SAA-C02

Course Scenario

- Animal Rescue & Awareness Org (IoT & Big Data)
- Global, with HQ in Brisbane - 100 staff
- Call center, Admin, IT, Marketing, Legal & Acct
- 100 Remote workers
- Major offices in London, New York & Seattle
- small on-prem datacenter in Brisbane
- Badly implemented AWS trial in SYD region
- few isolated Azure/GCP pilots
- Everyone uses Brisbane infrastructure
- Cost-conscious but progressive mgmt team

- Global Architecture

  On-prem → 192.168.10.0/24

  AWS Pilot → 10.0.0.0/16

  Azure " → 172.31.0.0/16

- Ideal Outcomes

  deploy into new regions quickly
  low cost & scalable
  agility
  fast performance
  Automation

Course Fundamentals & AWS Accounts

- AWS Accounts

  Isolation :-
  Login (Authentication)
  Service
  Security (Authorization)
  Billing
  Risk & Impact

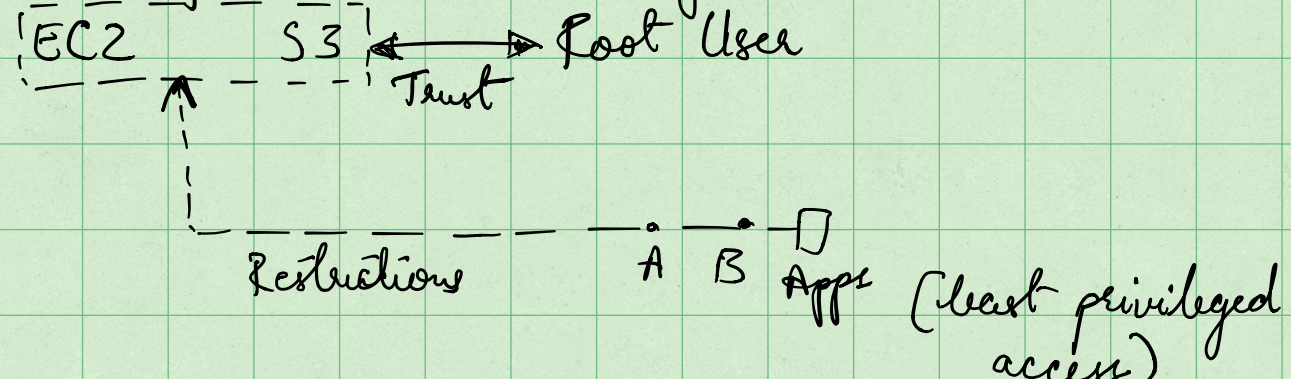
- Securing an AWS account

  email + pw for Account Root User
  always protect with MFA

- Controlling costs

  On-Demand billing
  Monthly
  Review costs anytime
  AWS Free Tier

- Identity and Access Management (IAM)



(least privileged access)

Users - humans or apps
Groups - collection of related users
Roles - AWS services / grant external access
Policy - Allow or Deny AWS services

ID Provider, Authenticates, Authorizes
No cost, global service (resilient), ALLOW or DENY,
no direct controls on external acc
Identity Federation and MFA

- IAM Access Keys
  longterm creds
  IAM user has 1 user & 1 password
  IAM user can have a maximum of 2 access keys
  Access keys can be created, deleted, deactivated, activated
  Two Parts to Access Key — [Access Key ID
  [Secret Access Key

Cloud Computing Fundamentals

- Five characteristics
  #1 On-demand Self-Service
  Can provision capabilities without requiring human interaction
  #2 Broad Network Access
  Capabilities are available over the network

& accessible through standard mechanisms

#3 Resource Pooling

There is a sense of location independence... no control or knowledge over the exact location of the resources

Resources are pooled to serve multiple consumers using a multi-tenant model

#4 Rapid Elasticity

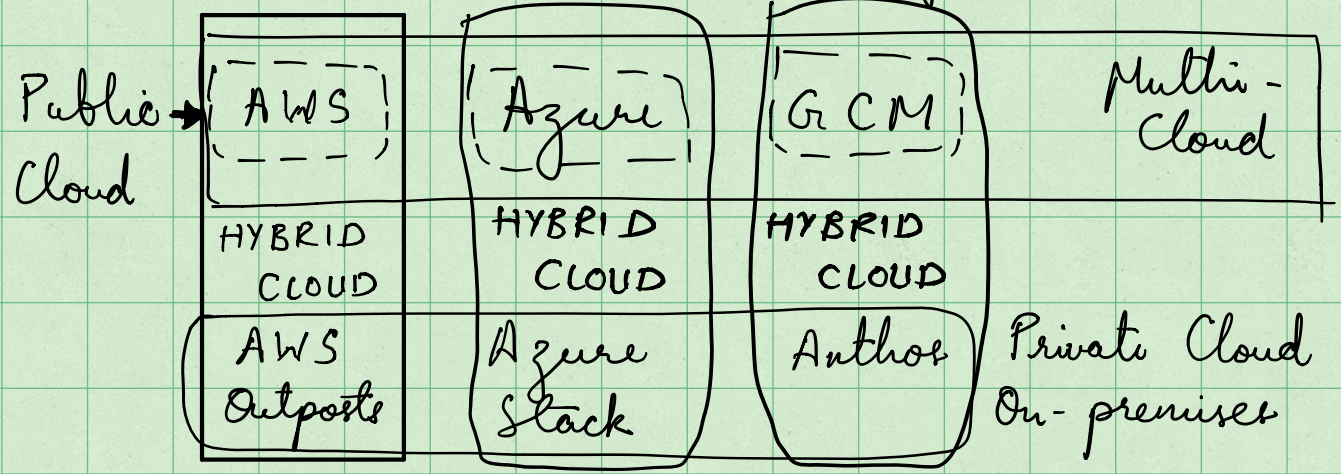
Capabilities can be elastically provisioned and released to scale rapidly outward and inward with demand.

To the consumer, the capabilities available for provisioning often appear to be unlimited.

#5 Measured Service

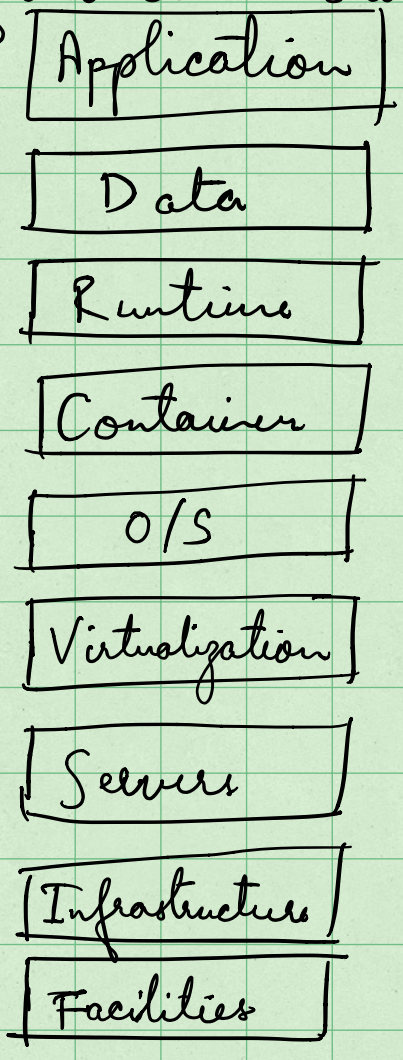
Resource usage can be monitored, controlled, reported... AND BILLED.

- Public vs Private vs Multi vs Hybrid



- Cloud Services Models

Infrastructure Stack



On-Premises vs DC Hosted

- Flexible
- datacenter hosting // facilities were rented

IaaS

- You can choose O/S and above
- Everything else is managed by the vendor
- lose flexibility
- EC2

PaaS

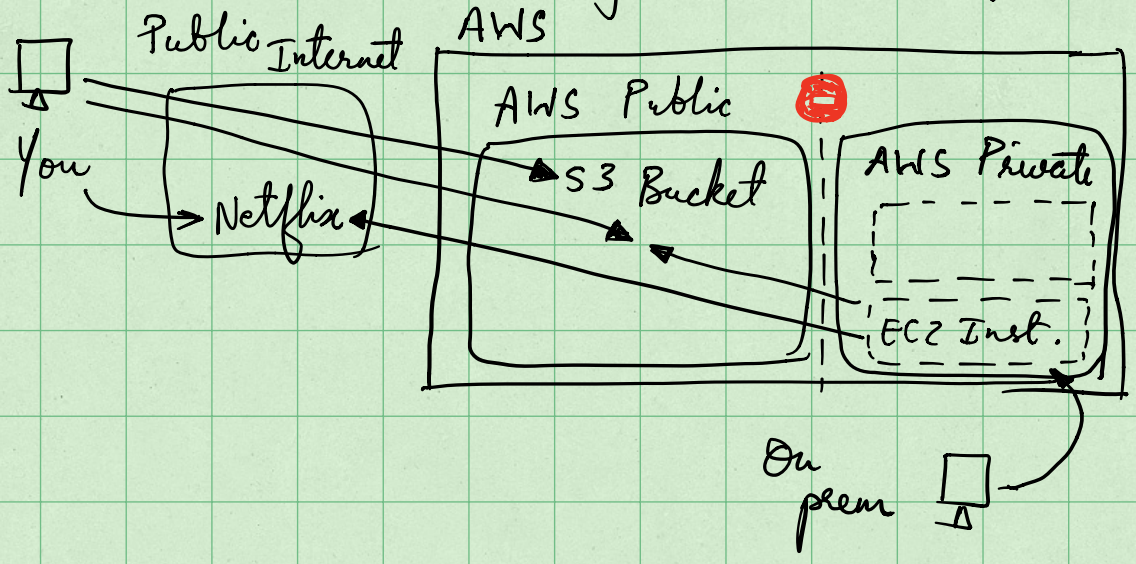
- Unit of consumption is Runtime & above
- Example is Heroku
- Mainly used by developers

