**Six advantages of cloud computing –**

Pay-as-you-go model

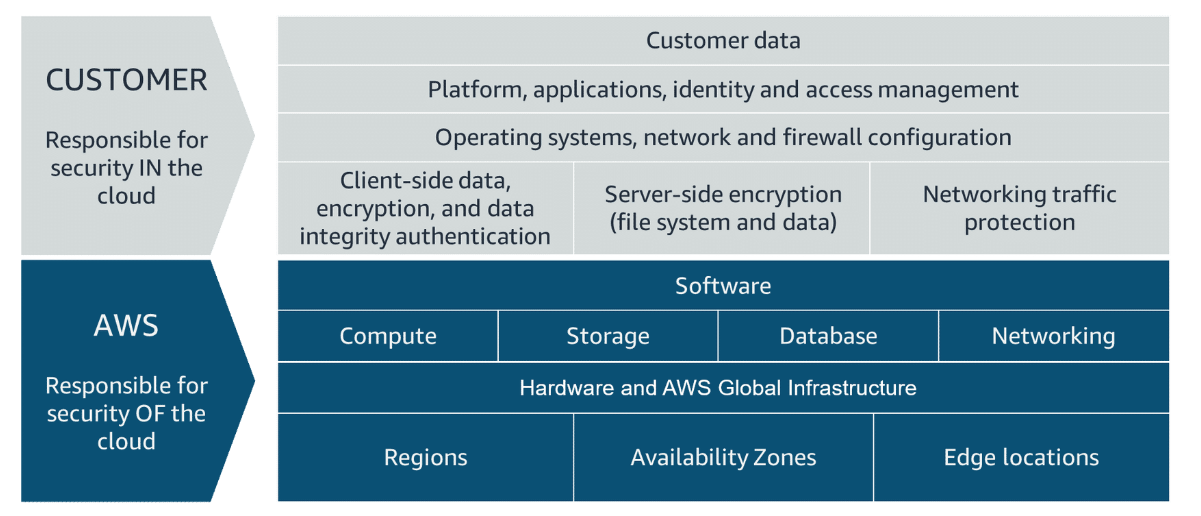
Benefit from massive economies of scale

Stop guessing capacity

Increase speed and agility

Realize cost savings

Go global in minutes



**AWS Region Considerations**

* Compliance
* Latency
* Price
* Service availability

**AWS root user credentials**

Email address + password

And access keys if accessing the CLI (consists of the access key ID and secret access key)

**Policy for IAM has four major JSON elements**

Version, Effect, Action and Resource

The Version element defines the version of the policy language. It specifies the language syntax rules that are needed by AWS to process a policy. To use all the available policy features, include "Version": "2012-10-17" before the "Statement" element in your policies.

The Effect element specifies whether the policy will allow or deny access. In this policy, the Effect is "Allow", which means you’re providing access to a particular resource.

The Action element describes the type of action that should be allowed or denied. In the example policy, the action is "\*". This is called a wildcard, and it is used to symbolize every action inside your AWS account.

The Resource element specifies the object or objects that the policy statement covers. In the policy example, the resource is the wildcard "\*". This represents all resources inside your AWS console.

Three types of storage – block storage, file storage, object storage

File storage – web serving, analytics, media and entertainment, home directories

Block storage – transactional workloads, containers, virtual machines

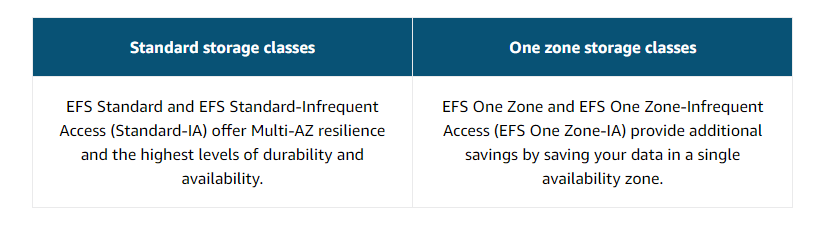
Object storage – data archiving, backup and recovery, rich media

**File Storage Options**

Amazon Elastic File System – only pay for what you use,

EFS Mount Helper – open source software for mounting for Linux

EFS Storage classes –



Amazon FSx – fully managed service to deploy high performance file systems in the cloud

**Block Storage Options**

Amazon EC2 Instance Store – if the instance is stopped, the block cannot be used.

Amazon EBS – this is separate to an EC2 instance, can have multiple EBS volumes linked to an EC2 instance, EBS Multi-attach can be attached to be multiple EC2 instances (not for all instance types and instances must be in the same AZ), size limited to the size of the external drive. If the instance goes down the data will still be available on the EBS. The two main different types of volumes fall under SSD backed volumes and HDD backed volumes. You back up the EBS volumes by taking snapshots. You pay for what you provision so need to provision storage in advance.

**Object Storage**

Amazon S3 – access data through URLs from anywhere on the web, storage as many objects as you would like with the individual object size of 5tb, buckets are created first and the object is placed inside the bucket, buckets are region specific , bucket names must be globally unique across all AWS accounts and must be DNS compliant, no special characters etc in bucket names, everything in a S3 is private by default so only viewed by the user or the AWS account that created that resource.

S3 Bucket Policies – use JSON format, specify what actions are allowed or denied on the bucket, can only be placed on the bucket

S3 is good to store static content, static websites, data lakes, backup and storage, media hosting, software delivery

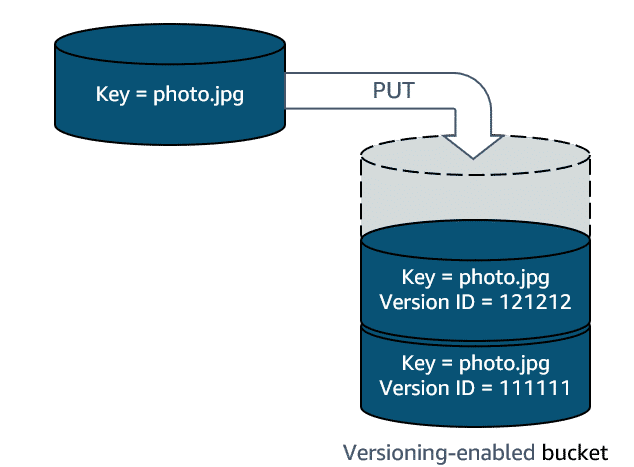
S3 uses Amazon S3 standard storage class as the default if one is not selected.

