May cause higher loading if one Availability Zone has fewer EC2 instances
  - Default mode in classic load balancer

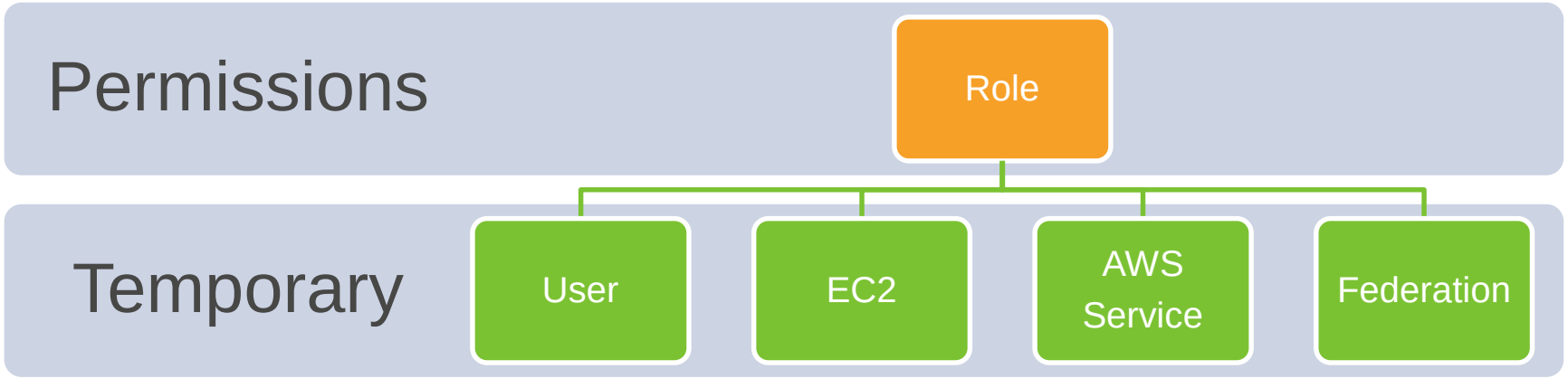
Enabled - Cross Zone Load Balancing

- Distribute traffic evenly across EC2 instances in all Availability zones
  - Happens when Cross Zone load balancing is “Enabled”
  - Availability Zone ‘A’ has 3 instances and Availability Zone ‘B’ has 2 instances. Each instance would receive 20% of the traffic (1/5th)
  - Default mode in application load balancer