SaaS

- Consume just a Application
- Eg: Netflix, Gmail, Dropbox

AWS Fundamentals

- Public vs Private Services
  - all about networking (connectivity to a service)



- AWS Global Infrastructure

AWS Regions

AWS Edge Locations

- Some services are global, some are region specific

- AWS Regions
  - Geographic Separation
    + Isolated Fault Domain
  - Geopolitical Separation
    + Different governance
  - Location Control
    + Performance

- Regions and AZs
  - Region Code // ap-southeast-2
  - Region Name // Asia Pacific (Sydney)

- Availability Zone
  ap-southeast-2a is physically isolated but connected using redundant high speed networking to ap-southeast-2b & ... 2c

- VPC - Virtual Private Cloud
  → way to create private network

- Service Resilience
  globally Resilient
  → service is globally available
  Eg IAM, Route 53
  Region Resilient

→ separate services in each region
replicate data in AZs

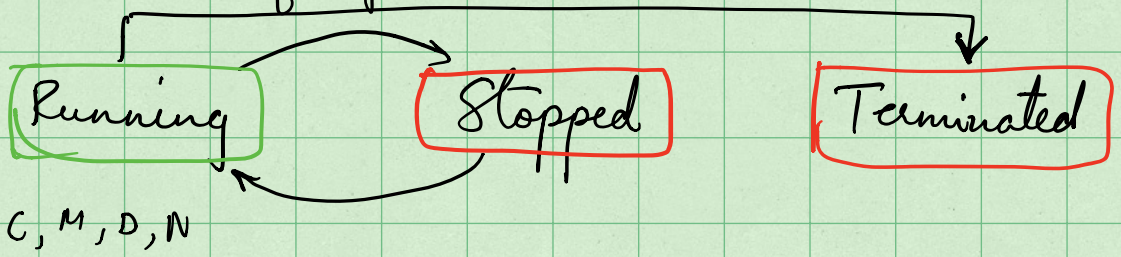
AZ Resilient
→ if single AZ fails, service fails

- Virtual Private Cloud (VPC) Basics
  - VPC = Virtual Network inside AWS
  - within 1 account & 1 region
  - Private & Isolated
  - Two types - Default VPC (only 1/region)
    - Custom VPC (many/region)

Default VPC
- strictly configured
- VPC CIDR 172.31.0.0/16
- one subnet for every AZ (/20)
- one per region (or zero)
- Internet Gateway (IGW), Security Group (SG) & NACL
- Subnets assign public IPv4 addresses

- Elastic Compute Cloud (EC2) Basics
  - IAAS - Provides VMs called Instances
  - Private service by-default → uses VPC networking

- AZ Resilient
- Different instance sizes & capabilities
- On demand billing - /second or / hr
- local on-host storage or EBS
- Instance Lifecycle



- Amazon Machine Image (AMI)

---AMI ⟶ EC2 ⟶ AMI

Permissions	Root Volume	Block Device Mapping
Public	drive that boots	links volumes to
Owner	the OS	device ID
Explicit		

- Connecting to EC2
  - Windows use RDP (Port 3389)
    Linux use SSH (Port 22)

- S3 Basics

- global storage platform - region based / resilient
- Public service, unlimited data & multi-user
- ... movies, audio, photos, large data sets
- Economical & accessed via UI / CLI / API / HTTP
- default storage service

- Objects & Buckets

- S3 Objects
  think of it like a file
  object → Key: koala.jpg → Bucket
  Value = Content being stored
  [0 bytes to 5 terabytes]
  version ID
  metadata
  Access Control
  Subresources

- S3 Buckets
  - created in specific region (stable)
    Blast radius = Region
  - Bucket name is globally unique [3-63 char, all lowercase, no underscores, start with letter or num]
  - can hold unlimited Objects
  - Flat structure
  - Buckets - 100 soft limit, 1000 hard per acc
  - Object: Key → name, Value → Data

- S3 Patterns and Anti-Patterns
  - object storage system [not file or block]

- You can't mount an S3 bucket
- large scale data storage, distribution or upload
- great for 'offload'
- INPUT and/or OUTPUT to many AWS products

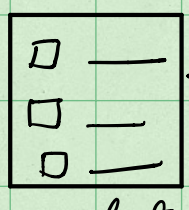
- Cloud Formation Basics

"create, updati & delete infrastructure using templates"

written in YAML or JSON

Template contents:

① list of resources [mandatory]
② Description // free text, restriction: must directly follow AWS Template Format Version
③ Metadata // controls how the UI presents the template
④ Parameters // add fields which prompt user
⑤ Mappings // Key/Value pairs, used for lookups
⑥ Conditions // decision making
⑦ Outputs // output from the template being applied



Template

→ All those other things

→ Resources

- CloudWatch Basics

"support service used by all other services"

- Collects and manages operational data
- Metrics - AWS Products, Apps, on-premise
- CloudWatch logs - " "
- CloudWatch Events - AWS Services & Schedules

4/12

- Shared Responsibility Model

- Responsible for security Customer IN OF AWS the cloud

- High Availability vs Fault Tolerance vs Disaster Recovery

High-Availability (HA)

- aims to ensure an agreed level of operational performance, usually uptime, for a higher than normal period

99.9% (Three 9's) = 8.77 hrs/yr downtime

99.999% (Five 9's) = 5.26 mins/yr downtime

Fault-Tolerance (FT)

- property that enables a system to continue operating properly in the event of the failure of some (one or more faults within) of its components.

Disaster Recovery (DR)

- a set of policies, tools and procedures to enable the recovery or continuation of vital technology infrastructure & systems following a natural or human-induced disaster

- Domain Name System (DNS)
  - DNS is a discovery service
  - translates machine into human & vice-versa
  - It's huge & has to be distributed
  - 4.3 billion IPv4 addresses

  - DNS Client ⇒ your laptop, phone, PC, etc
  - Resolver ⇒ software that queries DNS
  - Zone ⇒ part of the DNS database
  - Zonefile ⇒ physical database for a zone
  - Nameserver ⇒ where zonefiles are hosted

  - DNS Root → starting point of DNS

authoritative → think trusted

- DNS is a system of trust
  delegation ⇒ a root zone can delegate part of itself to another entity
- TLDs → general & country code

- Root Hints ⇒ config points to root servers
  Root Server ⇒ hosts the DNS root zones
  Root Zone ⇒ points at TLD authoritative servers
  gTLD ⇒ generic top level domain
  ccTLD ⇒ country-code top level domain

- Route 53 (R53) Fundamentals
  - global service, with single database
  1. Register domains
  2. Host Zone files on managed nameservers

  - globally resilient, FT, HA

Hosted Zones
- Zone files in AWS
- hosted on four managed name servers
- Can be public
- Or private. linked to VPC(s)
- stores records (recordsets)

- DNS Record Types

  Nameserver (NS)
  A and AAAA Records
  CNAME Records
  MX Records
  TXT Record

  TTL - Time To Live

IAM, Accounts And AWS Organisations

- IAM Identity Policies

  - type of policies that get attached to identities
  - grants or denies access
  - Statement has
    - a SID
    - Effect // Allow / Deny
    - Action
    - Resource
  - [1] Explicit DENY
    [2] Explicit ALLOW
    [3] Implicit DENY [Default]

// Explicit denies always take priorities

- Types of policies:
  - inline policies
  - managed policies

- IAM Users and ARNs

IAM Users are an identity used for anything requiring long-term AWS access eg Humans, Applications or service accounts

* If you can picture one thing, a named thing, the correct identity is IAM User.