|  |  |
| --- | --- |
| **Storage Class** | **Description** |
| **S3 Standard** | This is considered general-purpose storage for cloud applications, dynamic websites, content distribution, mobile and gaming applications, and big data analytics. |
| **S3 Intelligent-Tiering** | This tier is useful if your data has unknown or changing access patters. S3 Intelligent-Tiering stores objects in three tiers: a frequent access tier, an infrequent access tier, and an archive instance access tier. Amazon S3 monitors access patterns of your data and automatically moves your data to the most cost-effective storage tier based on frequency of access. |
| **S3 Standard-Infrequent Access (S3 Standard-IA)** | This tier is for data that is accessed less frequently but requires rapid access when needed. S3 Standard-IA offers the high durability, high throughput, and low latency of S3 Standard, with a low per-GB storage price and per-GB retrieval fee. This storage tier is ideal if you want to store long-term backups, disaster recovery files, and so on. |
| **S3 One Zone-Infrequent Access (S3 One Zone-IA)** | Unlike other S3 storage classes that store data in a minimum of three Availability Zones, S3 One Zone-IA stores data in a single Availability Zone, which makes it less expensive than S3 Standard-IA. S3 One Zone-IA is ideal for customers who want a lower-cost option for infrequently accessed data, but do not require the availability and resilience of S3 Standard or S3 Standard-IA. It's a good choice for storing secondary backup copies of on-premises data or easily re-creatable data. |
| **S3 Glacier Instant Retrieval** | Use S3 Glacier Instant Retrieval for archiving data that is rarely accessed and requires millisecond retrieval. Data stored in this storage class offers a cost savings of up to 68 percent compared to the S3 Standard-IA storage class, with the same latency and throughput performance. |
| **S3 Glacier Flexible Retrieval** | S3 Glacier Flexible Retrieval offers low-cost storage for archived data that is accessed 1–2 times per year. With S3 Glacier Flexible Retrieval, your data can be accessed in as little as 1–5 minutes using an expedited retrieval. You can also request free bulk retrievals in up to 5–12 hours. It is an ideal solution for backup, disaster recovery, offsite data storage needs, and for when some data occasionally must be retrieved in minutes. |
| **S3 Glacier Deep Archive** | S3 Glacier Deep Archive is the lowest-cost Amazon S3 storage class. It supports long-term retention and digital preservation for data that might be accessed once or twice a year. Data stored in the S3 Glacier Deep Archive storage class has a default retrieval time of 12 hours. It is designed for customers that retain data sets for 7–10 years or longer, to meet regulatory compliance requirements. Examples include those in highly regulated industries, such as the financial services, healthcare, and public sectors. |
| **S3 on Outposts** | Amazon S3 on Outposts delivers object storage to your on-premises AWS Outposts environment using S3 API's and features. For workloads that require satisfying local data residency requirements or need to keep data close to on premises applications for performance reasons, the S3 Outposts storage class is the ideal option. |

**S3 Versioning**

Without Amazon S3 versioning, every time you upload an object called employee.jpg (example) it will overwrite the original object. This can be an issue when using common names or if you are wanting to preserve a version.

To counteract these issues, you can use Amazon S3 versioning. Versioning keeps multiple versions of a single object in the same bucket. This preserves old versions of an object without using different names, which helps with object recovery from accidental deletions, accidental overwrites, or application failures.



By using versioning-enabled buckets, you can recover objects from accidental deletion or overwrite. The following are examples:

Deleting an object does not remove the object permanently. Instead, Amazon S3 puts a marker on the object that shows that you tried to delete it. If you want to restore the object, you can remove the marker and the object is reinstated.

If you overwrite an object, it results in a new object version in the bucket. You still have access to previous versions of the object.

**Versioning States**

Unversioned (default) – no new and existing objects have a version

Versioning-enabled - Versioning is enabled for all objects in the bucket. After you version-enable a bucket, it can never return to an unversioned state. However, you can suspend versioning on that bucket.

Versioning-suspended - Versioning is suspended for new objects. All new objects in the bucket will not have a version. However, all existing objects keep their object versions.

**Managing your storage lifecycle**

If you keep manually changing your objects, such as your employee photos, from storage tier to storage tier, you might want to automate the process by configuring their Amazon S3 lifecycle. When you define a lifecycle configuration for an object or group of objects, you can choose to automate between two types of actions: transition and expiration.

Transition actions define when objects should transition to another storage class.

Expiration actions define when objects expire and should be permanently deleted.

**S3 Bucket Naming Rules**

3-63 characters long

S3 bucket names are used to form URL links so rules are similar to valid URL rules

Must be unique across all AWS accounts in all AWS Regions, name can’t be reused until the original bucket is deleted

Buckets used with S3 Transfer Acceleration can’t have dots in their names

No uppercase, underscores in names

**S3 Bucket General Info**

By default, you can create 100 buckets (can create a service request to create up to 1000 buckets) and you must empty a bucket first before you can delete it

No max bucket size and no limit to the number of objects in a bucket – files can be between 0 and 5 TBs.

Two types of buckets – general purpose buckets (flat hierarchy), directory buckets (folder hierarchy) S3 express one zone storage class

**S3 Checksum -** The checksum will determine if something is wrong with the file. AWS allows you to change the checksum algorithm when uploading an object.

**S3 Access Grants** – map identities in a directory service (AD, okta)

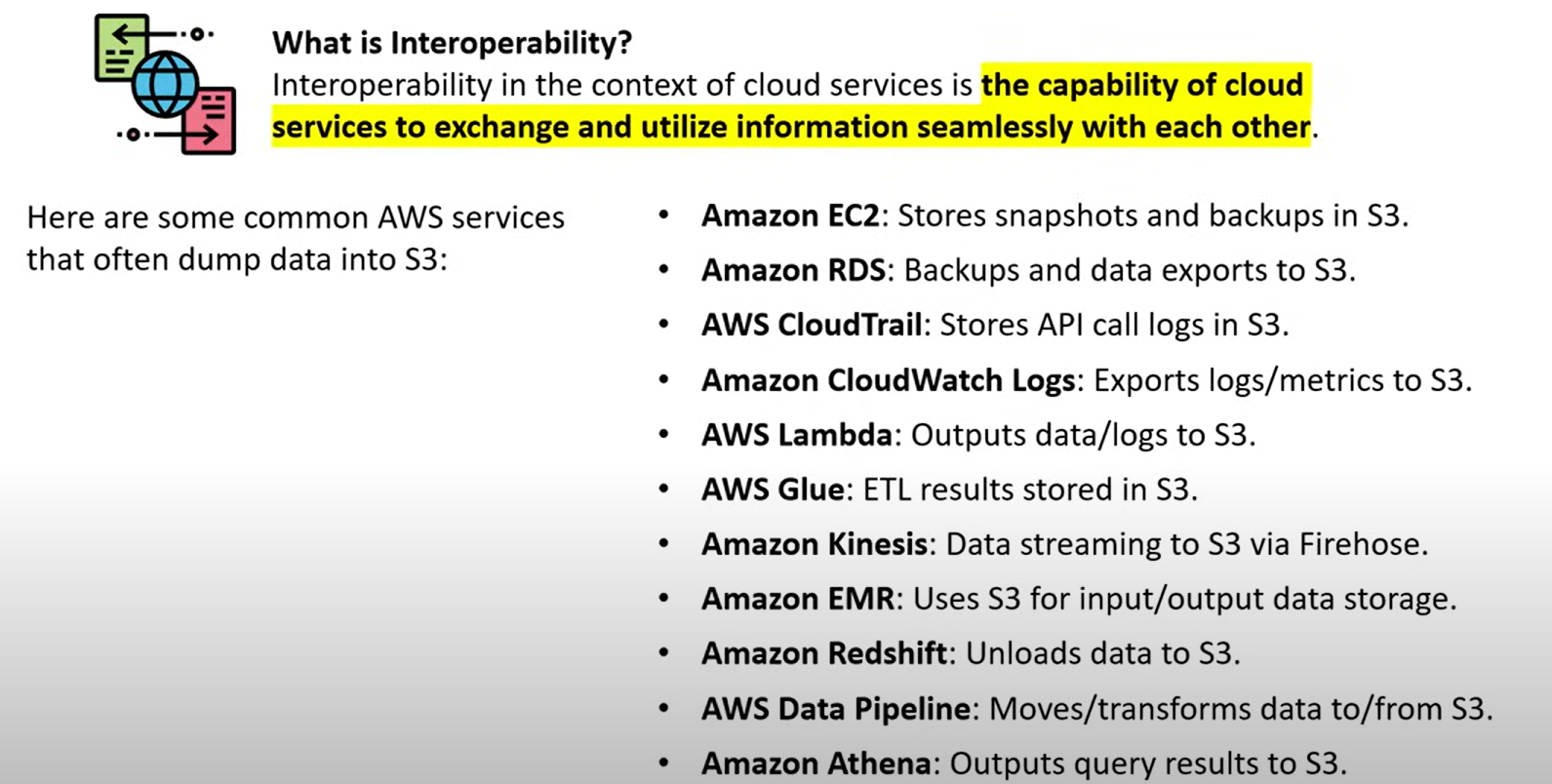
**S3 Select** – uses SQL to filter contents of S3 objects

