IAM

IAM allows you to manage users and the level of access to the AWS console

Offers:

- Shared access to your AWS account.
- Granular Permissions
- Identity Federation Services (AD, Facebook, LinkedIn, etc.)
- MFA as a feature.
- Temporary access for users/devices/services where necessary.
- Allows you to define a password rotation policy.
- PCI DSS Compliant

Key Terms:

- Users - End Users such as people, or employees of an organization etc.
- Groups - A collection of users. Each member within a group inherits permissions.
- Policies - Made up of documents. These documents are stored in JSON, they give permissions as to what an IAM User/Group/Role is able to do.
- Roles - You create roles and assign them to AWS resources.

Don't let users, or anyone for that matter access your AWS root account. You should enable MFA on this account in order to reduce risk.

IAM is always on a global basis, not by region. Same with roles and groups within your AWS account.

When creating an IAM user you define one of the following types of accounts:

- Programmatic Access: (API, CLI, SDK, etc.)
- AWS Management Access: (For accessing the AWS web console)

You can only get the secret access key once and it's upon creation of the user.

New users have no permissions when first created.

Password Policies allow for the following options:

- Require at least one uppercase letter
- Require at least one lowercase letter
- Require at least one number
- Require at least one non-alphanumeric character

- Allow users to change their own password
- Enable password expiration (in days)
- Prevent password reuse (number of passwords to remember)
- Password expiration requires admin reset

Access Key ID & Secret Access Keys are only for the AWS APIs and CLI. They're not the same as a password.

Creating A Billing Alarm

Creates alarm on a per region basis.

1. Open the billing dashboard and go to billing preferences.
2. Open CloudWatch from services and click billing
3. Create new alarm and set your quota and an email address to receive the alert
4. Confirm the email address via confirmation code from AWS

S3

S3 provides simple storage service. It's a highly scalable object (file) based storage system.
Allows you to upload anything from 0 bytes to 5TB.
A bucket is essentially a folder. The name for the bucket must be unique globally. This is due to the fact that a URL is assigned upon creation.
When you do upload to S3 you'll get an HTTP 200 response upon completion, meaning it was successful.

Filter access to objects in S3 using ACLs for S3 and Bucket Policies
Identified as a durable key-value store
Multipart uploads are preferred: They provide the ability to pause and resume object uploads, while delivering improved throughput. This inturn offers quick recovery from network related issues

Objects consist of the following:

- Key - Name of the object.
- Value - The data that is made up of the sequence of bytes.
- Version ID - If you use the same name the previous version is kept.
- Metadata - Data about data you're storing. Allows for better organization.
- Subresources - ACLs and Torrent

How does data consistency work with S3?

Read after write consistency for PUTS of new objects. - If you write a NEW FILE and read it immediately afterwards, you will be able to view the data.

Eventual Consistency for overwrite PUTS and DELETES. - If you update AN EXISTING FILE or delete a file and read it immediately, you may get the older version or you may not. Bassically changes to objects can take a little bit of time to propagate.

Amazon S3 Guarantees:

99.99% availability for the S3 platform itself.
11 9s for durability across all s3 storage class types. (Won't lose data)

S3 Features:

Tiered Storage Available
Lifecycle Management - (File older than X days more to this tier)
Versions
Encryption using AES-256
MFA Delete for objects
Secure your data using ACLs and Bucket Policies

S3 Storage Classes:

- Standard - Designed to sustain the loss of 2 data centers concurrently.
- IA (Infrequently Accessed) - For data used infrequently but with some redundancy.
- One Zone - IA - Stored in one avail zone and low cost.
- Intelligent Tiering - Uses ML and moves objects based off how often they're accessed.
- Glacier - Data archiving - Often used for FED or Compliance regulation.
- Glacier Deep Archive - Lowest cost, but data takes 12 hours to retrieve.

Billing on S3:

Charged for amount of Storage
Charged for # of requests
Storage Management Pricing
Data Transfer Pricing
Transfer Acceleration - Enables fast and secure transfers to end users and distributed via an edge location. Helps improve upload times for users.
Cross Region Replication Pricing - If you have a bucket and replicate to another region you'll be charged. This is often for high availability.

It’s important to note you can change storage class on both an object and a bucket level.

By default, all new buckets are PRIVATE. You can setup access controls using Bucket Policies and ACLs.

S3 buckets can be configured to create access logs which log requests made to the bucket. This can be setup to use another bucket within another account.