- Principal → an entity tryying to access AWS
- Principal needs to be Authenticated & Authorized

- Amazon Resource Names uniquely identify resources within any AWS accounts
  - single or group of resources

- 5000 IAM users per accounts
- IAM User can be a member of 10 groups

- IAM Groups

IAM groups are containers for IAM Users

- cannot log into a group, no credentials
- IAM User can be part of multiple IAM groups
- Two main benefits:
  - effective administration style management of users
  - can have policies attached to them
- No all user [5000] group by default
- Groups are not a true identity

- IAM Roles - The Tech

IAM Roles are assumed ie. you become that role

IAM Roles — trust policy
           — Permissions policy

sts: AssumeRole

- When to use IAM Roles

  - Anything that's not an identity eg Lambda's runtime environment
  - Emergency case
  - Signup more than 5000 identities using

ID Federation
- Creating a high volume mobile app
- Cross account access

- AWS Organizations

master account can invite other std AWS accs to join an org

Organization root → root container
- it can contain AWS accs, or other containers (organizational units)

Consolidated Billing
- all billing to master account
- single monthly bill

Create new accounts inside the organisation

Best Practice
Single AWS account in the organisation which is used for logins

- Service Control Policies
JSON document

Can be attached to:

- org as whole (root container)
- one or more OUs
- one or more AWS accs

→ SCPs are inherited

→ master accs can't be restricted using SCPs

account permission boundaries

→ do not grant any permissions

Allow list or Deny list

default ↙

FullAWSAccess Policy

Deny list architecture

- Cloudwatch logs

public service - usable from AWS or on-prem

Store, Monitor & access logging data

↙

info data + timestamp

AWS Integrations

→ EC2, Flow Logs, Lambda, CloudTrail, S3 etc can generate metrics based on logs

metric filter

- CloudTrail

- logs API calls/activities as a CloudTrail Event
- 90 days stored by default in Event History for free
- To customize, create 1 or more Trails
- Management Events and Data Events

Imp Facts

- Enabled by default ... but 90 days only & no S3
- Trails are how you configure S3 & CW logs
- Management events only by default
- IAM, STS, CloudFront ⇒ global service events
- NOT REALTIME - There is a delay (~15 min)

Simple Storage Service (S3)

- S3 Security (Resource Policies & ACLs)

S3 is private by default, only accessible by root acc that created it

S3 Bucket Policies
- form of resource policy
- Resource perspective permissions
- ALLOW/DENY same or different accounts
- like identity policies, but attached to a bucket
- ALLOW/DENY Anonymous principals
- Explicit principal component

Access Control Lists (ACLs)
- ACLs on objects & bucket
- A subresource
- Legacy
- Inflexible & simple permissions

Five perms:
R | W | READ_ACP | WRITE_ACP | FULL_CONTROL

Block Public Access

What to choose & when?

- Identity : Controlling different resources
- Identity : You have a preference for IAM
- Identity : Same account
- Bucket : Just controlling S3
- Bucket : Anonymous or Cross-Account
- ACLs : NEVER - unless you must

- S3 Static Hosting

- Normal access is via AWS APIs
- This feature allows access via HTTP - eg Blogs
- Index & Error documents are set
- Website Endpoint is create
- Custom Domain via R53 - BUCKETNAME MATTERS

- Object Versioning and MFA Delete

- object versioning starts off in a disabled state
  - can't be disabled once enabled

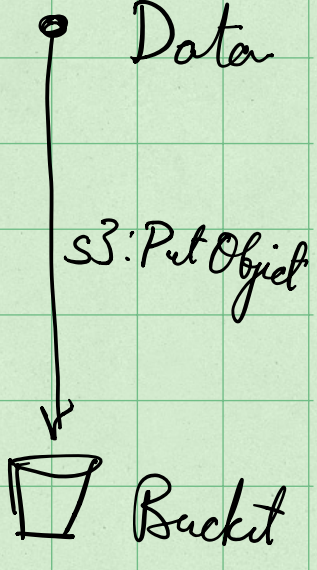
- stores multiple versions of objects

- cannot be switched off, only suspended
- Space is consumed by ALL versions

MFA Delete
- Enabled in versioning configuration
- MFA is required to change bucket versioning state
- MFA is required to delete versions
- Serial number (MFA) + Code passed with API calls

- S3 Performance Optimization

Single PUT Upload



- single data stream to S3
- stream fails - upload fails
- requires full restart
- Speed & reliability = limit of 1 stream
- Any upload up to 5 GB

Multipart Upload

- Data is broken up
- Min data size 100 MB for multipart
- 10,000 max parts, 5MB → 5GB
- Last part can be smaller than 5MB
- Parts can fail, & be restarted
- Transfer rate = speed of all parts

S3 Accelerated Transfer
nearest edge location using public internet then to destn edge location

- Encryption 101

→ Encryption Approaches

Encryption At Rest
eg. Laptop password

Encryption In Transit
eg. internet banking

→ Encryption Concepts
plaintext
algorithm
Key
Ciphertext

→ Symmetric Encryption
single key AES 256

local file encryption "at rest"
disk encryption

→ Asymmetric Encryption
public & private keys PGP, SSH, SSL

→ Signing
sign with your private key, encrypt with public

→ Steganography

art of hiding data in some other data

- Key Management Service (KMS)

- Regional & Public service
- create, store & manage keys
- Symmetric & Asymmetric Keys
- Cryptographic operations (encrypt, decrypt & ...
- Keys never leave KMS - Provides FIPS

140-2 (L2)

Customer Master Keys (CMK)

- CMK is logical - ID, date, policy, desc & state
- backed by physical key material
- generated or Imported
- CMKs can be used for upto 4KB of data

Data Encryption Keys (DEK)

- GenerateDataKey - works on > 4KB
- KMS doesn't store it
- Provides plaintext & ciphertext versions
- Encrypt data using plaintext key
- Discard plaintext key
- Store encrypted key with data

Key Concepts

- CMKs are isolated to a region & never leave
- AWS managed or Customer manage CMKs
- Customer managed keys are more configurable
- CMKs support rotation

AWS managed ←
- compulsory rotation
- every 1095 days

→ Cust managed
- rotation optional
- every year if

enabled

- Backing Key (and previous backing keys)
- Aliases

Key Policies & Security
- Key policy (Resource)
- Every CMK has one!
- Key policies + IAM Policies
- Role separation

- Object Encryption

Buckets aren't encrypted...

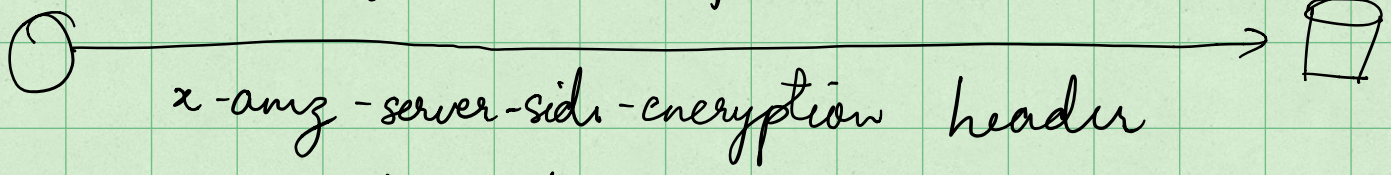
objects are..

- Client side encryption } at rest
- Server side encryption }

→ three types of server-side enc:

① SSE-C // customer managed keys
② SSE-S3 // amazon — " —
③ SSE-KMS // CMK stored in KMS

Bucket Default Encryption



aws:kms

- Object Storage Classes (S3)

- **S3 Standard**
  - default storage class
  - Objects are replicated to 3+ AZs
  - 99.999999999 % Durability
  - 99.99 % Availability
  - low latency, high throughput
  - No minimums or delays or penalties

- **S3 Standard-IA**
  "Infrequent Access"
  - less frequent but rapid access
  - Cheaper base rate vs S3 Standard (~54%)
  - 128KB minimum charge per object
  - 30 days minimum duration charge per object
  - per GB data retrieval fee
  - 99.9% Availability

- **S3 One Zone-IA**
  - All the same trade-offs as Standard-IA
  - 80% base cost of Standard-IA
  - Data stored in a single AZ - no 3+ AZ

replication

- Designed for 99.5% Availibility
- Can't withstand AZ failure

- Object Storage Classes (Glacier)

No immediate access to objects!

- S3 Glacier
  - Designed for cool Archival data
  - Cost is 17% of S3-Standard
  - 11 9's durability, 4 9's availability, 3+ AZ replication
  - 40KB minimum object capacity charge
  - 90 days min storage duration charge
  - Retrival in minutes or hours

- S3 Glacier Deep Archive
  - Designed for backups & as a tape-drive replacement
  - Cost is 4.3% of S3-Standard
  - 180 days minimum storage duration charge
  - Retrival within 12 hours
  - Can't make data public or download

manually

Use cases:

- Finance
  - → logs, audits, etc
- Health Care
  - → medical records,
- Media
  - → media & raw footage
- Physical Security - Cameras
- Scientific data sets

- Object Storage Classes (Intelligent Tiering)

"combination of S3 standard & std-IA"

→ automates movements between S3 std & Std-IA based on access monitoring

$0.0025 per 1000 objects

↑

monitoring fee

- Lifecycle Policies

- A lifecycle configuration is a

set of rules

- Rules consist of actions..
- .. on a Bucket or groups of objects
- Transition Actions
- Expiration Actions

- S3 Replication

two types

CRR → cross region

SRR → same region

Options:

- All objects or a subset
- Storage class - default is to maintain
- Ownership - default is the source acc
- Replication Time Control (RTC)
  - SLA that guarantees 15 min replication & monitoring

S3 Replication Considerations

→ Not retroactive & Versioning needs to be ON

→ One-way replication src to dest

→ Unencrypted, SSE-S3 & SSE-KMS (with extra config)

→ src bucket owner needs perm to objects
→ No system events, glacier or glacier deep archive
→ NO DELETES

Why use replication:
- SRR - log aggregation
- SRR - PROD & TEST Sync
- SRR - Resilience with strict sovereignty
- CRR - global resilience
- CRR - latency reduction

- S3 Pre Signed URLs

to give unauthenticated users access to objects in S3 for temp work

- You can create a URL for an object you have no access to
- When using the URL, the permissions match the identity which generated it
- Access denied could mean the generating ID never had access or doesn't have access now
- Don't generate with a role .. URL stops

working when temporary credentials expire..