**S3 Storage Class Analysis** – recommend objects to move between Standard and Standard\_IA

**Amazon S3 Storage Lens** – storage analysis tool, identify cost optimization, how much storage, improve performance

**S3 Static Website Hosting** – Amazon Cloudfront must be used to server HTTPS traffic, website hosting will provide a website endpoint

**S3 Multipart upload** – uploading single object in a set of parts, improved throughput



**AWS Databases**

**Relational database management system (RDS) -**

With a relational database management system (RDBMS), you can create, update, and administer a relational database. Some common examples of RDBMSs include the following:

* MySQL
* PostgresQL
* Oracle
* Microsoft SQL Server
* Amazon Aurora

You communicate with an RDBMS by using structured query language (SQL) queries. You can have multi-AZ replicas, replicas in another region or replicas of other read replicas.

DB instances is an isolated database environment running in the cloud, it can contain one or multiple user-created databases. DB instance class is very similar to EC2 instance class where it determines available compute and memory. DB instances use EBS volumes for database and log storage.

|  |  |
| --- | --- |
| **Unmanaged databases** | **Managed databases** |
|  |  |

RDS Proxy creates a connection pooler so that short-lived AWS functions Lambda functions connecting to RDS does not quickly exhaust all connections.

Automated backups are turned on by default. Automated backups are retained between 0 and 35 days. If you want to keep your automated backups longer than 35 days, use manual snapshots.

Master User Account in RDS is the initial account that is created when you provision a new DB instance.

RDS can use a DNS endpoint with public access to connect publicly, this can be done by using Cloudshell with a database client or a database management (DB IDE) tool to connect.

RSA Blue/Green Deployments copy a production database environment in a separate synchronized staging environment.

**Amazon DynamoDB-**

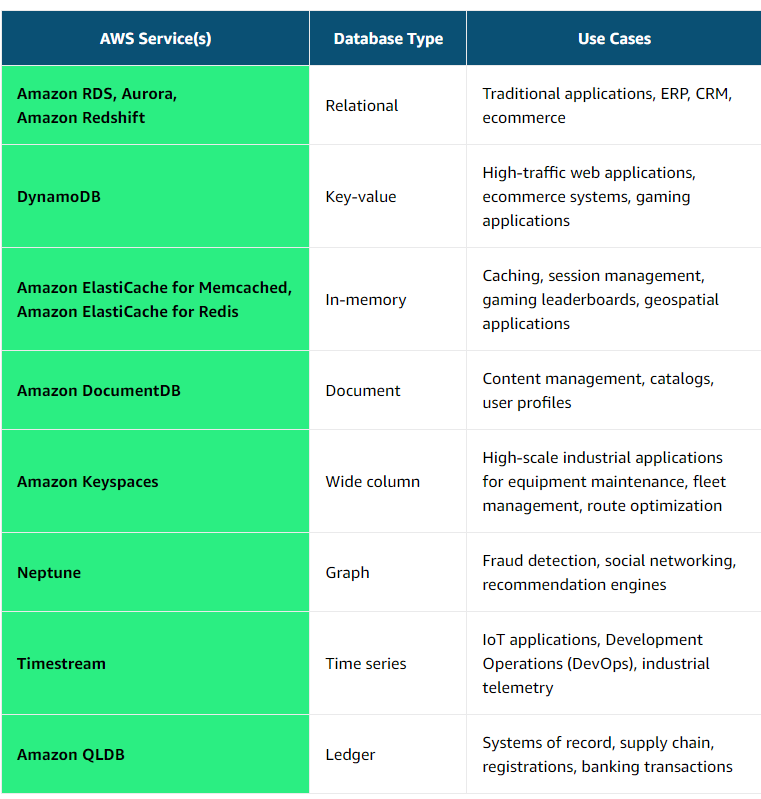
Fully managed NoSQL database service.

* Create database tables that can store and retrieve any amount of data and serve any level of request traffic.
* Scale up or scale down your tables' throughput capacity without downtime or performance degradation.
* Monitor resource usage and performance metrics using the AWS Management Console.

Potential use cases – develop software applications, create media metadata stores, scale gaming platforms and deliver seamless retail experiences.

**Amazon Aurora**

Amazon Aurora is a relational database management system (RDBMS) built for the cloud with full MySQL and PostgreSQL compatibility.



**Monitoring**

Amazon CloudWatch – ability to set alarms based on specific metrics and conditions, can take automated actions like scaling, can be used to be troubleshoot issues,

Log data sent to Cloud watch – event > log stream > group

Three possible states for a Cloudwatch alarm = ok, Alarm and Insufficient\_Data

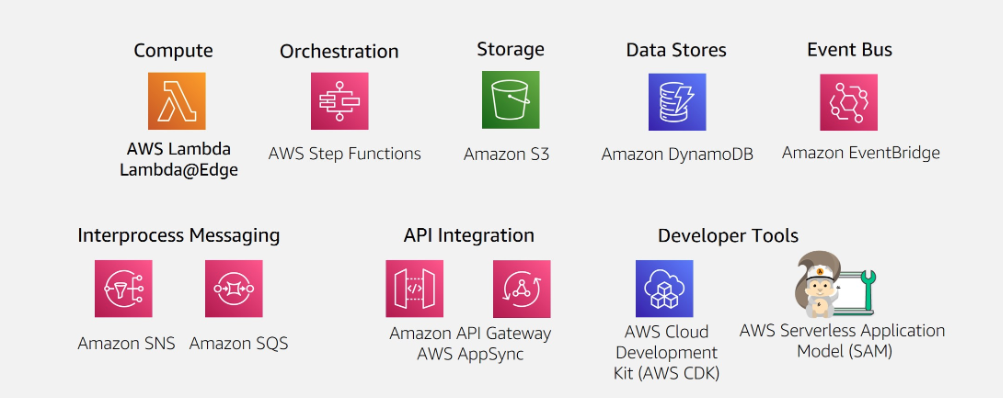
**Lambda Invocation Model**

Synchronous invocation - When you invoke a function synchronously, Lambda runs the function and waits for a response.