IAM Concepts

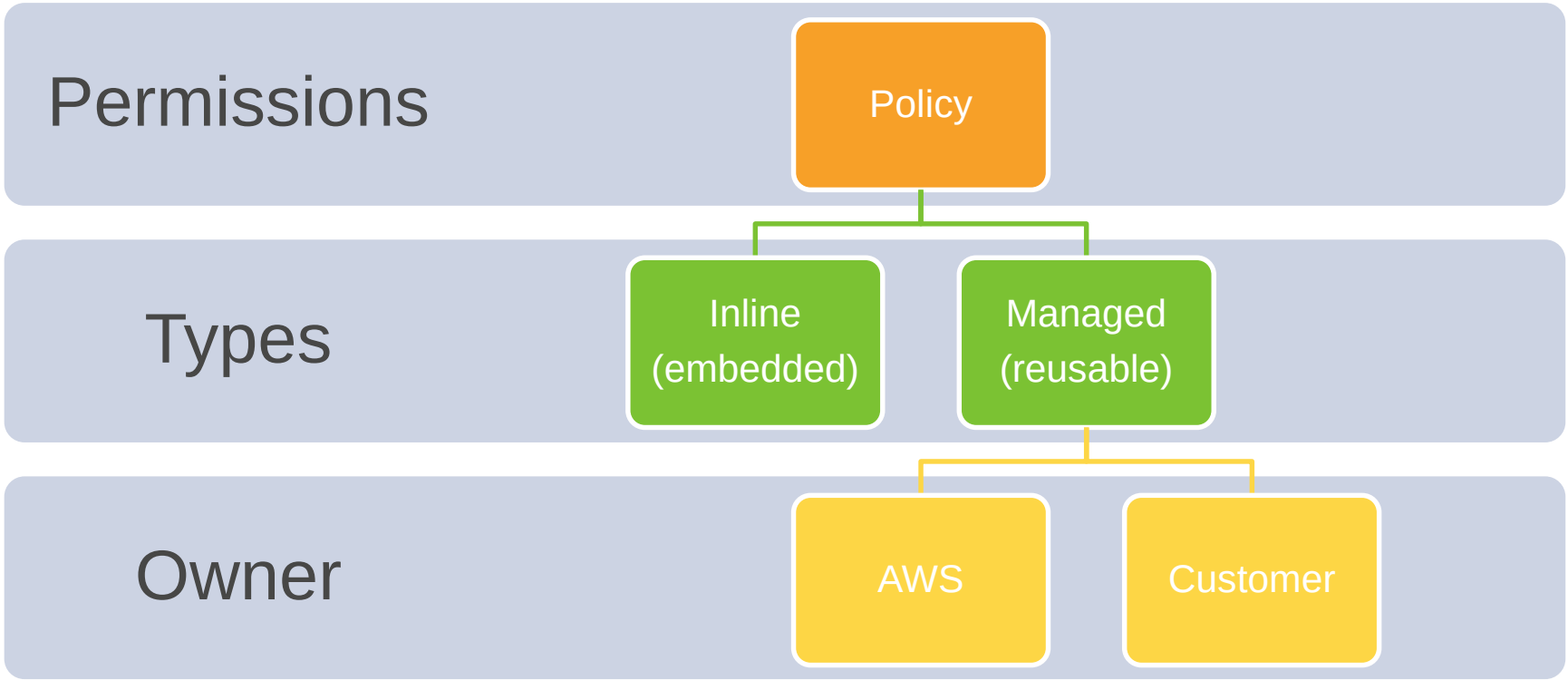


IAM Role



Role has two parts: 1. Who can assume the role and 2. What permissions does a role have

IAM Policy Types



Amazon Simple Storage Service (S3)

Content Prepared By: Chandra Lingam, Cotton Cola Designs LLC

amazon web services

Storage Class	Standard	Standard – Infrequent Access	Reduced Redundancy Storage	Glacier
Usage	Frequently Accessed Data	Less Frequently Accessed Data	Frequently Accessed non-critical data	Rarely Accessed. Data Archiving
Durability	99.999999999%	99.999999999%	99.99%	99.999999999%
Availability	99.99%	99.9%	99.99%	N/A
Availability SLA (Service Credit)	99.9%	99%	99.9%	N/A
Concurrent Facility Failure	2	2	1	2
Redundancy (Region)	Multiple devices in multiple AZ	Same as standard	Fewer copies	Same as standard
First Byte Latency	Milliseconds	Milliseconds	Milliseconds	4 hours
Minimum Storage Duration	N/A	30 days	N/A	90 days
Minimum Size		128 KB (minimum)		
x-amz-storage-class	STANDARD	STANDARD_IA	REDUCED_REDUNDANCY	GLACIER

S3 Data Protection

- In-transit protection
  - HTTPS endpoints for AWS Services
  - Client side encryption
- Data at rest
  - S3 Server Side Encryption – S3 encrypts object when storing and decrypts when retrieving
  - Client Side Encryption – Encrypt data on your side and upload encrypted data to S3. Encryption process, keys, and tools are managed by client

S3 Server Side Encryption

- S3 encrypts when writing object and decrypts when reading object
- For authorized users, no difference between encrypted and unencrypted object – Transparently handled by S3
- Three key management choices
    - S3 managed keys (SSE-S3)
    - Key Management Service managed keys (SSE-KMS)
    - Customer provided keys (SSE-C)
- AES256 Encryption Algorithm
- Object Data is encrypted. Metadata is not encrypted

S3 Access Management

- User Policies (covered in IAM Lecture)
- Resource Policies
    - Bucket Policies (covered in IAM Lecture)
    - Bucket Access Control List (ACL)
    - Object Access Control List (ACL)
- When to use ACLs?

S3 - Object ACL

- Control permissions at object level - Permissions vary by object
- If Object owner is different from bucket owner – Object ACL is the only way object owner can grant permissions
    - Bucket owner cannot read until given permission
    - Bucket owner can deny access to object
- No user level permissions. Only at account level
- Grantee: Another Account or predefined S3 groups. Account can be referred by email address or Canonical ID

S3 - Bucket ACL

Only recommended use for Bucket ACL

- Grant access to S3 `Log Delivery Group` to write S3 access logs to your bucket
- Bucket ACL is the only way in which `Log Delivery Group` can be granted access
- No user level permissions. Only at account level
- Grantee: Another Account or predefined S3 groups. Account can be referred by email address or Canonical ID

S3 - Bucket Policy

- User Policy or Bucket Policy – to manage access within same AWS account
- Bucket Policy
  - Can be used to grant cross-account access permissions for all S3 Actions (no need to use IAM role as a proxy)
  - Bucket ACL can also grant cross-account access but only for some S3 Actions