- S3 Select and Glacier Select

"ways to retrieve parts of objects"

- S3 can store HUGE objects (upto 5 TB)
- You often want to retrieve the entire object
- Retrieving a 5 TB object.. takes time uses 5 TB
- Filtering at the client doesn't reduce this
- S3/glacier select - lets you use SQL like statements
- ... to select part of the object, prefiltered by S3
- CSV, JSON, Parquet, BZIP2 compression for CSV & JSON

Virtual Private Cloud (VPC) Basics

- ## Networking Refresher

Internet Protocol

IPv4 - RFC 791 (1981)
IPv6 - RFC 8200 (2017)

IPv4 Classful Addressing

IPs split into ranges by RIR (regional internet registry)

Class A -> ~2.1 billion 128 networks
Class B -> ~1 billion IPs
Class C
Class D
Class E

Internet / Private IPs

RFC 1918

+ 10.0.0.0 - 10.255.255.255 (A Single Class A)
+ 172.16.0.0 - 172.31.255.255 (16 Class B)
+ 192.168.0.0 - 192.168.255.255 (256 Class C)

Classless Inter Domain Routing (CIDR)

10.0.0.0/16

↓ break into two smaller network

10.0.0.0/17 ⇒ 10.0.128.0/17

0.0.0.0/0 means ALL IP addresses

10.0.0.0/8 means 10.ANYTHING - Class A - 16 mil IP

10.0.0.0/16 means 10.0.ANYTHING - Class B - 65,536 IPs

10.0.0.0/24 means 10.0.0.ANYTHING - Class C - 256 IPs

1.3.3.7/32 means 1 IP address

IP, TCP, UDP

IP Version 6

IPv4 Dotted-Decimal with octets

IPv6 has two octets called hextet every colon

- VPC Sizing and Structure

VPC Considerations

- What size should the VPC be

- Are there any Networks we can't use
- VPC's, Cloud, on prem, partners & vendors
- Try to predict the future
- VPC Structure - Tiers & Resiliency (Az) Zones

Best Practice

Avoid using default VPC IP range

- VPC minimum /28 (16 IP), maximum /16 (65456 IPs)

  Avoid common ranges [10.1 - 10.10]
- Reserve 2+ networks per region being used per account

VPC Sizing

→ How many subnets will you need?

→ How many IPs total? How many per subnet?

Proposal

huge global

10.16 → 10.127

/16 per VPC - 3 AZ (+1), 3 Tiers (+1) → 16 subnets

/16 split into 16 subnets = /20 per subnet (4091 IPs)

- Custom VPCs

- Regional Service - All AZs in the region
- Isolated network
- Nothing IN or OUT without explicit configuration
- Flexible configuration - simple or multi tier
- Hybrid networking
- Default or Dedicated Tenancy!
- IPv4 Private CIDR Blocks & Public IPs
- 1 Primary Private IPv4 CIDR Block
- min /28 (16 IP) max /16 (65,536 IP)
- Optional secondary IPv4 Blocks
- Optional single assigned IPv6 /56 CIDR block

DNS in a VPC

- Provided by R53
- VPC 'Base IP + 2' Address
- enable Dns Hostnames - gives instances DNS Names
- enable Dns Support - enables DNS resolution in VPC

VPC Subnets

- AZ Resilient
- A subnetwork of a VPC - within a particular AZ
- 1 Subnet => 1 AZ, 1 AZ => 0+ Subnets
- IPv4 CIDR is a subset of the VPC CIDR
- Cannot overlap with other subnets
- Optional IPv6 CIDR (/64 subset of the /56 VPC-space for 256)
- Subnets can interact with other subnets in the VPC

Subnet IP Addressing

- Reserved IP Addresses (5 in total)
  - Network Address
  - Network + 1 - VPC Router
  - Network + 2 - Reserved (DNS)
  - Network + 3 - Reserved Future Use
  - Broadcast Address - last IP

- VPC Routing, Internet Gateway & Bastion Hosts

  - Every VPC has a highly available VPC Router
  - In every subnet ... `network + 1` address
  - Routes traffic between subnets

- Controlled by 'route tables' each subnet has one
- A VPC has a Main route table
  - subnet default

Internet Gateway (IGW)
- Region resilient gateway attached to a VPC
- 1 VPC = 0 or 1 IGW, 1 GW = 0 or 1 VPC
- Runs from within AWS Public Zone
- gateways traffic btw VPC & Internet or AWS Public Zone
- Managed - AWS handles it

Bastion Host / Jumpbox
- both are same
- An instance in a public subnet
- Incoming management connections arrive there
- Then access internal VPC resources
- Only way into a VPC

- Network Access Control Lists (NACLs)
  like a firewall for subnets

2 set of rules
→ inbound
→ outbound

- Stateless — INITIATION & RESPONSE
- Only impacts data crossing subnet border
- Can EXPLICITLY ALLOW and DENY
- IPs/Networks, Ports & Protocols - no logical resources
- NACLs can't be assigned to resources, only subnets
- Use with SG's to add explicit DENY (Bad IPs)
- One subnet = One NACL at a time

- Security Groups (SG)

"operates at higher level"

Stateful

- Stateful - TRAFFIC & RESPONSE = Same Rule
- SGs can filter based on AWS logical resources, other SGs & even themselves
- Implicit deny & Explicit allow
- NO EXPLICIT DENY

SGs vs NACLs

- NACLs used for products that don't support

SGs eg. NAT gateways
- NACLs when adding explicit DENY
- SG as the default almost everywhere

- Network Address Translation (NAT) and NAT Gateway

"giving private resource outgoing only access"

- Network Address Translation
- A set of processes - remapping SRC or DST IPs
- IP masquerading - hiding CIDR Blocks behind one IP
- Private IPv4 addresses are running out
- gives private CIDR range outgoing internet access*

NAT gateways
- Runs from a public subnet
- Uses Elastic IPs (Static IPv4 Public)
- AZ Resilient Service (HA in that AZ)
- For region resilience - NATGW in each AZ...
- ...RT in for each AZ with that NATGW as target
- Managed, scales to 45 Gbps, $ Duration &

Data Volumes

IPv6

- NAT isn't required for IPv6
- All IPv6 addresses in AWS are publicly available
- The IG works with ALL IPv6 IPs directly
- NAT gateways don't work with IPv6
- ::/0 Route + IGW for bidirectional connectivity
- ::/0 Route + Egress-Only Internet Gateway - Outbound only

Elastic Compute Cloud (EC2) Basics

- Virtualization 101

"process of running more than one OS"

- Emulated Virtualization
- Paravirtualization
- Hardware Assisted Virtualization
- [SR-IOV] single route I/O virtualization

↑ Enhanced Networking

- EC2 Architecture & Resilience

- EC2 instances are virtual machines (OS + Resources)
- EC2 instances run on EC2 Hosts
- Shared Hosts or Dedicated Hosts
- Hosts = 1AZ - AZ Fails, Host Fails, Instances Fail

When to use EC2?

- Traditional OS + Application compute
- Long running compute
- Server style applications ...
- ... either burst or steady-state load
- Monolithic application stacks
- Migrated application workloads or Disaster Recovery

- EC2 Instance Types

- Raw CPU, Memory, Local Storage Capacity & Type
- Resource Ratios
- Storage and Data Network Bandwidth
- System Architecture / Vendor
- Additional Features and Capabilities

EC2 Categories

- General Purpose - Default
- Compute Optimized
- Memory Optimized
- Accelerated Computing
- Storage Optimized

- Storage Refresher

- Direct (local) attached Storage - Storage on the EC2 Host
- Network attached Storage - Volumes delivered over the network (EBS)
- Ephemeral Storage - Temporary Storage
- Persistent Storage - Permanent Storage
- Block Storage — Mountable. Bootable.
- File Storage — Mountable. NOT Bootable.
- Object Storage — NOT mountable. NOT Bootable.

Storage Performance

IO (block) Size × IOPS = Throughput

- Elastic Block Store (EBS) and EBS Volumes

- Allocates block storage (Volumes) to instances
- Volume = ONE AZ, but HA / Resilient in that AZ
- Different physical storage types available (SSD/HDD)
- Varying levels of performance (IOPS, Throughput)
- Billed as GB/month (some have IOPS billable component)

Volume Types

- general purpose SSD (gp2)
- Provisioned IOPS SSD (io1)

SSD

HDD

- Throughput optimized HDD (st1)
- Cold HDD (sc1)

Dominant Performance Attribute

Remember!

- Volumes created in an AZ, isolated in that AZ
- AZ fails - Volume impacted.. snapshots help
- Highly available and resilient in that

AZ

- Generally one volume <-> 1 instance (...but multi-attach)
- GB/month fee regardless of instance state
- EBS MAX 80k IOPS (Instance), 64k (Vol) (io 1)
- MAX 2375 MB/s (Instance), 1000 MiB/s (Vol) (io 1)

- Instance Store Volumes

- Block Storage Devices
- Physically connected to one EC2 host
- Instances on that host can access them
- Highest storage performance in AWS
- Included in instance price..
- ATTACHED AT LAUNCH

Performance

- D2 = 3.5 GBps read & 3.1 GBps write
- I3 = 16 GB/second of sequential throughput
- More IOPs & Throughput vs EBS

PowerUp!