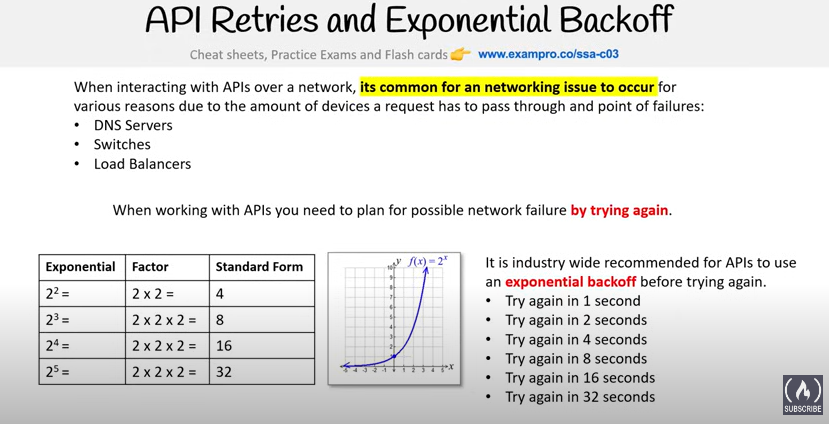
Asynchronous invocation - When you invoke a function asynchronously, events are queued and the requestor doesn't wait for the function to complete.

Polling invocation - This invocation model is designed to integrate with AWS streaming and queuing based services with no code or server management. Lambda will poll (or watch) these services, retrieve any matching events, and invoke your functions

**AWS Serverless Platform**

****

**AWS API**

****

Security Token Service – is a web service that enables you to request temporary limited-privilege credentials for IAM users,

Signing AWS API Requests – you can sign the requests so that AWS can identify who sent them, if you use AWS CLI or AWS SDK the request are signed for you automatically

AWS publishes all of its IP Address ranges at a URL in json format for organizations to whitelist

Service endpoints – to connect programmatically to an AWS service, you use an endpoint

**Security Groups**

Act as a stateful virtual firewall at an instance level

Can contain multiple instances in different subnets

Contains two different sets of rules – inbound and outbound rules

60 inbound rules and 60 outbound rules per security group

**Security Groups Vs Network ACLs**

|  |  |
| --- | --- |
| Security group is the firewall of EC2 Instances.  Security group supports allow rules only.  Instance can have multiple Security groups.  Security groups are stateful. | Network ACL is the firewall of the VPC Subnets.  Network ACL supports allow and deny rules.  Subnet can have only one NACL.  Network ACLs are stateless. |

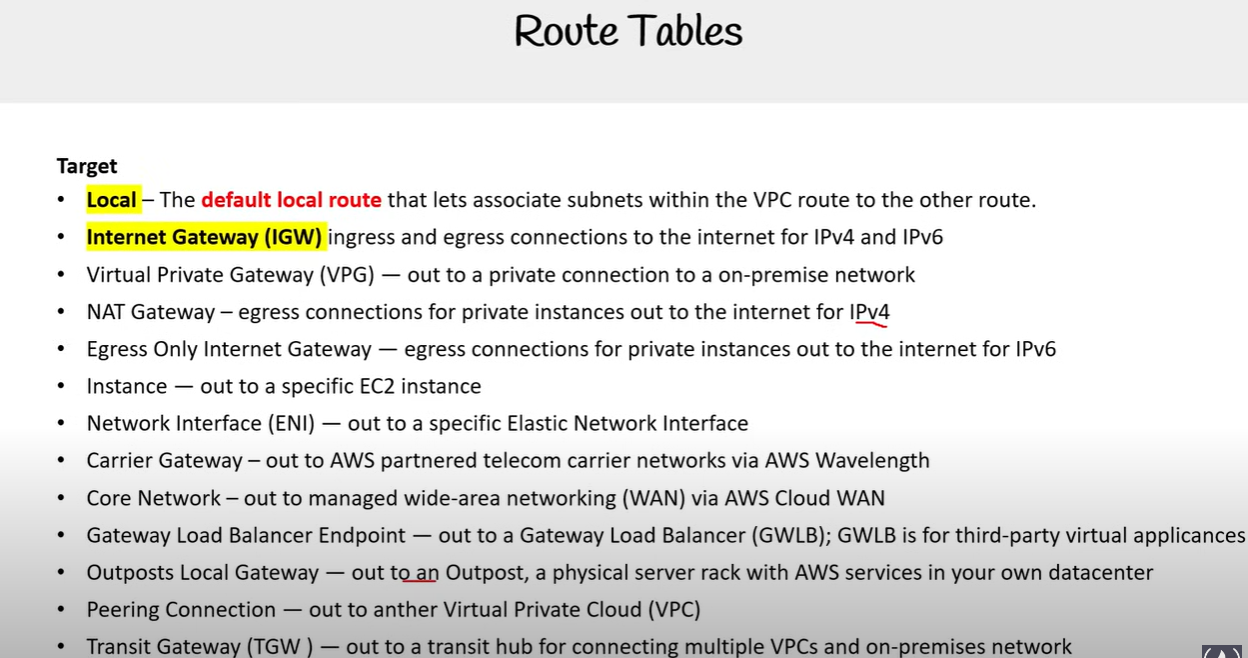
**Route Tables**

Route tables are used to determine where network traffic is directed

Each subnet must be associated with a route table

Main Route table – a default route table created alongside every VPC which cannot be deleted.

Custom route table – a route table you can create for your VPC



**Gateways**

Internet gateway – inbound and outbound for IPv4 and IPV6

Egress-Only internet gateway – outbound private traffic for ipv6

Carrier gateway – connecting to AWS partnered telecom network

NAT Gateway – outbound private traffic for IPv4

Virtual Private Gateway – endpoint into your AWS account for a VPN connection

Customer Gateway – endpoint into your on-premise account for a VPN connection

Gateway Load Balancer (GWLB) – layer 3 load balancer to run and scale third-party virtual applications – firewall, etc

Direct Connect Gateway – endpoint connection to a fiber optic connection at co-location data center

AWS Backup Gateway – endpoint connection for AWS managed backups

IoT Device Gateway – endpoint connection to send IoT data in both directions

AWS Transit Gateway – Hub and spoke model to simplify VPC peering

Amazon API Gateway - abstracts API endpoints to services

AWS Storage Gateway – syncing, caching and extending local storage to cloud storage

**Direct Connect**

3 factors to pricing – capacity (port size), port hours, data transfer out

**VPC Endpoints**

Allows private connection from your VPC to other AWS services

Eliminates the need for internet gateway, NAT device, VPN connection or AWS Direct Connect