Route 53 and DNS Concepts

Terminology	Description
Generic Top Level Domain (TLD)	Last part of a domain name (.com, .org, .cloud).
Geographic Top Level Domain	Domains associated with geographic areas. (.uk, .fr, .io, .in)
Domain Name System (DNS)	Worldwide network of servers that maintains domain names to IP Addresses
Name Servers (NS)	Servers in DNS that respond to DNS queries
Authoritative Name Server	NS that has definitive information about one part of a domain name

Route 53 and DNS Concepts

Terminology	Description
Hosted Zone – Route 53	A container that has information on how to route traffic for a domain (example.com) and sub domains (www.example.com, retail.example.com)
Resource Record Set	Configuration that maps domain name to resources that can process the request. Several types of resource records are supported
Time To Live (TTL)	Time in Seconds a particular Resource Record Set can be cached
Alias Resource Record Set	Route 53 specific extension to route traffic to AWS resources such ELB, S3, CloudFront and so forth – automatically tracks backend resources. TTL setting is inherited from target service. Cannot change in Route 53

Route 53 Routing Policy

Routing Policy	Description
Simple Routing	Used when you have a single resource performing a function. For Example, one web server serving content. In Simple Routing, Route 53 simply returns the configured values for matching resource recordset
Weighted Routing	Used when you have multiple resources performing similar function and you want to route traffic to resources in proportions that you specify. For example: Several web servers serving content, A/B testing
Latency Routing	Used when you have deployed your application across multiple regions and want to route customers to resources that offer best possible latency.
Failover Routing	Active-Passive failover support. All traffic is routed to Primary endpoint (known as Active). If primary is down, then all traffic is send to Second endpoint (known as Passive).
Geolocation Routing	Used when you want to route traffic to resources in the same geography as your users. Can be used for compliance requirements. You can support a default record set to handle requests where you don't have resources. Otherwise, Route 53 will return a "No Answer" response

SQS Types

Standard Queues – Maximum throughput, best effort ordering, and at-least-once delivery

FIFO Queues – Limited throughput, exact ordering, and exactly-once processing

Table: Comparison of SQS Queue Types

Standard Queue Concepts

At-Least-Once Delivery

- On rare occasions, you might receive duplicate messages with Standard Queues
- Design your application to handle duplicate messages

Figure: Sampling

FIFO Queue Concepts

Exactly-Once Processing

- No duplicate messages sent to receivers

Deduplication

- Helps you avoid sending duplicate messages during 5-minute interval
- Content based Deduplication ID or Producer provided Deduplication ID

FIFO Queue Concepts

Message Group ID

- Ordering is preserved within a message group
- Multiple message groups within a single FIFO Queue
- Only one consumer can have an inflight message in a message group
- Multiple consumers can access messages in different message groups – one consumer per message group
- Improve throughput and latency

Amazon Simple Notification Service

Fully managed Push Notification Service

Send individual messages or fan-out messages to large number of recipients

Send push notifications to mobile devices

Deliver messages to Amazon Simple Queue Service (SQS), AWS Lambda, HTTP(S) endpoint, Email

SNS Concepts

Topic – Logical Access Point and Communication Channel

Publisher – Sends message to a topic

Subscriber – Subscribes to a Topic using variety of supported protocols and receives messages

Figure - Components

SNS Usage Scenarios

Fanout – broadcast a message to multiple consumers

Application & System Alerts – Alert about changes to your application or infrastructure

Push Email & Text Messaging – Transmit messages to individuals or groups via email and SMS

Mobile App Push Notification – Notify directly to your App users

DynamoDB Core Concepts

Tables

- Items
- Attributes

Primary Key

- Partition Key
- Partition Key and Sort Key

Secondary Indexes

- Global Secondary Index
- Local Secondary Index

DynamoDB Core Concepts

DynamoDB Streams

- Captures data modification events in DynamoDB Tables
- Ordered Set of events
- Near real-time
- Lifetime of 24 hours
- Figure: Lambda functions to process stream events

Examples: New customer – welcome email, Add new product to ElastiCache or ElasticSearch

DynamoDB - Provisioned Throughput

- Consistent Low Latency Performance
- Read Capacity Units
- Write Capacity Units
- Modify any time
- Reduce cost – Purchase Reserved Capacity

Kinesis Platform

- Continuous capture, store, analyze
- Fully Managed
- Scales automatically – TBs per hour
- Capabilities
  - Kinesis Streams
  - Kinesis Firehose
  - Kinesis Analytics

  Figure: Pipeline - Clickstream Analytics

Kinesis Streams Concepts

- Stream is divided into Shards
- Data is stored in Shards
- One Shard provides: 1 MB/s WRITE, 2 MB/s READ, and up to 1,000 PUT operations
- Add or remove Shards dynamically depending on need