- Local on EC2 Host

- Add at launch only
- Lost on instance move, resize or hardware failure
- High performance
- You pay for it anyway - included in instance price
- TEMPORARY

- Choosing between the EC2 Instance Store and EBS

When EBS?

- Highly available and Reliable storage
- Persist independently from the EC2 instance
- Clusters - Multi-Attach feature of io1
- Region Resilient Backups
- Require up to 64,000 IOPS and 1000 MiB/s per volume
- Require up to 80,000 IOPS and 2375 MB/s per volume

When Instance Store?

- Value - Included in instance cost
- More than 80,000 IOPS & 2,375 MB/s

- Temp Storage volumes
- Stateless services
- Rigid lifecycle link storage ↔ Instance

- Snapshots, Restore & Fast Snapshot Restore (FSR)

"efficient way to backup"

EBS Snapshots

- Snapshots are incremental volume copies to S3
- The first is a full copy of 'data' on the volume
- Future snaps are incremental
- Volumes can be created (restored) from snapshots
- Snapshots can copied to another region

EBS Snapshots / Volume Performance

- New EBS volume = full performance immediately
- Snaps restore lazily - fetched gradually
- Requested blocks are fetched immediately
- Force a read of all data immediately

- Fast Snapshot Restore (FSR) - Immediate restore
- upto 50 snaps per region. Set on the Snap & AZ

Snapshot Billing
- Gigabyte-month
- Used NOT allocated data
- Incremental

- EBS Encryption

- Accounts can be set to encrypt by default
  - default CMK
  - custom CMK
- Each volume uses 1 unique DEK
- Snapshots & future volumes use the same DEK
- Can't change a volume to NOT be encrypted
- OS isn't aware of the encryption... no performance loss

- Network Interfaces, Instance IPs and DNS

- Secondary ENI + MAC = Licensing
- Multi-homed (subnets) Management & Data
- Different Security Groups - multiple interfaces
- OS - DOESN'T see public IPv4
- IPv4 Public IPs are Dynamic .. Stop & Start = Change
- Public DNS = private IP in VPC, public IP everywhere else

- Amazon Machine Images (AMI)

  - AMI's can be used to launch EC2 instances
  - AWS or Community Provided
  - Marketplace (can include commercial software)
  - Regional ... unique ID eg. ami-0a887e401f65...
  - Permissions (Public, Your, Account, Specific ppl)
  - You can create an AMI from an EC2 instance you want to template

→ AMI = One Region, only works in that one region
→ AMI Baking ... creating an AMI from a configured instance + application
→ An AMI can't be edited .. launch instance, update configuration & make new AMI

→ Can be copied between regions (includes its snapshots
→ Remember permissions.. default = your account

- Instance Billing Models

- On Demand Instances
- Spot Instances
- Reserved Instances
- Dedicated Hosts

On Demand Instances
- Instances have an hourly rate
- Billed in seconds (60s minimum) or Hourly
- Default Pricing Model
- No long-term commitments or upfront payments
- New or uncertain application requirements
- Short-term, spiky, or unpredictable workloads which can't tolerate any disruption

Spot Instances
- Spot pricing offers up to 90% off vs On-Demand
- A spot price is set by EC2 based on

spare capacity

- You can set a maximum price you'll pay
- If spot price goes above yours - instances terminate
- Applications that have flexible start & end times
- Apps which only make sense at low cost
- Apps which can tolerate failure and continue later

Reserved Instances

- Up to 75% off vs On-demand - for a commitment
- 1 or 3 years, All upfront, partial upfront, No upfront
- Reserved in region, or AZ with capacity reservation
- Scheduled Reservations
- Known steady state usage
- Lowest cost for apps which can't handle disruption
- Need reserved capacity

- Instance Status Checks & Auto Recovery

'per instance checks
- System Status
- Instance Status

- Horizontal & Vertical Scaling

Vertical Scaling
- Each resize requires a reboot - disruption
- Larger instances often carry a $ premium
- There is an upper cap on performance - instance size
- No application modification required
- Works for ALL application - even Monolithe

Horizontal Scaling
- Sessions are everything
- Requires application support OR off-host sessions
- No disruptions when scaling
- No real limits to scaling
- Often less expensive - no large instance premium
- More granular

- Instance Metadata

- EC2 Service that provides data to instances
- Accessible inside ALL instances
- http://169.254.169.254/latest/meta-data/
- Environment
- Networking
- Authentication
- User-Data
- NOT AUTHENTICATED or ENCRYPTED

Containers & ECS

- Introduction to Containers

- Dockerfiles are used to build images
- Portable - self-contained, always run as expected
- Lightweight - Parent OS used, fs layers are shared
- Container only run the application & environment it needs
- Provides same isolation as VM's
- Ports are 'exposed' to the host & beyond

- Applications stacks can be multi-container

- ECS - Concepts

- Container Definition - Image & Ports
- Task Definition - Security (Task Role), Container(s), Resources
- Task Role - IAM Role which the TASK assumes
- Service - How many copies, HA, Restarts

- ECS - Cluster Modes

EC2 vs ECS (EC2) vs Fargate

- If you use containers → ECS
- Large workload - price conscious → EC2 Mode
- Large workload - overhead conscious → Fargate
- Small / Burst workloads - Fargate
- Batch / Periodic workloads - Fargate

Advanced EC2

- Bootstrapping EC2 using User Data

- Bootstrapping allows EC2 Build Automation
- User Data - Accessed via the metadata IP
- http://169.254.169.254/latest/user-data
- Anything in User Data is executed by the instance OS
- ONLY on launch
- EC2 doesn't interpret, the OS needs to understand the User Data

User Data Key Points

- It's opaque to EC2.. its just a block of data
- It's NOT secure
- User data is limited to 16 KB
- Can be modified when instance stopped
- But only executed once at launch

- Enhanced Bootstrapping with CFN-INIT

- cfn-init helper script - installed on EC2 OS

- Simple configuration management system
- Procedural (User Data) vs Desired State (cfn-init)
- Packages, Groups, Users, Sources, Files, Commands and Services
- Provided with directives via Metadata & AWS::CloudFormation::Init on a CFN resource

- EC2 Instance Roles & Profiles

- Credentials are inside meta-data
- iam/security-credentials/role-name
- Automatically rotated - Always valid
- Should always be used rather than adding access keys into instance
- CLI tools will use ROLE credentials automatically

- SSM Parameter Store

- Storage for configuration & secrets
- String, StringList & SecureString
- License codes, Database Strings, Full Configs

& Passwords
- Hierarchies & Versioning
- Plaintext and Ciphertext
- Public Parameters - Latest AMIs per region

- System and Application Logging on EC2

- CloudWatch is for metrics
- CloudWatch Logs is for logging
- Neither natively capture data inside an Instance
- CloudWatch Agent is required
- ... plus configuration & permissions

- EC2 Placement Groups

Cluster - Pack instances close together
Spread - Keep instances separated
Partition - groups of instances spread apart

Cluster Placement Groups
- Can't span AZs - ONE AZ ONLY
- Can span VPC peers - but impacts performance
- Requires a supported instance type

- Use the same type of instance (not mandatory)
- Launch at the same time (not mandatory)
- 10 Gbps single stream performance
- Use cases: Performance, fast speeds, low latency

Spread Placement Groups

- Provides infrastructure isolation, each INSTANCE runs from a different rack
- 7 instances per AZ (HARD LIMIT)
- Not supported for dedicated instances or Hosts
- Use case: small number of critical instances that need to be kept separated from each other

Partition Placement Groups

- 7 partitions per AZ
- Instances can be placed in a specific partition
- ... or auto placed
- Partition placement groups are not supported for Dedicated Hosts
- Great for HDFS, HBase & Cassandra

- Dedicated Hosts

- EC2 Host dedicated to you
- Specific family eg a1, c5, m5
- No instance charges .. you pay for the host
- On-demand & Reserved Options available
- Host hardware has physical sockets and cores

Limitations & Features

- AMI Limits - RHEL, SUSE Linux, and Windows AMIs aren't supported
- Amazon RDS instances are not supported.
- Placement groups are not supported for Dedicated Hosts.
- Hosts can be shared with other ORG Accounts using RAM (Resource Access Manager)

- Enhanced Networking & EBS Optimized

Enhanced Networking

- Uses SR-IOV - NIC is virtualization aware
- No charge - available on most EC2 types
- Higher I/O & Lower Host CPU Usage
- More Bandwidth
- Higher packets-per-second (PPS)
- Consistent lower latency

EBS Optimized

- EBS - Block storage over the network
- Historically network was shared .. data and EBS
- EBS Optimized means dedicated capacity for EBS
- Most instances support and have enabled by default
- Some older ones support, but enabling costs extra

Route 53 - Global DNS

- Route 53 Public Hosted Zones

- A R53 Hosted Zone is a DNS DB for a domain.
- Public = Hosted on R53 provided public DNS Servers
- Globally resilient (multiple DNS Servers)
- Created with domain registration via R53 - can be separately
- Host DNS Records (eg A, AAAA, MX, NS, TXT)
- Hosted Zones are what the DNS references - Authoritative for a domain

- Route 53 Health Checks

- Health check are seperate from, but are used by records
- Health checkers located globally
- 10s or 30s (default)
- TCP, HTTP/HTTPS, HTTP/HTTPS with String Matching
- Healthy or Unhealthy
- Endpoint, CloudWatch Alarm, Checks of Checks (cal)

- RS3 Routing Policies

  - Simple
  - Failover
  - Weighted
  - Latency-based
  - Geolocation
  - Multi-value

Relational Database Service

- Database Refresher

Relational (SQL) vs Non-Relational (NoSQL)

- Structured Query Language (SQL)
- Structure in & between tables of data - Rigid schema
- Relationships between tables
- NoSQL - Not one single thing
  → different models
- generally a much more relaxed schema
- Relationships handled differently

- Databases on EC2

Why an EC2?