Encryption In Transit:

SSL/TLS - HTTPS

Encryption at REST:

Server Side-

- S3 Managed Keys - SSE-S3
- AWS KMS - Managed Keys - SSE-KMS
- Server Side Encryption With Customer Provided Keys - SSE-C

Client Side-

- Encrypt yourself before uploading to S3.

You can change encryption on an object or at the bucket level.

Versioning:

- Great for backup
- Once enabled, cannot be disabled. Can be suspended.
- Integrates with Lifecycle rules
- MFA Delete capability
- (Delete Markers are created upon deleting an object) If you delete the delete marker the file is restored to the latest version.

Lifecycle Rules:

Transitions can apply to current/previous versions or both. These are defined to move between storage classes after X amount of days. You can have multiple levels of movement within the same rule.

Expiration allows you to delete the object after X days of creation. You can also delete incomplete multipart uploads after X days here.

These rules do not require versioning, but can be used with.

Cross Region Replication:

- Done in Management -> Replication
- Requires versioning to be enabled on the bucket level for source and destination.
- Requires Destination of another bucket within the same of a different AWS account
- Has to be in a different region. Can't be the existing one you're using.
- Doesn't move existing files automatically to the destination.

Transfer Acceleration:

Utilizes the edge network to accelerate uploads. Provides a URL to upload directly to.
Test tool shows you how much faster the upload would be per edge location.

CloudFront

A CDN that delivers web pages and other types of content to a user based on the geographic location of the user, the origin of the webpage, and content delivery server.

Delivers content with the best possible performance per location.

Objects are cached for the duration of the TTL specified.

You can clear the cached content, but will be charged.

Type of distribution:

- Web - Typically for Websites
- RTMP - For media streaming

Key Terms:

- Edge Location - This is a location that the content is cached.
- Origin - This is the location where original content is stored. This can be an S3 bucket, EC2 instance, ELB or Route53.
- Distribution - The name given to the CDN which consists of a collection of Edge Locations.

Creating a CloudFront Distribution:

- Origin Domain Name - Name of S3 Bucket for example.
- Origin Path - Can specify bucket path if you wish.

- Origin ID - Use default.
- Restrict Bucket Access - Yes or No
- Viewer Protocol Policy - HTTPS/HTTP etc
- Allowed HTTP Methods - GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE
- Cached HTTP Methods - GET, HEAD
- Cookies
- TTL - Min, Max, Default
- Signed URLs or Signed Cookies
- Can specify AWS WAF - Layer 7 Firewall
- Can enable Logging
- Can take up to an hour to create. Also takes a good amount of time to disable.

After creating your CloudFront Distribution you can edit settings. You can also invalidate objects or paths you don't wish to have cached. This removes it from the edge locations.

Snowball:

256 bit encryption and TPM for transfers of data. Comes in either 50TB or 80TB in size.

Imports or Exports data to/from S3.

Available Internet Connection	Theoretical Min. Number of Days to Transfer 100TB at 80% Network Utilization	When to Consider AWS Import/Export Snowball?
T3 (44.736Mbps)	269 days	2TB or more
100Mbps	120 days	5TB or more
1000Mbps	12 days	60TB or more

Snowball Edge:

100TB in size. On-board storage and compute capabilities such as Lambda. You can use this as a temp storage tier for large local datasets or to support local workloads in remote locations. Connects to existing infras using standard storage interfaces, streamlining the data transfer process.

Snowmobile:

100PB per Snowmobile. Good for moving massive amounts of data or performing a data center migration.

Storage Gateway:

Service that connects an on-prem software appliance. Allows you to move data to and from AWS and is cost effective. It replicates data into AWS.

Types:

- File Gateway - NFS & SMB - For flat files, stored directly in S3
- Volume Gateway - iSCSI (Copies of Hard Drives) Creates EBS Snapshots
- Tape Gateway - VTL

EC2

EC2 is a web service that provides resizable compute capacity in the cloud. EC2 reduces the time required to obtain and boot new server instances in minutes, allowing you to quickly scale capacity, both up and down, as requirements change.

Pricing Models:

- On Demand - Allows you to pay a fixed rate per hour without commitment
- Reserved - Provide you with a capacity reservation and offer a significant discount on the hourly charge. These are 1 or 3 year contracts.

Reserved Types:

Standard - 75% off on demand instances.
Convertible - Offer up to 54% off on demand instances. Allows you to change types.
Scheduled - For running in time frames.

- Spot - Enables you to bid whatever price you want for instance capacity, providing even greater savings if you applications have flexible start and end times.