Kinesis Streams Concepts

- Data Record – unit of data stored in streams
  - Sequence Number
  - Partition Key
  - Data Blob (stored in Base64 encoding)
- Max Size per Data Record 1MB

Kinesis Streams Concepts

- Partition Key is used to route Data Record to different Shards.
- Partition Key is specified by the producer

Kinesis Streams Concepts

- Sequence Number is an unique identifier for every data record
- Assigned by Kinesis Streams
- Sequence number for a partition key generally increase over time

ECS Architecture

Different Components at Play:

Figure: Architecture

Figure: Scheduling

Figure: Container Agent

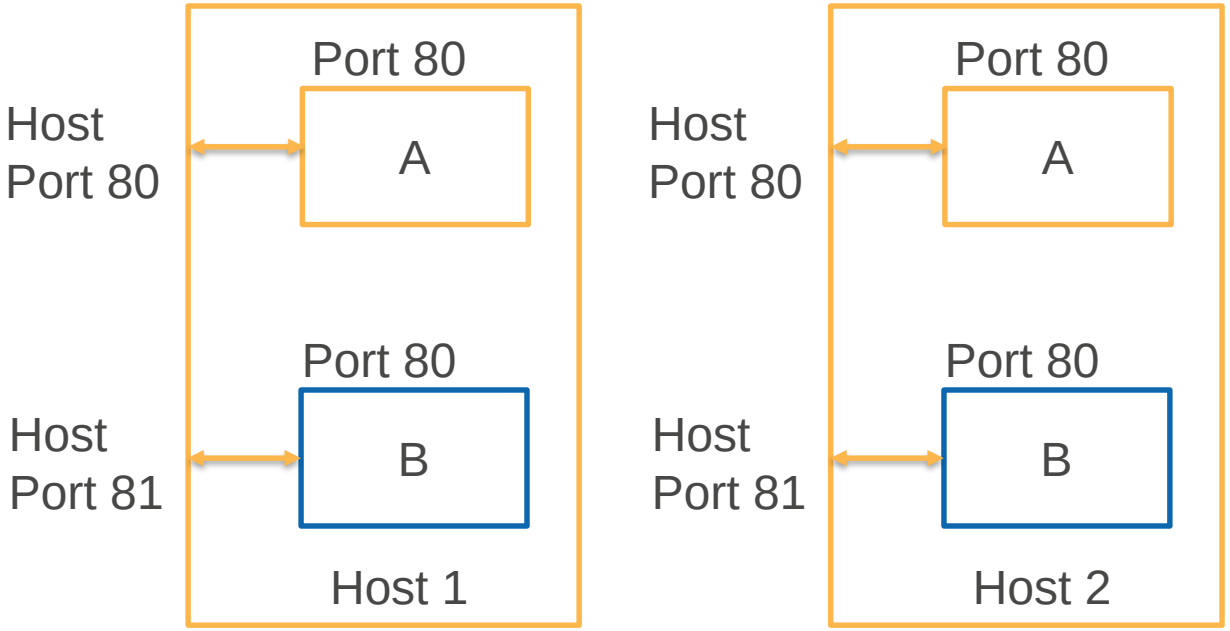
ECS Terminology

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

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 Static Port Mapping



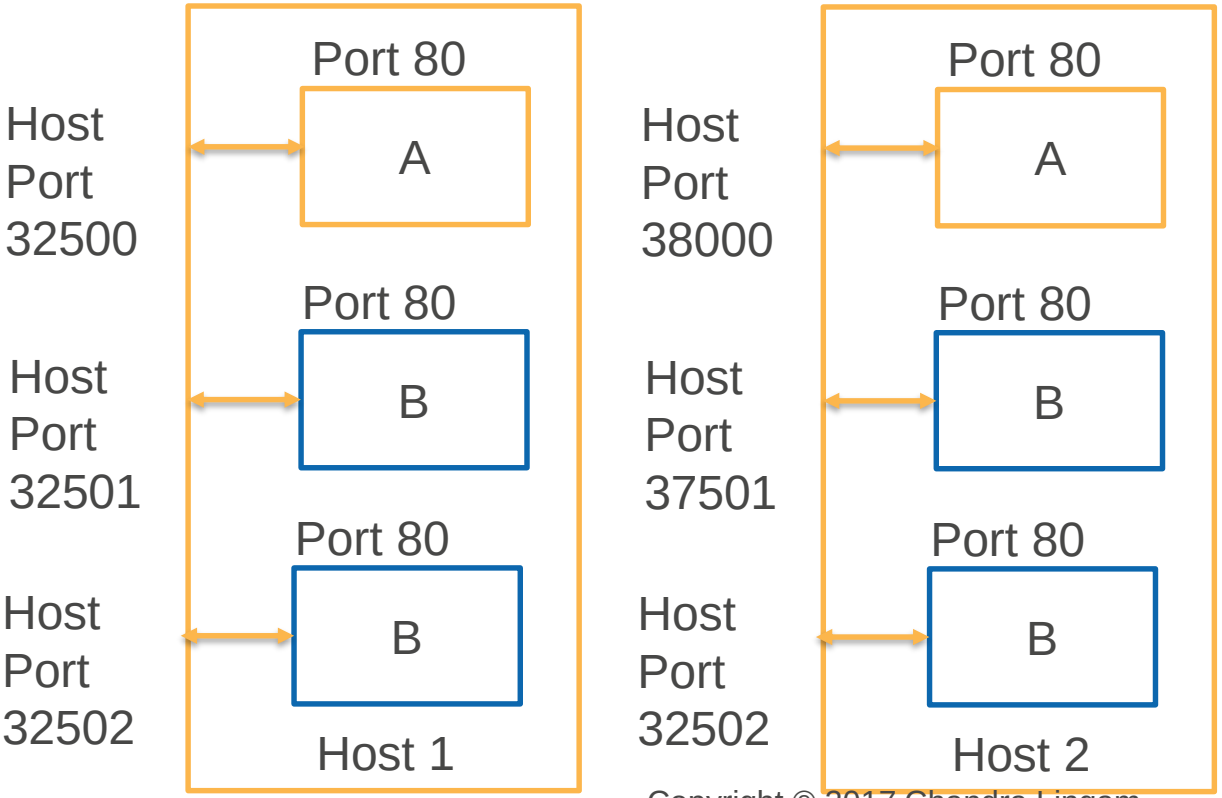
Host Port to Container Port mapping is hardcoded

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

ECS 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

ECS - 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

Comparison

- Elastic Beanstalk - Easy Solution for web apps and web services
- CloudFormation
  - Building-block Service that allows you to build and manage any AWS resource
  - Require you to author a template in JSON/YAML
  - Application deployment can be cumbersome
- OpsWorks
  - Powerful end-to-end solution. Scripting in Ruby
  - Complete application lifecycle from resource provisioning, configuration management, deployment, updates, monitoring, access control

Elastic Beanstalk Concepts

Concept	Description
Application	Logical Collection of Elastic Beanstalk components
Application Version	Labeled version of a deployable code
Environment	Resources provisioned to run a single application version
Environment Tier	Two types of environments: • Web server environment to handle http requests • Worker Environment to process SQS messages
Environment Configuration	Collection of parameters and settings to manage the resources