- Access to to the DB instance OS
- Advanced DB Option ... (DBROOT)
- ... Vendor demands
- DB or DB Version, AWS don't provide
- Specific OS/DB Configuration AWS doesn't provide
- Architecture AWS don't provide (replication / resilience)
- Decision makers who 'just want it'

Why you shouldn't?

- Admin overhead - managing EC2 & DBHost
- Backup / DR Management
- EC2 is single AZ
- Features - some AWS DB products are toit
- EC2 is ON or OFF - no serverless, no easy scaling
- Replication - skills, setup time, monitoring & effectiveness
- Performance - AWS invest time into optimisation & adv. features

- Relational Database Service (RDS) Architecture

- Database-as-a-service (DBaaS)
- .. Database Server-as-a-service
- Manage Database Instance (1+ Databases)
- Multiple engines MySQL, Maria DB, PostgreSQL, Oracle, Microsoft SQL Server
- Amazon Aurora

- RDS High-Availability (Multi AZ)

- No free tier - Extra cost for standby replica
- Standby can't be directly used
- 60-120 seconds failover
- Same region only (other AZs in the VPC)
- Backups taken from Standby (removes performance impact)
- AZ Outage, Primary Failure, Manual Failover, Instance type change & software patching

- RDS Automatic Backup, RDS Snapshots and Restore

Recovery Point Objective

&

Recovery Time Objective

RDS Restore

- Creates a NEW RDS Instance - new endpoint address
- Snapshots = single point in time, creation time
- Automated = any 5 min point in time
- Backup is restored & transaction logs are 'replayed' to bring DB to desired point in time
- Restores aren't fast - Think about RTO

- RDS Read-Replicas

Performance Improvements

- 5x direct read-replicas per DB instance
- Each providing an additional instance of read performance
- Read-Replicas can have read-replicas - but lag starts to be a problem
- Global performance improvements

Availability Improvements

- Snapshots & Backups improve RPO

- RTO's are a problem
- RR's offer near zero RPO
- RR's can be promoted quickly - low RTO
- Failure only - watch for data corruption
- Read only - until promoted
- Global availability ... global resilience

- Aurora Architecture

- Aurora architecture is VERY different from RDS
- Use a "Cluster"
- A single primary instance + 0 or more replicas
- No local storage - uses cluster volume
- Faster provisioning & improved availability & performance

Aurora Storage Architecture

- All SSD Based - high IOPS low latency
- Storage is billed based on what's used
- High water mark - billed for the most used
- Storage which is freed up can be re-used
- Replicas can be added & removed without requiring storage provisioning

Costs

- No free-tier option
- Aurora doesn't support micro instances
- Beyond RDS single AZ (micro) Aurora offers better value
- Compute - hourly charge, per second, 10 minute minimum
- Storage - GB-Month consumed, IO cost per request
- 100% DB Size in backups are included

Aurora Restore, Clone & Backtrack

- Backups in Aurora work in the same way as RDS
- Restores create a new cluster
- Backtrack can be used which allow in-place rewinds to a previous point in time
- Fast clones makes a new db MUCH faster than copying all the data - copy-on-write

- Aurora Serverless

- Scalable - ACU - Aurora Capacity Units

- Aurora Serverless cluster has a MIN & MAX ACU
- Cluster adjusts based on load
- Can go to 0 and be paused
- Consumption billing per second basis
- Same resilience as Aurora (6 copies across AZs)

Use Cases

- Infrequently used applications
- New applications
- Variable workloads
- Unpredictable workloads
- Development and test databases
- Multi-tenant applications

- Aurora Global Database

- Cross-Region DR and BC
- Global Read Scaling - low latency performance improvements
- ~1s or less replication between regions
- No impact on DB performance
- Secondary regions can have 16 replicas

- Can be promoted to R/W
- Currently MAX 5 secondary regions ...

- Multi-master writes

- Default Aurora Mode is Single-Master
- One R/W & 0+ Read Only Replicas
- Cluster Endpoint is used to write, read endpoint is used for load balancing
- Failover takes time - replica has to be promoted to R/W
- In multi-Master mode all instances are R/W

- Database Migration Service (DMS)

- A managed database migration service
- Runs using a replication instance
- Source and Destination Endpoints point at
- Source & Target Databases
- One endpoint MUST be on AWS

Network Storage

- EFS Architecture

- EFS is an implementation of NFSv4
- EFS Filesystems can be mounted in Linux
- Shared between many EC2 instances
- Private service, via mount targets inside a VPC
- Can be accessed from on-premises - VPN or DX

EFS

- Linux Only
- General Purpose & Max I/O Performance Modes
- General Purpose = default for 99% of uses
- Bursting & Provisioned Throughput Modes
- Standard & Infrequent Access (IA) Classes
- Lifecycle Policies can be used with classes

HA & Scaling

- Load Balancing Fundamentals

- Clients connect to the Load Balancer
- ... specifically the listener of the DB
- The LB connects on your behalf to 1+ targets (servers)
- 2 connections ... listener & backend
- Client abstracted from individual servers
- Used for High-Availability, Fault-Tolerance & Scaling

- Application Load Balancing (ALB)

- ALB is a 'layer 7' LB
  - understands HTTP/S
- Scalable & highly-available
- Internet-Facing or Internal
- Listens on the outside -> Send to Target(s) (Groups)
- Hourly rate and LCU Rate (Capacity)

Exam Powerup

- Targets ⇒ Target Groups which are addressed via rules
- Rules are path based or host based
- Support EC2, ECS, EKS, Lambda, HTTPS, HTTP/2 and Websockets
- ALB can use SNI for multiple SSL Certs - host based rules
- Recommended vs CLB (Legacy)

- Launch Configuration and Templates

- Allow you to define the configuration of an EC2 instance in advance
- AMI, Instance Type, Storage & Key pair
- Networking & Security Groups
- Userdata & IAM Role
- Both are NOT editable - defined once. LT has versions.
- LT provide newer features - including T2/T3 Unlimited, Placement Groups, Capacity Reservations, Elastic Graphics

- Auto Scaling Groups

- Automatic Scaling & Self Healing for EC2

- Uses Launch Templates or Configurations
- Has a Minimum, Desired and Maximum Size (1:2:4)
- Provision or Terminate instances to keep Desired level (between Min/Max)
- Scaling Policies automate based on metrics

Scaling Policies

- Manual scaling - Manually adjust the desired capacity
- Scheduled scaling - Time based adjustment e.g. - Sales...
- Dynamic Scaling
  - Simple - "CPU above 50% + 1", "CPU below 50% - 1"
  - Stepped Scaling - Bigger +/- based on difference
  - Target tracking - Desired aggregati CPU = 40% ... ASG handle it
- Cooldown Periods

Key Points

- Autoscaling Groups are free
- Only the resources created are billed
- Use cooldowns to avoid rapid

scaling

- Think about more, smaller instances
  - granularity
- Use with ALBs for elasticity
  - abstraction
- ASG defines WHEN & WHERE,
  LT defines WHAT

- Network Load Balancing (NLB)

- NLB's are Layer-4 - only understand TCP and UDP
- Can't understand HTTP/S but are faster
  - ~100ms vs 400ms for ALBs
- Rapid scaling - millions of requests per second
- 1 interface w/ static IP per AZ, can use Elastic IPs ("whitelisting")
- Can do SSL Pass through
- Can load balance non HTTP/S applications
  - doesn't care about anything above TCP/UDP

- SSL Offload & Session Stickiness

"Bridging, Pass-through, Offload"

Serverless & Application Services

- Architecture Evolution

monolithic & tiered architecture

Using Queues

Event-driven

- No constant running or waiting for things
- Producers generate events when something happens
- ... clicks, errors, criteria met, uploads, actions
- Events are delivered to consumers
- ... actions are taken & the system returns to waiting
- Mature event-driven architecture only consumes resources while handling events

- AWS Lambda

- Function-as-a-Service (FaaS)
- Event-driven invocation (execution)
- Lambda function = piece of code in our language

- Lambda functions use a runtime
- Runs in a runtime environment
- You are billed only for the duration a function runs
- Key component of serverless architecture

Key considerations
- Currently - 15 minute execution limit
- New runtime environment every execution
  - no persistence
- Execution Role provides permissions
- Load data from other services (eg S3)
- Store data to other services (eg S3)
- (free tier) 1M free requests per month and 400,000 GB-seconds of compute time per month

- Cloud Watch Events and Event Bridge

- If X happens or at Y time, do Z
- Eventbridge is CloudWatch Events v2(*)
- A default Event bus for the account
- In Cloud Watch Events this is the only bus
  → implicit bus
- Event bridge can have additional event

busses

- Rules match incoming events (or schedules)
- Route the events to 1+ targets (eg. Lambda)

- API Gateways

"Application Programming Interface"

- API Gateway is a managed API Endpoint Service
- Create, Public, Monitor & Secure APIs as a service
- Billed based on number of API Calls, Data Transfer & additional performance features such as caching
- Can be used directly for serverless architecture
- Or during a architecture evolution

- Serverless Architecture

- Serverless isn't one single thing
- You manage few, if any servers - low overhead
- Applications are collections of small & specialised functions
- Stateless and Ephemeral environments -

duration billing!