- Dedicated Hosts - Physical EC2 server dedicated for you use. Useful for regulatory compliance. Great for licensing which does not support multi-tenancy or cloud deployments.

EC2 Instance Types:

F - For FPGA
I - For IOPS
G - Graphics
H - High Disk Throughput
T - Cheap General Purpose
D - For Density

R - For RAM
M - Main choice for Gen Purpose apps
C - For Computer
P - Graphics (Think Pictures)
X - Extreme Memory
Z - Extreme Memory and CPU
A - Arm-based Workloads
U - Bare Metal

EC2 Creation:

1. Choose your AMI (OS) to run on the instance.
2. Select the instance type you wish to use.
3. Select the Network (VPC), Subnet and Public IP (If needed)
4. Tie in IAM role, Shutdown behavior, Termination Protection, CW Monitoring, T2/T3 Unlimited
5. 5. Create storage, root being boot drive. You can also add storage volumes here.
6. Add tags for organization and tie to Instance/Volumes.
7. Security group rules to allow port traffic to/from IP/CIDR or All.

- Termination Protection is turned OFF by default, you must turn it on.
- On an EBS-backed instance, the DEFAULT ACTION is for the root EBS volume to be deleted when the instance is terminated.
- EBS Root Volumes of your DEFAULT AMI's cannot be encrypted. You can also use a 3rd party tool (such as bitlocker) to encrypt the root volume. You can also do this by creating your own AMIs.

Security Groups:

- All inbound traffic is blocked by default.
- All outbound traffic is allowed.
- Changes take effect immediately.
- You can have any number of EC2 instances within a sec grp.
- You can have multiple attached to EC2 instances.
- STATEFUL.
- If you create an inbound rule with traffic in that same traffic is allowed out.
- You cannot block specific IP addresses using Sec Grps, instead use Network Access Control Lists.
- You can allow rules, but not deny rules.

EBS:

Provides block storage volumes for use with EC2 instances. It's a virtual hard disk in the cloud.

5 Types of EBS Volumes:

- General Purpose (SSD) - Most Workloads - gp2
- Provisioned IOPS (SSD) - Databases - io1 (Up to 64,000 IOPS)
- Throughput Optimized Hard Disk Drive - Big Data & Data Warehouses - st1
- Cold Hard Disk Drive - File Servers - sc1
- Magnetic - Infrequently Accessed Data - standard

Volumes & Snapshots:

- Snapshots exist on S3.
- Snapshots are point in time copies of your EBS volumes
- Snapshots are incremental - this means that only the blocks that have changed since your last are moved to S3.
- First time snapshots take some time to create.
- When creating a root device volume snapshot you should stop the instance before hand. You can snapshot volumes while the underlying instance is running.
- You can create AMI's from both Volumes and Snapshots.
- You can change the EBS volume size on the fly and also the storage type.
- Volumes are always in the same AZ as the EC2 instance they're running on.
- Can migrate an EC2 volume from one AZ to another using a snapshot.
- To move to a new region you can snapshot, create an AMI from that snapshot. Copy the AMI to your new region and create an EC2 instance there.

AMI Types:

(EBS vs Instance Store)

You can select your AMI based on:

- Region
- OS
- Architecture
- Launch Permissions
- Storage for the Root Device: Instance Store or EBS Backed Volumes

All AMIs are categorized by either backed by Amazon EBS or backed by instance store.

For EBS Volumes: The root device for an instance launched from the AMI is an Amazon EBS volume created from an Amazon EBS snapshot.

For Instance Stored Volumes: The root device for an instance launched from the AMI is an instance store volume created from a template stored in Amazon S3.

- Instance Store Volumes are sometimes called Ephemeral Storage.
- Instance store volumes cannot be stopped. If the underlying host fails you will lose your data.
- EBS backed instances can be stopped. You will not lose the data on this instance if it is stopped.
- You can reboot both, you will not lose your data.
- By default, both ROOT volumes will be deleted on termination. However, with EBS you can tell AWS to keep the root device volume.

Encrypted Root Device Volumes & Snapshots:

- Snapshots of encrypted volumes are encrypted automatically.
- Volumes restored from encrypted snapshots are encrypted automatically.
- You can only share snapshots if they're unencrypted.
- You can now encrypt root device volumes upon creation of an EC2 instance.