Traffic between your VPC and other services do not leave the AWS network

**Interface Endpoints**

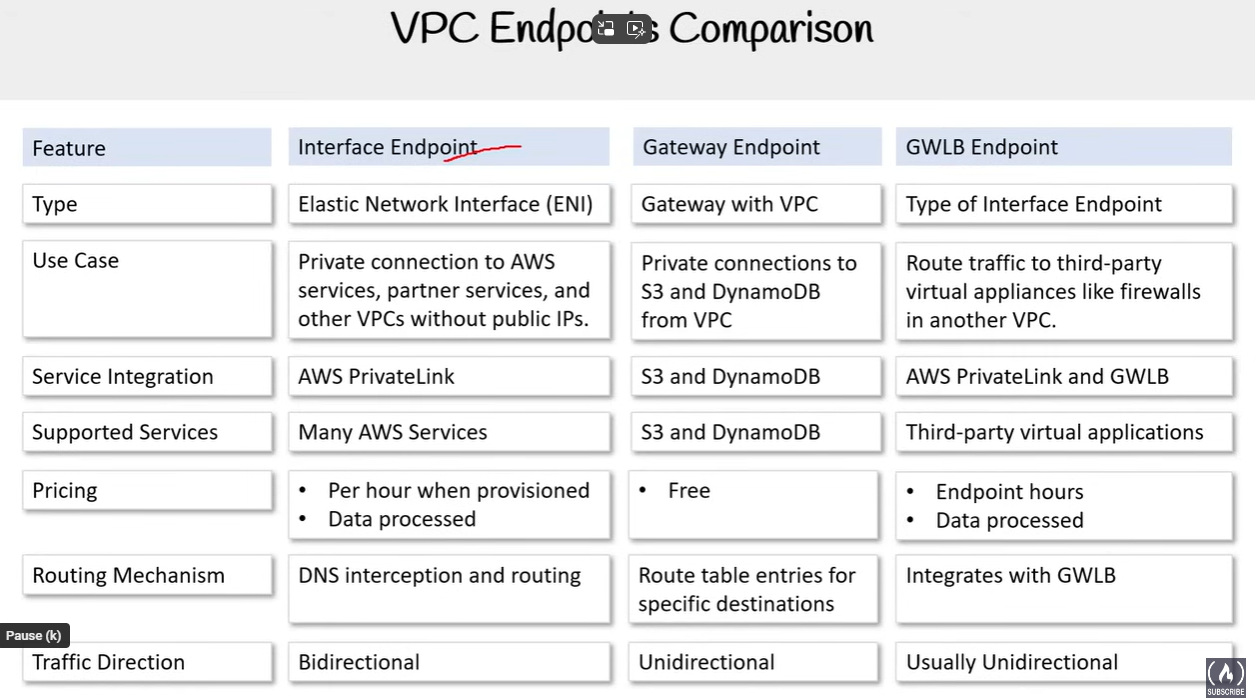
Elastic Network Interfaces (ENI) with a private IP address – entry point for traffic going to a supported service

**Gateway Load Balancer Endpoints**

GWLB Endpoints powered via PrivateLink allows you to distribute traffic to a fleet of network virtual appliances.

**VPC Gateway Endpoints**

A Gateway Endpoint provide reliable connectivity to Amazon S3 and DynamoDB without requiring an internet gateway or a NAT device for your VPC.



**VPC Flow Logs** – allows you to capture IP traffic information through your VPC.

**Virtual Private Gateway** – VPN endpoint on the Amazon side of Site-to-Site VPN connection that can be attached to a single VPC. When you create a VGW you need to assign an Amazon Autonomous System Number and once the VGW is created you cannot change the ASN

**Customer Gateway**- is a resource that you create in AWS to represents the customer gateway device in your on-premises network

**Transit Gateway** – transit hub that you can use to interconnect your VPCs and your on-premises networks, can support either IPv4 or IPv6 traffic inside VPN tunnels. Operates as a virtual router at a regional level. You can attach up to 5000 VPCs to each gateway.

**AWS Client VPN** – fully managed client-based VPN service that enables you to securely access AWS resources in your on-premises network.

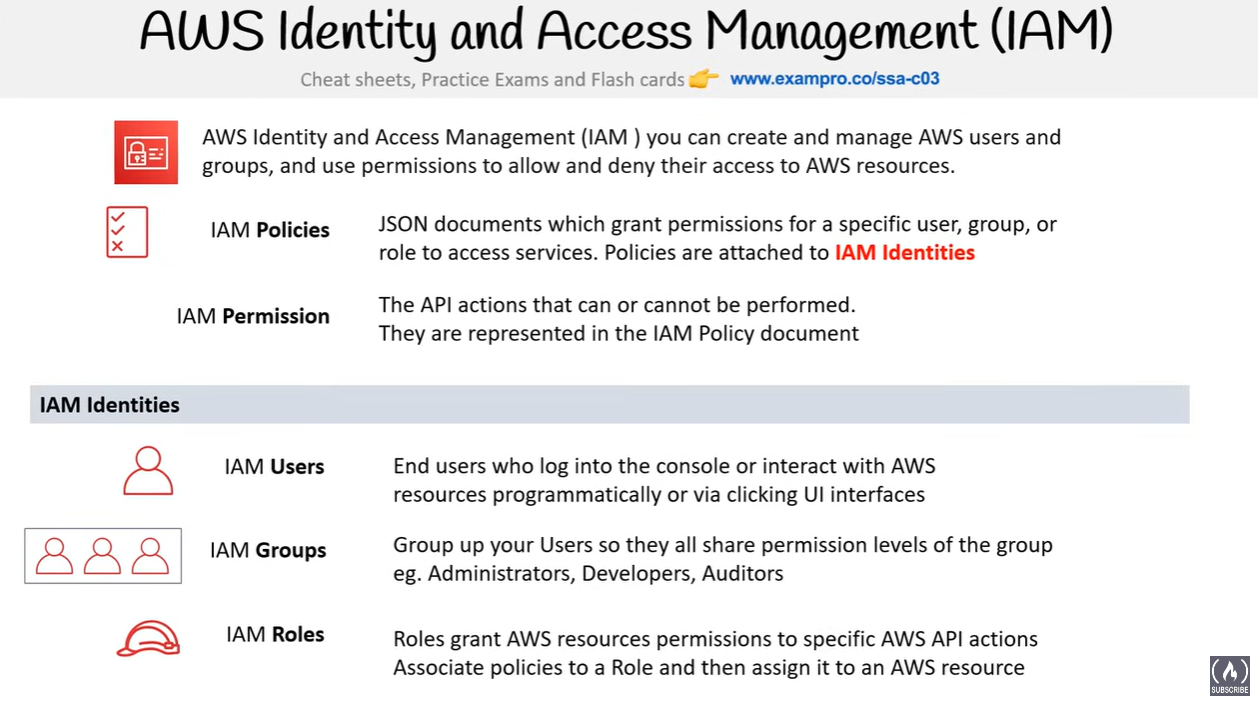
**NAT Gateway** – fully managed NAT service to allow instances in your private subnet to establish outbound connections. A NAT gateway is needed per subnet. You pay per hour and per Gb data processed.

NAT Gateway has two connections modes:

Public (Default) – you must associate an Elastic IP address

Private – instances in private subnets can connect to other VPCs or your on-premises network through a private NAT Gateway. You cannot associate an elastic IP address with a private gateway.