- Event-driven - consumption only when being used
- FaaS is used where possible for compute functionality
- Managed services are used where possible

- Simple Notification Service

- Public AWS Service - network connectivity with Public Endpoint
- Coordinates the sending & delivery of messages
- Messages are <= 256 KB payloads
- SNS Topics are the base entity of SNS - permissions & configuration
- A Publisher sends messages to a TOPIC
- TOPICS have Subscribers which receive messages
- eg. HTTP(s), Email (JSON), SQS, Mobile push, SMS Messages & Lambda

SNS

- Delivery Status - (including HTTP, Lambda, SQS)

- Delivery Retries - Reliable Delivery
- HA and Scalable (Region)
- Server side encryption (SSE)
- Cross-account via TOPIC Policy

- Step Functions

Problems with Lambda
- Lambda is FaaS
- 15-minute max execution time
- Can be chained together
- Gets messy at scale
- Runtime environments are stateless

State Machines
- Serverless workflow
  - START → STATES → END
- States are THINGS which occur
- Maximum duration 1 year
- Standard Workflow and Express Workflow
- Started via API Gateway, IOT Rules, EventBridge, Lambda, etc
- Amazon States Language (ASL)
  - JSON Template

- IAM Role is used for permissions!

States
- SUCCEED & FAIL
- WAIT
- CHOICE
- PARALLEL
- MAP
- TASK (Lambda, Batch, DynamoDB, ECS SNS, SQS, Glue, SageMaker, EMR, Step Funcs)

- Simple Queue Service

- Public, Fully Managed, Highly-Available Queues - Standard or FIFO
- Messages upto 256 KB in size - link to large data
- Received messages are hidden (Visibility Timeout)
- then either reappear (retry) or are explicitly deleted
- Dead-letter queues can be used for problem messages
- ASGs can scale and Lambdas invoke based on queue length

SQS

- Standard = at-least-once, FIFO = exactly-once
- FIFO (Performance) 3,000 messages per second with batching or up to 300 messages per second without
- Billed based on 'requests'
- 1 requests = 1-10 messages up to 64 KB total
- Short (immediate) vs Long (waitTimeSeconds) Polling
- Encryption at rest (KMS) & in-transit
- Queue Policy

- Kinesis & Kinesis Firehose

Kinesis Concepts

- Kinesis is a scalable streaming service
- Producers send data into a kinesis stream
- Streams can scale from low to near infinite data rates
- Public service & highly available by design
- Streams store a 24-hour moving window of data

- Multiple consumers access data from that moving window

SQS vs Kinesis

- SQS 1 production group, 1 consumption group
- Decoupling and Asynchronous communications
- No persistence of messages, no window
- Kinesis designed for huge scale data ingestion
- and multiple consumers ... rolling window
- Data ingestion, Analytics, Monitoring, App Clicks

Global Content Delivery & Optimization

- CloudFront Architecture Basics

- CloudFront is a global object cache (CDN)
- Content is cached in locations close to customers
- Lower latency and higher throughput
- Load on the content server is decreased
- It can handle static & dynamic content

Origin → The source location of your content

Distribution → The 'configuration' unit of CF

Edge Location → local infrastructure which hosts a cache of your data

Regional Edge Cache → larger version of an edge location. Provides another level of caching.

- ACM

- HTTP - Simple and Insecure
- HTTPS - SSL/TSL layer of Encryption added to HTTP
- Data is encrypted in transit

- Certificates prove identity
- Signed by a trusted authority
- Create, renew and deploy certificates with ACM
- Supported AWS Services ONLY (eg CF & ALBs)
  → NOT EC2

\

- Securing CF and S3 using OAI

"Origin Access Identity"

- Global Accelerator

- Moves the AWS network closer to customers
- Connections enter at edge .. using anycast IPs
- Transit over AWS backbone to 1+ locations
- Can be used for NON HTTP/S (TCP/UDP).

** Difference from CloudFront **

Advanced VPC Networking

- VPC Flow Logs

- Capture packet Metadata .. NOT packet CONTENTS
- Applied to a VPC - All interfaces in that VPC
- Subnet - interfaces in that Subnet
- Interfaces directly
- VPC Flow Logs are NOT realtime
- Destination can be S3 or CloudWatch Logs

- Egress-Only Internet Gateway

- With IPv4 addresses are private or public
- NAT allows private IPs to access public networks
- ... without allowing esternally initiated connections (IP)
- With IPv6 all IPs are public
- Internet Gateway (IPv6) allows all IPs IN and OUT

- Egress-Only is outbound-only for IPv6

- VPC Endpoints (Gateway)

Gateway Endpoints

- Provide private access to S3 and DynamoDB
- Prefix List added to route table ⇒ Gateway Endpoint
- Highly available (HA) across all AZs in a region by default
- Endpoint policy is used to control what it can access
- Regional, can't access cross-region services
- Prevent Leaky Buckets - S3 Buckets can be set to private only by allowing access ONLY from a gateway endpoint

- VPC Endpoints (Interface)

Interface Endpoints

- Provide private access to AWS public services
- ... anything NOT S3 and DDB

- Added to specific subnets - an ENI
  - not HA
- For HA, add one endpoint, to one subnet, per AZ used in the VPC
- Network access controlled via Security Groups
- Endpoint Policies - restrict what can be done with the endpoint
- TCP and IPv4 only
- Uses PrivateLink

- Endpoint provides a NEW service endpoint DNS
- e.g. vpce-123-xyz.sns.us-east-1.vpce.amazonaws.com
- Endpoint Regional DNS
- Endpoint Zonal DNS
- Applications can optionally use these or..
- Private DNS overrides the default DNS for services

- VPC Peering

- Direct encrypted network link between two VPCs
- Works same / cross-region and same /

cross-account

- (optional) Public Hostnames resolve to private IPs
- Same region SG's can reference peer SGs
- VPC Peering does NOT support transitive peering
- Routing Configuration is needed, SGs & NACLs can filter

Hybrid Environments and Migration

- AWS Site-to-Site VPN

- A logical connection between a VPC & on-premises network encrypted using IPSec, running over the public internet
- Full HA - if you design & implement it correctly
- Quick to provision, less than an hour
- Virtual Private Gateway (VGW)
- Customer Gateway (CGW)
- VPN Connection between the VGW & CGW

VPN Considerations

- Speed limitations - 1.25 Gbps
- Latency Considerations - inconsistent, public internet
- Cost - AWS hourly cost, GB out cost, data cap (on premises)
- Speed of setup - hours .. all software configuration
- Can be used as a backup for Direct Connect (DX)

- Can be used with Direct Connect (DX)

- Direct Connect

- A 1 Gbps or 10 Gbps Network Port into AWS
- at a DX Location (1000-Base-LX or 10 GBASE-LR)
- to your Customer Router (requires VLANS / BGP)
- or Partner Router (if extending to your location)
- Multiple Virtual Interfaces (VIFS) over one DX
- Private VIF (VPC) & Public VIF (Public Zone Services)

Considerations

- Takes MUCH longer to provision vs VPN
- DX Port provisioning is quick, the cross-connect take longer
- extension to premises can takes weeks/months
- Use VPN first, then replace with DX (Or leave as backup)
- Faster ~ 40 Gbps with Aggregation

- Low consistent latency, doesn't use business bandwidth
- NO ENCRYPTION

- Transit Gateway

  - Network Transit Hub to connect VPCs to on premise networks
  - Significantly reduces network complexity
  - Single network object - HA and Scalable
  - Attachments to other network types
  - VPC, Site-to-Site VPN & Direct Gateway

Considerations
- Supports transitive routing
- Can be used to create global networks
- Share between accounts using AWS RAM
- Peer with different regions, same or cross account
- Less complexity vs w/o TGW

- Storage Gateway

  - Hybrid Storage Virtual Appliance

(On-premises*)

- Extension of File & Volume Storage into AWS
- Volume storage backups into AWS
- Tape Backups into AWS
- Migration of existing infrastructure into AWS

- Tape Gateway (VTL) Mode
  - Virtual tapes ⇒ S3 & Glacier
- File - SMB and NFS
  - File storage backed up S3 Objects
- Volume Mode (Gateway Cached/Stored) - iSCSI
  - Block storage backed by S3 & EBS Snapshots

- Snowball / Edge / Snowmobile

- Move large amounts of data IN of OUT AWS
- Physical storage, suitcase or truck
- Ordered from AWS Empty, Load up & Return
- Ordered from AWS with data, empty & Return

Snowball

- Ordered from AWS, log a Job, Device Delivered (not instant)
- Data Encryption uses KMS
- 50TB or 80TB Capacity
- 1 Gbps (RJ45 1GBase-TX) or 10 Gbps (LR/SR) Network
- 10TB to 10PB economical range (multiple devices)
- Multiple device to multiple premises
- Only storage

Snowball Edge

- Both Storage & Compute
- Larger capacity vs Snowball
- 10 Gbps (RJ45), 10/25 (SFP), 45/50/100 Gbps (QSFP+)
- Storage optimized (with EC2)
  - 80TB, 24 vCPU, 32 GiB RAM, 1TB SSD
- Compute Optimized
  - 100TB + 7.68 NVME, 52 vCPU & 208 GiB RAM
- Compute with GPU
  - as above with a GPU
- Ideal for remote sites or where data processing or ingestion is needed

Snowmobile

- Portable DC within a shipping container on a truck
- Special order
- Ideal for a single location when 10PB+ is required
- Upto 100 PB per snowmobile
- Not economical for multi-site (unless huge) or sub 10PB

- Directory Service

Directory ?