To encrypt an unencrypted volume:

1. Create a snapshot of the root device volume
2. Create a copy of the snapshot and select the encrypt option
3. Create an AMI from the encrypted snapshot
4. Use the AMI to launch a new encrypted instance

CloudWatch:

- Monitoring service that watches performance of applications and services within AWS.
- EC2, Autoscaling, ELBS, Route53 | Storage and Content Delivery: EBS Volumes, Storage Gateways, CloudFront.
- EC2 Default Checks: CPU, Network, Disk, Status Check
- Default (5 minute interval | Detail monitoring (1 min interval)
- You can create alarms which trigger notifications.

Provides: Dashboards, Alarms, Events, and Logs

CloudTrail:

- Monitors and provides insight using API calls into who changes what from where.
- Targets Auditing, not performance.

AWS CLI:

- Interact with AWS from anywhere in the world just by using the CLI
- You will need to set up access in IAM
- Commands themselves are not in this exam, but basic commands will be useful to you

IAM with EC2:

- Roles are more secure than storing your access key and secret access key on each EC2 instance.
- Roles are easier to manage.
- Roles can be assigned to an EC2 instance after it is created using either the console or CLI.
- Roles are universal - Use in any region across your account.

Using Bootstrap Scripts with EC2:

- When configuring instances you can enter scripts under the User data portion. This allows you to provision services and patch your instances upon launching.

  Example:

  ```
  #!/bin/bash
  yum update -y
  yum install httpd -y
  service httpd start
  chkconfig httpd on
  cd/ var/www/html
  echo “<html><h1>Hello World!</h1></html>”
  > index.html
  aws s3 mb s3://woopwoopayo
  aws s3 cp index.html s3://woopwoopayo
  ```

EC2 Meta Data:

- Used to get information about an instance (such as public IP)
- curl http://8.8.8.8/latest/meta-data
- curl http://8.8.8.8/latest/user-data

EFS:

- File storage service for EC2 instances. Provides a simple interface that allows to create and configure file systems quickly and easily. With EFS, storage capacity is elastic, growing and shrinking automatically as you add and remove files.
- Allows you to share storage between EC2 instances.
- Works with Lifecycle rules. If things aren't being used, they can be removed after X days.
- Can enable encryption with KMS when creating.
- Makes storage available across multiple AZs.
- You need to open up NFS within the security groups to allow EFS to communicate
- Supports the NFSv4 protocol.
- You only pay for the storage that you use.
- Can support 100s of concurrent connections.
- Read after write consistency.

Placement Groups:

- Clustered Placement Group - A grouping of instances within a single AZ. Recommended for apps that need low network latency, high network throughput or both.
- Spread Placement Group - Recommended for applications that have a small number of critical instances that should be kept separate from one another. Can use multiple AZs.
- Partitioned Placement Group - Multiple EC2 instances within a partition. Each rack has its own network and power source. No two partitions share the same racks. This isolates hardware failure within your application. Can use multiple AZs.
- You can't merge placement groups.
- You can't move an existing instance into a placement group. You can create an AMI from your existing instance, and then launch a new instance from the AMI into a placement group.

Databases

Relational databases are what most people are used to. Think of a spreadsheet with tables and rows. Relational database types on AWS are:

- MSSQL
- Oracle
- MySQL
- PostgreSQL
- Aurora
- MariaDB

RDS (OLTP):

- Multi-AZ - For disaster recovery - Automatic Failover
- Read Replicas - For performance - Write to the first DB is replicated to the read replica. Capped at 5 total of the same RDS database.
- Runs on virtual machines.
- You do not log into RDS operating systems. Amazon is responsible for RDS OS patching and DB patching.
- RDS is NOT serverless. (Aurora however, is)

Non relational Database:

- Collection = table
- Document = row
- Key Value Pairs = fields

Data Warehousing:

Used for BI, an example is SSRS. It's used to pull in large and complex data sets often used for management to do queries on data.

Data Warehousing Solutions is called Redshift. It's for online analytical processing.

ElastiCache:

A web service that makes it easy to deploy, operate, and scale an in-memory cache in the cloud. The service improves the performance of web applications by allowing you to retrieve info from fast, managed, in memory caches, instead of relying entirely on slower disk-based databases.

It supports two open-source in-memory caching engines which are memcached and redis.

Requirement	Memcached	Redis
Simple Cache to offload DB	Yes	Yes
Ability to scale horizontally	Yes	No
Multi-threaded performance	Yes	No
Advanced data types	No	Yes
Ranking/Sorting data sets	No	Yes
Pub/Sub capabilities	No	Yes
Persistence	No	Yes
Multi-AZ	No	Yes
Backup & Restore Capabilities	No	Yes

- Use ElastiCache to increase database and web application performance.
- Reddis is Multi-AZ.
- You can do backups and restores of Reddis.
- If you need to scale horizontally, use Memcached.
- Memcached is quick to deploy and use.