**AWS Network Firewall** – stateful, managed, network firewall



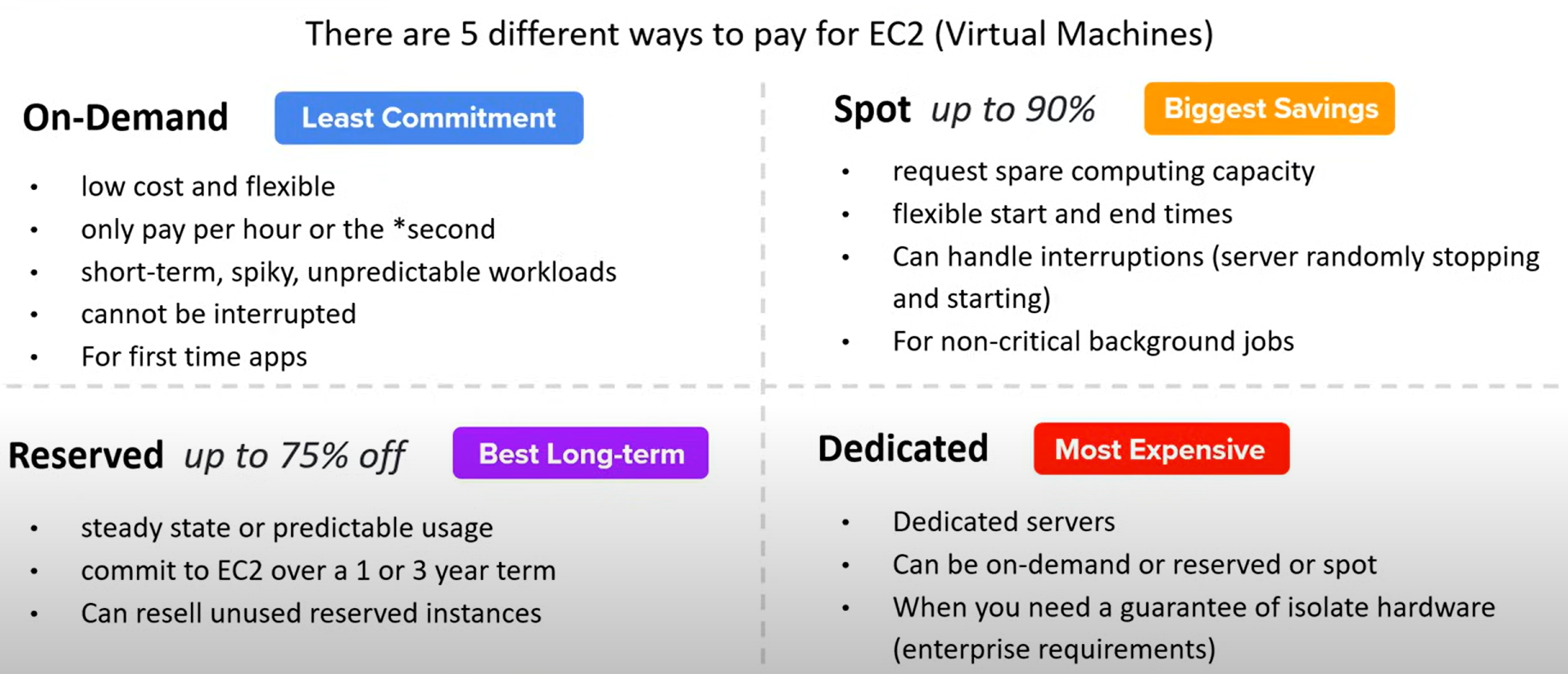
**IAM – Security Token Service (STS)** – is a global service that enables you to request temporary limited privilege credentials for IAM users, it will return an AccessKeyID and SecretAccessKey

**EC2**

**EC2 Instance Profile** – is a reference to an IAM role that will be passed and assumed by the EC2 instance when it starts up, instance profile allows you to avoid passing through live AWS credentials, only a single IAM role can be associated with an instance profile.

**EC2 Placement Groups** – Cluster, Partition, Spread

EC2s pay default use On-Demand pricing.



RI attributes – instance type, region, tenancy, platform



**Auto Scaling Groups** – automatic scaling can occur via capacity settings, health check replacements and scaling policies. ASGs are used to scale EC2 instances. Fargate does not use ASGs.

Capacity settings – setting a min and max for EC2s

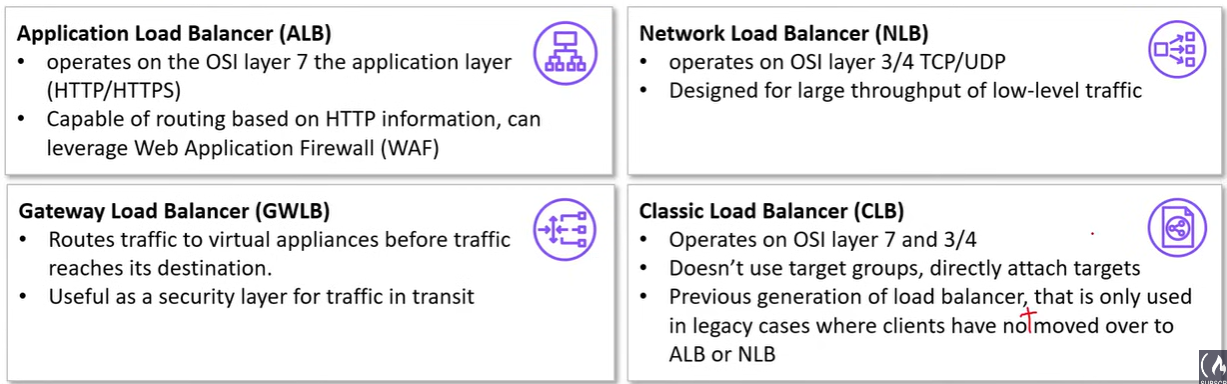
Health check – ec2 health check or ELB health check

Dynamic Scaling Policies – policies are triggered based on Cloud Watch Alarms. There are 3 dynamic scaling policies:

Simple Scaling, Step Scaling, Target Tracking

An Elastic Load Balancer (ELB) can be attached to your Auto Scaling Group (ASG)

**Elastic Load Balancer** – is a suite of load balances use to balance/distribute traffic to multiple EC2, ECS, Fargate and EKS instances.



Listeners – incoming traffic is evaluated against listeners, commonly listen on port 443 or port 80

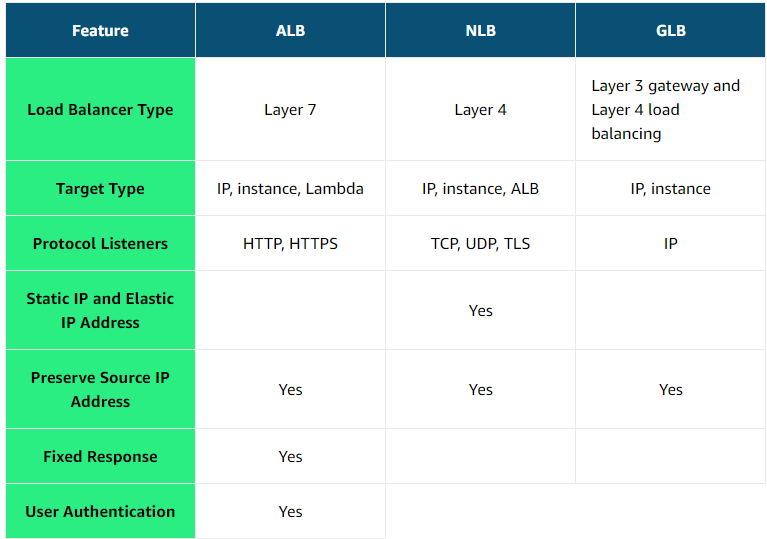
Rules – only available for Application Load Balancer, listeners will then invoke rules to decide what to do with the traffic