- Stores objects with a structure
- Multiple trees → forest
- Commonly used in Windows Envs
- Sign-in to multiple devices with same user/pass for centralised management
- Microsoft Active Directory Domain Service (AD DS)
- Alternatively, SAMBA

Directory Service

- AWS Managed implementation
- Runs within a VPC

- HA by deploying into multiple AZs
- Some services NEED a directory
  eg: Amazon Workspaces
- Can be isolated
- or integrated with on-premises system
- or act as 'proxy' back to on-premises

Picking between modes

- Simple AD - The default. Simple reqs, A directory in AWS
- Microsoft AD - Applications in AWS which need MS ADS, or you need to TRUST AD DS
- AD Connector - Use AWS Services which need a directory without storing any directory info in the cloud, proxy to your on-premises Directory

- DataSync

- Data transfer service TO & FROM AWS
- Migrations, Data Processing Transfers, Archival/Cost Effective Storage or DR/BC
- designed to work at huge scale
- Keeps metadata (eg permissions/timestamps)

- Built in data validation

Features

- Scalable - 10 Gbps per agent (~ 100 TB per day)
- Bandwidth limiters (avoid link saturation)
- Incremental & scheduled transfer options
- Compression & encryption
- Automatic recovery from transit errors
- AWS Service Integration - S3, EFS, FSx
- Pay as you use per GB cost for data moved

Components

- Task
- Agent
- Location

- FSx for Windows Server

- Fully managed native windows file shares
- Designed for integration with win envs
- Integrates with Directory Service or Self-Manage AD

- Single or Multi AZ within a VPC
- On-demand and Scheduled Backups
- Accessible using VPC, Peering, VPN, Direct Connect

Key Features

- VSS - User driven restores
- Native file system accessible over SMB
- Windows permission model
- Supports DFS scale-out file share structures
- Managed - no file server admin
- Integrates with DS AND your own directory

Security, Deployment & Operations

- AWS Secrets Manager

- It shares functionality with parameter store
- Designed for secrets
- Usable via Console, CLI, API or SDKs
- Supports automatic rotation, this uses lambda
- Directly integrates with some products (RDS)

- AWS WAF & Shield

Shield

- Provides AWS resources with DDoS protection
- Shield Standard - free with R53 and CloudFront
- Protection against Layer 3 and Layer 4 DDoS attacks
- Shield Advanced - $3000 / month
- EC2, ELB, CloudFront, Global Accelerator and R53
- DDoS Response Team & Financial Insurance

Web Application Firewall

- Layer 7 (HTTPS) Firewall
- Protects against coupled layer 7 attacks / exploits
- SQL Injection, XSS, geo blocks, rate awareness
- Web access control list (WEBACL) integrated with ALB, API Gateway & CloudFront
- Rules are added to a WEBACL & evaluated when traffic arrives

- Cloud HSM

- KMS is a shared service & managed by AWS
- True "Single Tenant" Hardware Security Module
- AWS provisioned, fully customer managed
- Fully FIPS 140-2 Level 3 (KMS is L2 Overall, some L3)
- Industry std APIs - PKCS#11, JCE, CryptoNG
- KMS can use Cloud HSM as a custom key store

When to use?

- No native AWS integration eg no S3 SSE
- Offload the SSL/TSL Processing for Web Servers
- Enable Transparent Data Encryption (TDE) for Oracle Databases
- Protect the Private Keys for an Issuing Certificate Authority (CA)

No SQL Databases & Dynamo DB

- Dynamo DB
  - No SQL Public DB aaS
    ⇒ Key/Value & Document
  - No self-managed servers or infrastructure
  - Manual/Provisioned provisioned performance IN/OUT or On-Demand
  - Highly resilient across AZs & global (opt.)
  - Really fast, single-digit ms (SSD based)
  - Backups, point-in-time recovery, enc at rest
  - Event-driven integration, do things when data changes

Considerations
- NoSQL, preference DynamoDB
- Relational Data ~ NOT DynamoDB
- Key/Value ~ preference DynamoDB
- Access via console, CLI, API ~ NO SQL
- Billed based RCU, WCU, Storage & features

- DynamoDB - Operations, Consistency and Performance

Reading & Writing

- On-Demand $ price per million R or W
- Provisioned
- Every operation 1 RCU / WCU atleast
- 1 RCU is 1 x 4 KB read operation per second
- 1 WCU is 1 x 1 KB write " "
- Every table has a RCU & WCU burst pool (500 seconds)

"Query & Scan"

WCU Calculation

- store 10 items 2.5 K

$$\text{WCU per item} = \text{ROUND UP}\left[\frac{\text{ITEM SIZE}}{1\text{KB}}\right] \quad [3]$$

↓

Multiply by average number of writes per second

⇒ WCU Required is 30

RCU Calculation

Read 10 items 2.5 K

$$\text{RCU per item} = \text{ROUND UP}\left[\frac{\text{ITEM SIZE}}{4\text{KB}}\right] \quad [1]$$

↓

Multiply by avg read ops per second [10]

⇒ Strongly Consistent = 10 RCU needed

⇒ Eventually Consistent

$= \frac{10}{2} = 5$ RCU

- DynamoDB - Streams & Lambda Triggers

Streams

- Time ordered list of ITEM CHANGES
- 24-Hour rolling window
- Enabled on a per table basis
- Records INSERTS, UPDATES & DELETES
- Different view types influence what's in the stream

Trigger Concepts

- ITEM changes generate an event
- That event contains the data which changed
- An action is taken using that data
- AWS = Streams + Lambda
- Reporting & Analytics

- Aggregation, Messaging & Notifications

- DynamoDB Local & Global Secondary Indexes

Indexes
- Query is most efficient but can only work on 1 PK at a time
- & optionally a single or range of SK values
- Indexes are alternative views
- diff SK (LSI) or diff PK & SK (GSI)
- Some or all attributes (projection)

Local Secondary Indexes (LSI)
- alt view
- MUST be created with table
- 5 LSI's per base table
- Alternative SK on the table
- Shares the RCU & WCU
- Attributes - ALL, KEYS_ONLY & INCLUDE

Global Secondary Indexes (GSI)
- Can be created any time
- Default limit of 20 per base table
- Alt PK and SK

- GSI's have their own RCU & WCU
- Attributes - ALL, KEYS_ONLY & INCLUDE

Considerations
- Careful with projection
- Queries on attributes NOT projected are expensive
- Use GSIs as default, LSI only when strong consistency is required
- Use indexes for alt access patterns

- DynamoDB - Global Tables

- provide multi-master cross-region replication
- Last writer wins is used for CR
- Reads & Writes occur in any region with sub-second replication
- Strongly consistent reads ONLY in the same region as writes

- DynamoDB - Accelerator (DAX)

"in-mem cache"

DAX Considerations

- Primary NODE (Writes) & Replicas (Read)
- Nodes are HA, Primary failure = election
- In-Memory cache - Scaling ~ Much faster reads, reduced costs
- Scale UP and Scale OUT (Bigger or more)
- Supports write-through
- DAX deployed WITHIN a VPC

- Amazon Athena

- Serverless Interactive Querying Service
- Ad-hoc queries on data - pay only data consumed
- Schema-on-read - table-like translation
- Original data never changed - remains on S3
- Schema translates data => relational like when read
- Output can be sent to other services

EXAM

- Exam Techniques

- 130 Minutes as standard
- 65 Questions = 2 Minutes Per Questions
- 720/1000 Pass Mark
- Multi choice or Multi Select

P1 → Go through all and answer easy first
P2 → Skip hard questions and answer medium questions
P3 → Answer the hard questions

- Be efficient
- 2 mins → read, answer & make a decision
- Don't guess until the end

Question Techniques
- 1-2 lines of scenario ← don't matter
- Then the actual question
- 4-5 Answers (multi choice or multi select)
- Mostly answer is right or wrong
- Occasionally 'most suitable' answer
- 1 or 2 answers can be eliminated

- Most questions have overall criteria or restriction:
  - Cost Effective
  - Best Practice Security
  - Highest Performance
  - TimeFrame
- Eliminate crazy answers