RDS Backups:

- Automated Backups - Allow you to recover within any point in time between 1 and 35 days. Takes one full daily snapshot and stores transaction logs. When you do a recovery you can restore a database state down to the second. These backups are enabled by default, and stored in S3.
- Database Snapshots - Manual process stored even after you delete the original RDS instance, unlike RDS automated backups.
- Whenever you restore either type, the restore version is a new RDS instance with a new DNS endpoint tied to it. Any applications pointed to the previous instance must be changed.

RDS Encryption:

- At rest is supported for MySQL, Oracle, MSSQL, PostgreSQL, MariaDB & Aurora. Encryption is done via the KMS service. Once your RDS instance is encrypted. The data

stored at rest in the underlying storage is encrypted, as are the automated backups, read replicas, and snapshots.

Multi-AZ:

- Give multiple copies of your DB within more than one AZ. This allows you to have an exact copy in another AZ and seamlessly through AWS. RDS fails over to the working should the AZ DB ever go down. This is for DR only and not for improving performance.
- Works with: MSSQL, Oracle, MySQL, PostgreSQL and MariaDB

Read Replicas:

- Copies of the primary DB. Can be configured to point an EC2 instance at a specific replica for performance improvements. The writes from these are written to all your replicas. Allow you to have a RO copy of your PROD DB.
- You can have up to 5 copies of a single DB.
- Must have automated backups on in order to deploy a read replica.
- You can have read replicas of read replicas. (But watch for latency)
- Each replica has its own DNS endpoint.
- You can create read replicas of Multi-AZ source DBs.
- Read replicas can be promoted to be their own databases. This break replication.
- Can have a read replica in a second region.

DynamoDB:

ProvisionedThroughputExceededException means the throughput is not balanced across your partitions. One partition is being subjected to a disproportionate amount of the traffic and is, therefore, exceeding limits.

Use cases: Storing the metadata of BLOB data in S3. Storing JSON. Storing Web Session Data.

- Fast and flexible NoSQL DB service for all applications that need consistent, single-digit millisecond latency at any scale. It is a fully managed database and supports both document and key-value data models. It’s flexible data model and reliable performance makes it a great fit for mobile, web, gaming, ad-tech, IoT, and others.
- Stored on SSD storage
- Spread across 3 geographically distinct data centers
- Eventual Consistent Reads (default) - Across all data is usually released within a second. Repeating a read after a short time should return the updated data.
- Strong Consistent Reads - Reads the data within a second.

Redshift:

- Fast, powerful, fully managed, petabytes scale data warehouse service in the cloud. Customers can start at 0.25 an hour with no commitment or upfront costs. If you scale past a petabyte or more it's $1,000 per terabyte per year, less than a 10th of most other data warehousing solutions.
- Single Node (160Gb)
- Multi-Node (Leader Node - Manages client connections and receives queries. Compute Node - Stores data and performs queries and computations) Up to 128 Compute Nodes
- Advanced Compression - Compresses individual columns.
- Massively Parallel Processing - Automatically distributes data and query load across nodes to your data warehouse and enables you to maintain fast query performance as your data warehouse grows.
- Backups enabled by default with a 1 day retention, can be set up to 35 days. Red shift always attempts to maintain at least three copies of your data. (original and replica on the compute nodes and a backup in S3)
- Redshift can also asynchronously replicate your snapshots to S3 in another region for disaster recovery.
- Priced based on compute node hours. You are billed for 1 unit per node per hour. A 3 node data warehouse cluster running for an entire month would incur 2,160 instance hours. You will not be charged for leader not hours, only compute hours. You are charged for backups and data transfers.
- Encrypted in transit using SSL and at rest using AES-256.
- By default RedShift takes care of key management. Can use HSM and KMS.
- Redshift only runs in 1 AZ currently.
- Can restore snapshots to a new AZ in case of an outage.

Aurora:

- Strats with 10GB, scales in 10GB increments to 64TB max. (Storage Autoscaling)
- Compatible ONLY with MySQL and PostgreSQL
- Compute resources can scale up to 32vCPUs and 244GB of Memory.
- 2 copies of your data are contained in each AZ, with a minimum of 3 AZs. 6 copies total.
- Designed to handle the loss of two copies of your data without affecting write availability and up to three copies without affecting read availability.
- Aurora storage is self-healing. Data blocks and disks are continuously scanned for errors and repaired automatically.
- Two replica types to choose from: Aurora Replicas (15 currently) | MySQL Read Replicas (Currently 5)
- Automated failover only works with Aurora Replicas.