Configuration Template	Starting point for creating a new environment configuration

Elastic Beanstalk Workflow

Elastic Beanstalk Workflow

- Create Application
- Upload Code - Application Version
- Launch Environment
- Manage Environment

Elastic Beanstalk Permissions

- Elastic Beanstalk Service Role
  - Used for AWS resource management on your behalf
  - Monitoring resources
- Elastic Beanstalk Instance Profile – EC2 Instance IAM Role
  - Used by instance to log to S3
  - Upload Debug data to AWS X-Ray
  - …

Elastic Beanstalk Source Bundle

- Single Zip file
- Single WAR file
- Max size 512 MB
- Cannot contain a parent directory in the source bundle. Subdirectories are supported

Elastic Beanstalk Deployment Options

Deployment Option	Description
All at once	All instances are updated at the same time
Rolling	Updates are performed in batches. Old version and new version running in the environment until all instances are updated
Rolling with additional batch	Maintains full capacity by launching additional instances. When deployment completes, additional instances are terminated
Immutable	Full set of new instances for new version. Old instances are terminated after successful deployment

Elastic Beanstalk - Blue/Green Deployment

Eliminate downtime using Blue/Green Deployment

Blue - Production running old version

Green - New environment running new version

When Green deployment is successful, simply swap the CNAMEs of two environments using “Swap Environment URLs” option. Green now becomes the new Blue Production environment

Elastic Beanstalk Platform Updates

- AWS releases periodic updates to Elastic Beanstalk Platform
  - Software Component (AMIs, Tools, Elastic Beanstalk Scripts)
  - Configuration Component (Default settings applicable)
- Manual Update
- Managed Updates - automatically upgrades to latest version during scheduled maintenance window.
  - Only patches and minor version updates are supported
  - Major version changes are not automatically applied