Target Groups – (Not available for Classic Load Balancer) are a logical grouping of possible target such as specific EC2 instance, IP addresses

**Application Load Balancer** – designed to balance HTTP and HTTPS traffic, has a feature called Request Routing, can only be accessed via its hostname, Global accelerator can be placed in front of ALB to improve global availability. Use cases include microservices, retail websites, SaaS applications, corporate websites and web applications

**Network Load Balancer** – designed to balance TCP/UDP. Preserves the client source IP. Use cases: real-time and multiplayer gaming platforms, financial trading platforms, IoT and smart device ecosystems, telecommunications networks

**Classic Load Balancer** – first AWS load balancer, not recommended for use



**Route 53** – Domain Name Service, register and manage domains, create various record sets on a domain, resolve VPC’s outside of AWS, continuously monitor records via health checks, implement complex traffic flows

Hosted Zone – a container for record sets scope to route traffic for a specific domain or subdomains.

* Public Hosted Zones – how you want to route traffic inbound from the Internet
* Private Hosted Zones – how you want to route traffic within an Amazon VPC

Record sets – are a collection of records which determine where to set traffic – AA record type etc

**Global Accelerator** – Two types: standard and custom routing, standard will automatically route to the nearest healthy endpoint and custom routing will routing to specific EC2 instances. There is a speed comparison tool to use.

**CloudFront** – content delivery network, Lambda@Edge is a lambda function that overrides the behaviours of requests and responses on CloudFront, CloudFront functions are also able to used – have different properties and use different programming languages to Lambda@Edge.

CloudFront Origin – is where CloudFront will send requests

**AWS Backup** – allows you to centrally manage backups across AWS services, has a backup vault, ability to create a backup plan

**AWS Snow Family-**

* Snowcone – 8tb of storage (HDD) and 14tb of storage (SSD), can physically ship the device back to AWS or AWS DataSync to run on the devices compute
* Snowball Edge – 80tb, 210 tb or 39.5 tb, local processing, edge-computing workloads
* Snowmobile – 100 pb, shipping container

**AWS Migration Hub** – single place to plan migrations and monitor the status

**AWS Auto Scaling** – service that can discover scaling resources within your AWS account, dynamic scaling or predicative scaling can be employed.

**Simple Notification System** – publishers send messages to an event bus and subscribers receive messages based on the subscriptions of the groups, allows you to decouple microservices and serverless applications, destinations can be application to application or application to person, topics allow you to group multiple subscriptions at once, topics can be encrypted via KMS, two types of topics – standard or FIFO (guaranteed to be delivered in an order)

**Simple Queue System** - two types of queue– standard or FIFO (guaranteed to be delivered in an order), libraries on Github used for larger message than the maximum of 256kb with a maximum of 2gb, delay queues allow you to postpone the delivery of new messages to consumers,

**Amazon MQ** – managed message broker service for opensource projects

**CloudWatch** – monitoring solution for your AWS resources, it is an umbrella service meaning it is a collection of monitoring tools, cloudwatch logs are used to monitor, store and access log files, some metrics are not tracked by default and need to have the agent installed, cloudwatch agent can be installed using AWS Systems Manager (SSM)

Cloudwatch alarm – monitors a CloudWatch Metric based on a defined threshold. When the alarm breaches, it changes state and then you can define the action that would trigger. Composite alarms are alarms that watch other alarms to reduce alarm noise.

**Amazon EventBridge** - a serverless event bus that ingests data from your own apps, SaaS apps, and AWS services and routes that data to targets. Rules are used to filter events and routes them to targets for processing. An event bus receives events from a source and routes events to a target based on rules.

**AWS Lambda** – is a serverless compute service for running code without having to provision or manage servers, can be invoked by sync or async invocations, pay as you go pricing, performance optimization, automatic scaling, Lambda runtimes are a preconfigured environment to run specific programming language, deployment packages are a package that contains your function code that Lambda will deploy – can deploy ZIP or IMAGE

**Step Functions** – coordinate multiple AWS services into serverless workflows, two state machines – standard and express for streaming data

**Elastic Beanstalk** – not recommended for Production applications, is powered by CloudFormation template setups

**Amazon Kinesis** – AWS fully managed solution for collecting, processing and analyzing streaming data in the cloud, it allows real time data to be collected

**ElastiCache** – fully managed in-memory datastore for either open-source data stores Memcached or Redis. Elasticache is only accessible by resources in the same VP, two deployment options – standard mode or serverless.

**MemoryDB** – Redis-compatible in-memory database for ultra-fast performance.

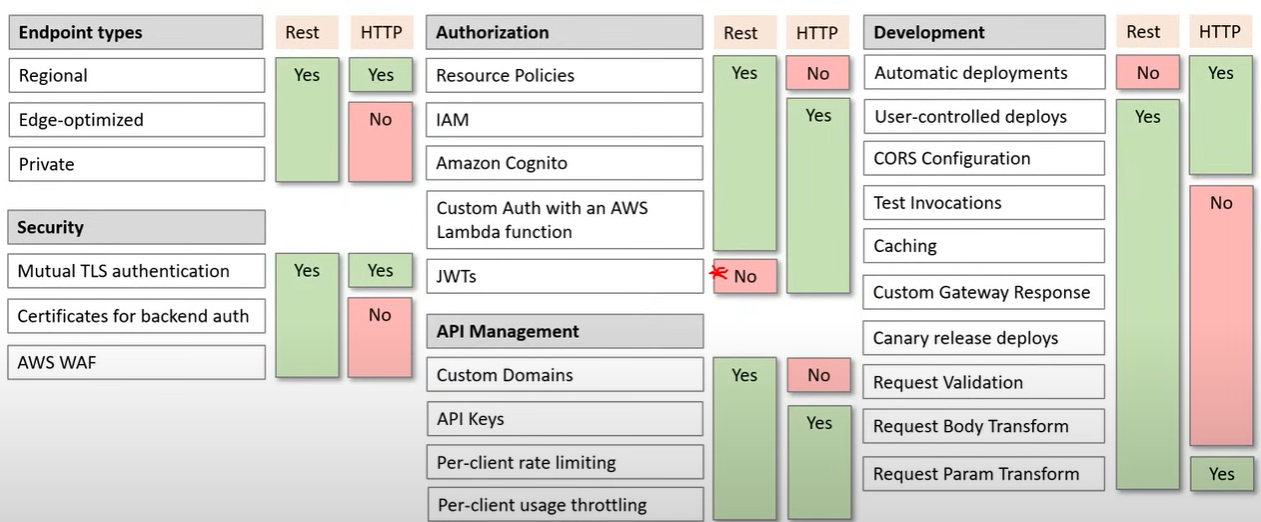
**Amazon Redshift** – a data warehouse (enables complex queries across all data and generates reports from multiple sources), two types of nodes – dense compute (DC) or dense storage (Ds), backups are enabled by default within a 1 day retention period – can be updated to up to 35 days, Redshift always attempts to maintain at least 3 copies of your data, is single-AS – to run multi-AZ you would have to run clones across the different AZs with the same inputs, a single node is 160gb in size and a multi-node is comprised of a leader node and multiple compute nodes, billed per hour for each node – excluding the lead node, Redshift most common use case is Business Intelligence,