Feature	Amazon Aurora Replicas	MySQL Replicas
Number of replicas	Up to 15	Up to 5
Replication type	Asynchronous (milliseconds)	Asynchronous (seconds)
Performance impact on primary	Low	High
Act as failover target	Yes (no data loss)	Yes (potentially minutes of data loss)
Automated failover	Yes	No
Support for user-defined replication delay	No	Yes
Support for different data or schema vs. primary	No	Yes

- Automated backups are always enabled on Aurora DB instances. Backups do not impact performance.
- You can also take snapshots with Aurora, this also doesn't impact performance.
- You can share Aurora snapshots with other AWS accounts.
- MySQL -> Aurora by creating a snapshot or creating a read replica and promoting it.

DNS

Route53:

- ELBs do not have a pre-defined IPv4 address, you resolve using a DNS name.
- Understand the differences between an Alias Record and a CNAME.
- Given the choice on the exam, always choose Alias over a CNAME.
- Routing is controlled using the a policy type. All types are displayed below.

Simple Routing Policy:

- One record with multiple IP addresses. If you specify multiple values in a record, Route53 returns all values to the user in a random order.

Weighted Routing Policy:

- Splits traffic based on load and determines where to resolve to.

Latency Routing Policy:

- Sends user traffic based on the lowest latency time between targets.

Failover Routing Policy:

- Sends traffic to whatever target is alive. If your active target was to fail, traffic is sent to the passive target.

Geolocation Routing Policy:

- Lets you choose where your traffic will be sent based on the country or region users are coming from.

Health Checks:

- You can set health checks on individual record sets.
- If a record set fails a health check, it will be removed from Route53 until it passes the health check.
- You can set SNS notifications to alert you if a health check is failed.

VPCs

- When you create a VPC, a default Route Table, Network Access Control List and Security Group are created. It doesn't create any subnets or an internet gateway.
- Amazon always reserves 5 IPs within your subnets.
- You can only have one Internet Gateway per VPC.
- Security Groups can't span VPCs.
- Think of a VPC as a logical datacenter in AWS.
- Consists of IGWs (Or Virtual Private Gateways). Route Tables, Network Access Control Lists, Subnets, and Security Groups.
- 1 Subnet = 1 AZ
- Security Groups are Stateful; Network ACLs are stateless - Can add deny and allow. When you add inbound it doesn't open outbound.
- NO TRANSITIVE PEERING (Must create peering between every VPC.)

VPC Flow Logs:

- Can be created at the VPC, Subnet or Network Interface level.
- Uses S3 to store the logs.
- You cannot enable flow logs for VPCs that are peered with your VPC unless the peer VPC is in your account.
- You cannot tag a flow log.
- After you've created a flow log, you cannot change its config. For example, you can't associate a new or different IAM role with an existing flow log.
- Not ALL IP traffic is monitored. Traffic generated by instances when they contact the Amazon DNS server. If you use your own DNS server, all traffic to that DNS server is

logged. Traffic generated by a Windows instance for Amazon Windows license activation. Traffic to and from 169.254.169.254 for metadata. DHCP Traffic. Traffic to the reserved IP address for the default VPC router.

NAT Instances:

- When creating a NAT instance, Disable Source/Destination Check on the Instance.
- NAT instances must be in a public subnet.
- There must be a route out of the private subnet to the NAT instance, in order for it to work.
- The amount of traffic that NAT instance can support depends on the instance size. If you're bottlenecking, increase the size.
- You can create high availability using Autoscaling Groups, multiple subnets in different AZs, and a script to automate failover.
- Always exist behind a Security Group

NAT Gateways:

- Redundant inside the AZ.
- Preferred by enterprise.
- Starts at 5Gbps and scales up to 45Gbps.
- No need to patch the OS.
- Not associated with a Security Group.
- Automatically assigned a public IP address.
- Remember to update your route tables.
- No need to disable Source/Destination checks.

If you have resources in multiple AZs and they share on NAT gateway, in the event that the NAT gateway's AZ is down, resources in the other AZ will lose internet access. To create an AZ-independent architecture, create a NAT in each AZ you're using and configure the route tables to ensure they're targeting the proper NAT gateway.

Network ACLs:

- You can associate a Network ACL with multiple subnets. However, a subnet can be associated with only one Network ACL at a time. When you associate a network ACL with a subnet, the previous association is removed.
- Contain a numbered list of rules that is evaluated in order, starting with the lowest number.
- Have separate inbound and outbound rules, and each rule can either allow or deny traffic.
- Stateless; responses to allow inbound traffic are subject to the rules for outbound traffic (and vice versa.)

Bastion Host:

- A computer on a network designed and configured to withstand attacks. The computer generally hosts a single application, for example a proxy. It is hardened primarily due to its location and purpose, which is either on the outside of a firewall or in a DMZ and usually involves access from untrusted networks and computers. It's often used to SSH or RDP into private AWS subnets.
- A NAT gateway or NAT instance is used to provide internet traffic to EC2 instances in private subnets.
- It's used to securely administer EC2 instances.
- You cannot use a NAT gateway as a bastion host.

Direct Connect:

NOT A VPN

- Used to connect on premise networks to AWS networks.
- Useful for high throughput workloads.
- If you need a stable and reliable secure connection to AWS.

VPC Endpoint:

- Enables you to privately connect your VPC to AWS services and doesn't require an Internet Gateway, NAT device, VPN or AWS Direct Connect connection. Instances in your VPC do not require public IP addresses to communicate with resources in the service. Traffic between your VPC and the other services don't leave the Amazon network.
- Endpoints are virtual devices. They are horizontally scaled, redundant and highly available VPC components that allow communication between instances in your VPC and services without imposing availability risks or bandwidth constraints on your network.

VPN connections in AWS consist of both a Virtual Private Gateway and Customer Gateway.

Load Balancers

LB Types:

- Application - Balance HTTP/HTTPS traffic. Operate at Layer 7 and are application aware.
- Network - Balance TCP traffic where extreme performance is required. Operate at Layer 4, capable of handling millions of requests per second.
- Classic - Legacy and not often used. Often for HTTP/HTTPS.

Error 504 (Gateway Timeout) means not responding. If you ever have this issue when resolving an ELB your web application is having issues.

If you need the IPv4 address of an end user, look for the X-Forwarded-For header in your logs.

- Instance monitored by ELB are reports as: InService or OutofService
- Health checks check the instance health by talking to it.
- Load Balancers has their own DNS name. You are never giving an IP address.

Advanced Load Balancer Theory:

- Sticky sessions enable your users to stick to the same EC2 instance. This is useful if you are storing information locally to that instance.
- Cross Zone LBs enable you to load balance across multiple AZs.
- Path patterns allow you to direct traffic to different EC2 instances based on the URL contained in the request.

Autoscaling Groups:

- Used to spin up additional EC2 instances based on load.
- It's most common to target CPU usage for these.
- You assign the VPC, # of instances you'd like and Subnet upon creation.
- You assign a warm up period (Boot OS and Services) time when creating.

HA Architecture

- Always design for failure.
- Use multiple AZs in multiple Regions wherever you can.
- Know the difference between Multi-AZ and Read Replicas for RDS.
- Know the difference between scaling out (ASGs) and scaling up (Increase Resource or Change Instance Type).
- Know ALL the S3 storage classes.

CloudFormation

- A way of scripting your cloud environment for provisioning.
- Quick Start is a bunch of CF templates already built by AWS Solution Architects that allow you create complex environments quickly.

Elastic Beanstalk

- Aimed at developers and used for the deployment of AWS services.

- Allows you to quickly deploy and manage applications in the AWS cloud without worrying about the infrastructure that runs those applications. You simply upload your application, and it automatically handles the details of capacity provisioning, load balancing, scaling and application health monitoring.

Applications

SQS:

A web service that gives you access to a message queue that can be used to store messages while waiting for a computer to process them. It's a distributed queue system that enables web service applications to quickly and reliably queue messages that one component in the application generates to be consumed by another component. A queue is a temp repo for messages that are awaiting processing.

WaitTimeSeconds - When the consumer instance polls for new work, the SQS service will allow it to wait a certain time for one of messages to be available before closing the connection.

- SQS is pull based, not pushed based.
- Messages are 256 KB in size.
- Messages can be kept in queue from 1 minute to 14 days.
- Default retention is 4 days.
- Visibility TimeOut is the amount of time that the message is invisible in an SQS queue after a reader picks up the message. Provided the job is processed before the visibility timeout expires, the message will then be deleted from the queue. If the job is not processed within that time, the message will become visible again and another reader will process it. This could result in the same message being delivered twice.