**Amazon Athena** – interactive query service that makes it easy to analyze data directly from S3, uses SQL to query the S3 buckets, serverless

**AWS Glue** – serverless data integration service, AWS Glue Data Crawler is a tool that is used to analyze a targeted data source to determine its schema and generate AWS Glue Data Tables,

**Amazon API Gateway** – solution for creating secure APIs at any scale, three types of APIs for the API Gateway:

* Rest API – higher costs
* HTTP API – lower costs
* WebSockets API – for real-time use cases



**MongoDB** – open-source document database which stores JSON-like documents, data structure is BSON, scales horizontally, can be used as a file system,

**Amazon DocumentDB** – a NoSQL document database that is MongoDB compatible, cluster types: instance-based cluster or elastic cluster, clusters are deployed into a customer’s VPC,

**DynamoDB** - is a fully managed NoSQL database service that is ideal for high-performance applications that require low-latency data access. It uses a key-value pair and document data model to provide a simple approach to data storage. Automatically creates partitions as your data grows (for every 10 GB of data).

When you create a table, you have to define a Primary Key – this cannot be changed later. A sort key will determine how data should be sorted on a partition.

**Amazon Neptune** – highly available and durable graph database service

**Elastic Container Service** – is a fully managed container orchestration service that helps you easily deploy, manage, and scale containerized applications.

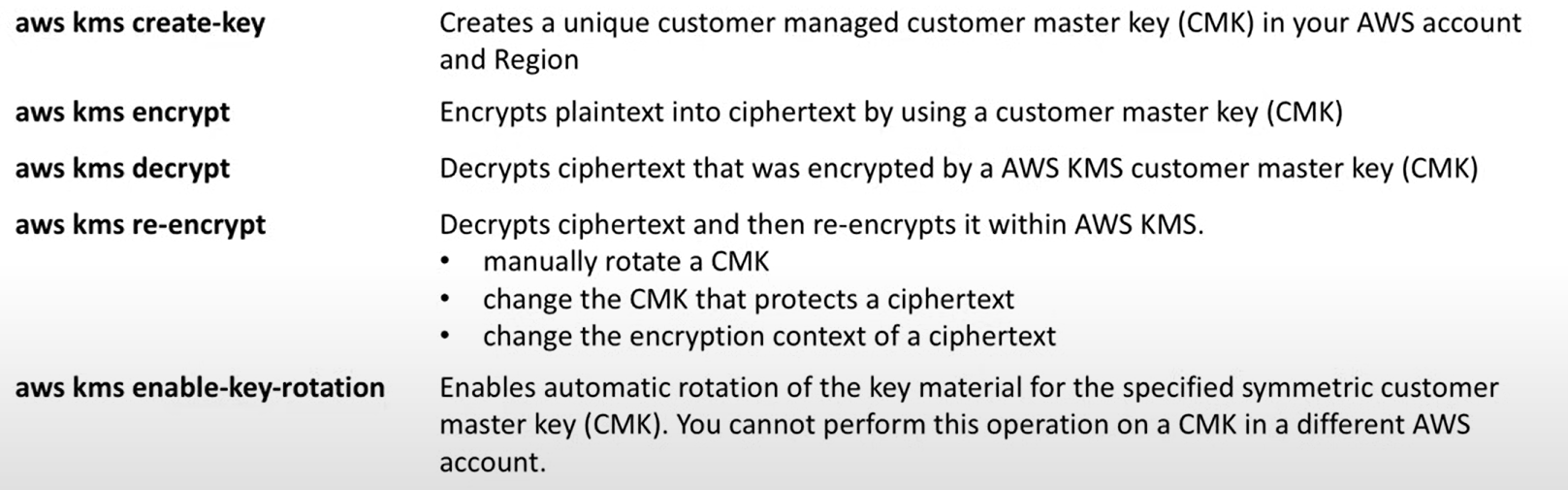
There are three layers in Amazon ECS:

* Capacity - The infrastructure where your containers run
* Controller - Deploy and manage your applications that run on the containers
* Provisioning - The tools that you can use to interface with the scheduler to deploy and manage your applications and containers

**EKS Cloud** – Elastic Kubernetes Services is a managed service that eliminates the need to have your own Kubernetes control plane on AWS

**Key Management Service** – create and manage encryption keys, it is a multi-tenant Hardware Security Module, most AWS services can just check the box for Encryption and then choose a KMS key,

Maine commands for AWS CLI



**ACM** – handles the complexity of creating and managing SSL/TLS certificates, it attaches to API Gateway, Cloudfront and Elastic Load Balancer,

**Amazon Cognito** – customer identity and access management,

**Amazon Detective** – analyzes, investigates and quickly identity the root cause of security findings or suspicious activities

**Amazon Inspector** - Amazon Inspector automatically discovers workloads, such as Amazon EC2 instances, containers, and Lambda functions, and scans them for software vulnerabilities and unintended network exposure.

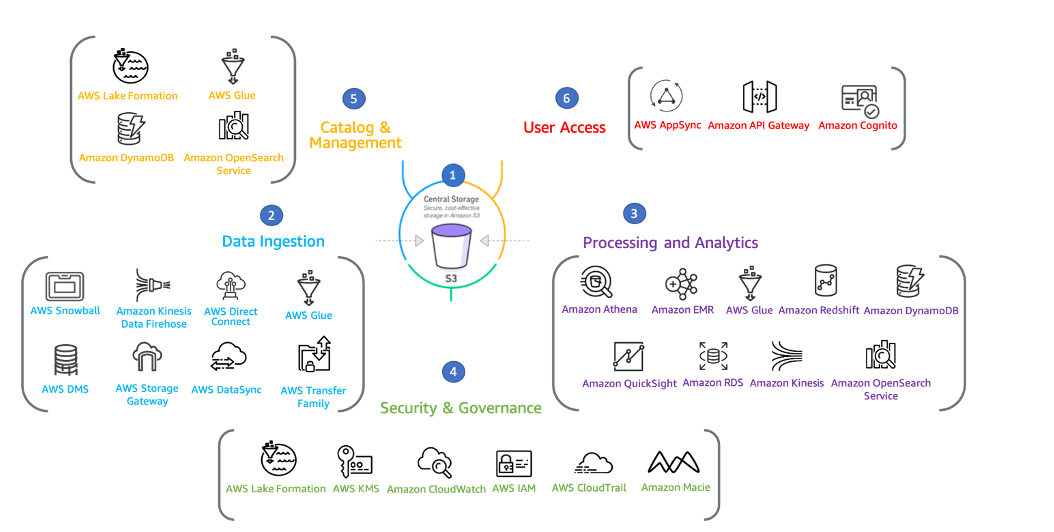
**Secrets Manager** – store and automatically rotate database credentials – RDS, Redshift, DocumentDB, rotation is performed via a Lambda function

**AWS Shield** – managed DDoS protection service that safeguards applications running on AWS

**AWS Artifact** – checking AWS compliance.

**AWS Storage Gateway** – connects on-premise software applications with cloud-based storage

Data Lake Example



Notes:

RDS Proxy doesn't support Aurora DBs.