SWF:

Simple Work Flow service is a web service that makes it easy to coordinate work across distributed application components. It's used to be designed for the coordination of tasks. Tasks in this case represent invocations of various processing steps in an application which can be performed by executable code, web service calls, human action, and scripts.

SWF Actors -

- Workflow Starters - An application that can initiate (start) a workflow. Could be your e-commerce website following the placement of an order, or a mobile app searching for bus times.
- Deciders - Controls the flow of activity tasks in a workflow execution. If something has finished or failed, a decider decides what to do next.
- Activity Works - Carry out the activity tasks.

SQS vs SWF:

- SQS has a retention period of up to 14 days; with SWF, workflow executions can last up to one year.
- SWF presents a task-oriented API, whereas Amazon SQS offers a message-oriented API.
- SWF ensures that a task is assigned only once and is never duplicated. With SQS, you need to handle duplicate messages and may also need to ensure that a message is processed only one.
- SWF keeps track of all the tasks and events in an application. With SQS you have to implement your own application-level tracking, especially if your application uses multiple queues.

SNS:

Simple notification service allows you to set up, operate, and send notifications from your account. It provides developers with a highly scalable, flexible, and cost-effective solution to publish messages from an application and immediately deliver them to subscribers or another application.

- Allows push notifications on mobile devices.
- SMS, Email and HTTP endpoints are also supported.
- Allow you to group multiple recipients on a topic basis.
- One topic can send notifications to multiple types of devices.
- All messages stored in SNS are stored across multiple AZs.
- Instantaneous push-based delivery
- Simple API and easy integration with your applications.
- Flexible pay-as-you-go model with no upfront cost.

SNS vs SQS:

- Both messaging services in AWS.
- SNS - Push
- SQS - Pull

Elastic Transcoder:

- Media transcoder in AWS
- Converts media files from their original format into different formats to play across smartphones, tablets, PCs, etc.
- Provides transcoding presents for popular output formats, which means that you don’t need to guess about which settings work best on particular devices.

- Pay based on minutes that you transcode and the resolution at which you do.

API Gateway:

A service that makes it easy for developers to publish, maintain, monitor, and secure APIs at any scale. It's a doorway into your AWS environment. It can communicate with Lambda, EC2 and DynamoDB.

- Expose HTTPS endpoints to define a RESTful API.
- Serverless-ly connect to services.
- Send each API endpoint to a different target.
- Run efficiently with low cost.
- Scale effortlessly.
- Track and control usage using an API key.
- Throttle requests to prevent attacks.
- Connect to CloudWatch for monitoring.
- Users API Gateway domain, by default.
- Support custom domains.
- Supports AWS Cert Manager for free SSL/TLS certs.
- CORS is enforced by the client.

API Gateway Setup-

1. Select supported HTTP methods (verbs).
2. Set security rules.
3. Choose your gateway resource targets (such as EC2, Lambda, DynamoDB, etc.).

Kinesis:

A platform that allows you to send streaming data. This makes it easy to load and analyze streaming data. It also provides the ability for you to build your own custom applications for business needs.

Kinesis Types:

- Streams - Stores data for 24 hours to 7 days. It contains the data in shards. EC2 can access these shards and store the data in other services.
- Firehose - If you want to analyze data on the fly with Lambda you use firehose.
- Analytics - Analyzes the data in both platforms.

Web Identity Federation & Cognito:

- Sign-up and sign-in to your apps using a WEB ID provider (Google, FB, AMZN)
- Access for guest users.
- Acts as an ID broker between your application and Web ID providers, there's no need for additional code.
- Synchronized user data from multiple devices.
- Recommended for all mobile application AWS services.
- User pool is user based. It handles things like user registration, auth, and account recovery.
- Identity pools authorise access to your AWS resources.

Serverless

Lambda:

It's an Ultimate abstraction layer. It's a computer service where you can upload code and create functions with said code. You can use it as an event-driven computer service by defining triggers. It can also run in a response with an AWS SDK.

API Gateway, DynamoDB, and S3 are capable of invoking directly via Lambda.

- First 1 million requests are free. .20 per 1 million requests after. You're also billed for the duration of the request.
- Scales out (not up) automatically.
- Lambda functions are independent, 1 event = 1 function.
- AWS X-ray allows lambda debugging.
- Lambda can do things globally, so you can use it to replicate S3 buckets for instance.
- RDS cannot trigger Lambda.