AWS Certified Solutions Architect

STUDY NOTES

ASSOCIATE SAA-C02 EXAM PREP

PREPARED BY: NIZAR B. HAKEEM

JULY, 2020

www.hakeem.me

Contents

Resources

- AWS Certified Solutions Architect Study Guide, SYBEX – Second Edition by Ben Piper and David Clinton
- AWS Certified Solutions Architect Practice Tests, SYBEX – By Brett McLaughlin
- A Cloud Guru https://acloud.guru
- AWS Documentation
- AWS Whitepapers, example: AWS Well-Architected Framework https://d1.awsstatic.com/whitepapers/architecture/AWS_Well-Architected_Framework.pdf
- AWS Core Services Pages and Q&As.
- AWS official CSAA Exam Readiness Webinar
- Linkedin Learning Courses by Tom Carpenter and Lynn Langit

AWS Core Services - **Compute**

Elastic Compute Cloud (EC2)

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment.

Instance Types

There are currently more than 75 instance types, arranged in the following categories:

General Purpose	A1, T3, T3a, T2, M6g, M5, M5n, M4
Compute Optimized	C6g, C5, C5a, C5n, C4
Memory Optimized	R6g, R5, R5a, R5n, R4, X1e, X1, z1d
Accelerated Computing	P3, P2, Inf1, G4, G3, F1
Storage Optimized	I3, I3en, D2, H1

Amazon EC2 allows you to choose between Fixed Performance Instances (e.g. M5, C5, and R5) and Burstable Performance Instances (e.g. T3). Burstable Performance Instances provide a baseline level of CPU performance with the ability to burst above the baseline. You can accumulate CPU credits when your instance is underutilized that can be applied during high-demand periods in the form of higher CPU performance.

- The T2, M5, M4 and M3 classes provide a balance of memory and network resources.
- The C5, C4 and C3 classes are useful for CPU-intensive applications.
- The X1e, X1, R4, and R3 classes are best for high memory demand instances.
- The H1, I3 and D2 classes are optimized for storage access
- The P3, P2, G3 and F1 classes support specialty hardware such as GPU and FPGAs.
- I3en and C5n are bare metal instances. Bare metal means that the instance is running directly on the hardware instead of a hypervisor.

Important Points

- Amazon Machine Images (AMI) are template document that contains EC2 instance configuration, OS and application software to include on the root volume. There are four (4) kinds of AMIs: Amazon Quick Start AMIs, AWS Marketplace AMI, Community AMI, Private AMI.
- You can share images as AMIs or import VMs from your local infrastructure using AWS VM Import/Export tool.
- A particular AMI will be available in **only a single region**.
- The three primary details to specify after choosing the instance type are: geographic region, virtual private cloud (VPC), and tenancy model.

- Stopped instance that had been using a non-persistent public IP address will most likely be assigned a different IP address when it's restarted. If you need a persistent IP address that survives restarts, you will need to allocate an Elastic IP address (EIP).
- The default user to SSH to amazon AMI is "EC2-User"
- Bootstrapping provide code to be run on an instance at launch.
- VM import/export can be used to import existing machines into EC2.
- Host Recovery service restarts EC2 instances when a problem is detected or when a new host is available.
- Traffic monitoring copies network traffic from an elastic network interface ENI of an EC2 instance and sends it whenever you want it to go.

Placement Groups

When you launch a new EC2 instance, the EC2 service attempts to place the instance in such a way that all of your instances are spread out across underlying hardware to minimize correlated failures. You can use placement groups to influence the placement of a group of interdependent instances to meet the needs of your workload. Depending on the type of workload, you can create a placement group using one of the following placement strategies:

Cluster Placement Groups– packs instances close together inside an Availability Zone. This strategy enables workloads to achieve the low-latency network performance necessary for tightly coupled node-to-node communication that is typical of HPC applications.

Partition Placement Groups – spreads your instances across logical partitions such that groups of instances in one partition do not share the underlying hardware with groups of instances in different partitions. This strategy is typically used by large distributed and replicated workloads, such as Hadoop, Cassandra, and Kafka.

Spread Placement Groups – strictly places a small group of instances across distinct underlying hardware to reduce correlated failures.

Note that there is no charge for creating a placement group.

EC2 Pricing / Tenancy Models

There are five ways to pay for Amazon EC2 instances: **On-Demand**, **Reserved Instances**, **Spot Instances**, **Savings Plans,** and **Dedicated Hosts**.

On-Demand Instances

With On-Demand instances, you pay for compute capacity by the hour or the second depending on which instances you run. No longer-term commitments or upfront payments are needed. You can increase or decrease your compute capacity depending on the demands of your application and only pay the specified per hourly rates for the instance you use.

On-Demand instances are recommended for:

- Users that prefer the low cost and flexibility of Amazon EC2 without any up-front payment or long-term commitment
- Applications with short-term, spiky, or unpredictable workloads that cannot be interrupted
- Applications being developed or tested on Amazon EC2 for the first time

Reserved Instances

Reserved Instances provide you with a significant discount (up to 75%) compared to On-Demand instance pricing. In addition, when Reserved Instances are assigned to a specific Availability Zone, they provide a capacity reservation, giving you additional confidence in your ability to launch instances when you need them.

For applications that have steady state or predictable usage, Reserved Instances can provide significant savings compared to using On-Demand instances. See How to Purchase Reserved Instances for more information.

Reserved Instances are recommended for:

- Applications with steady state usage
- Applications that may require reserved capacity
- Customers that can commit to using EC2 over a 1 or 3 year term to reduce their total computing costs.

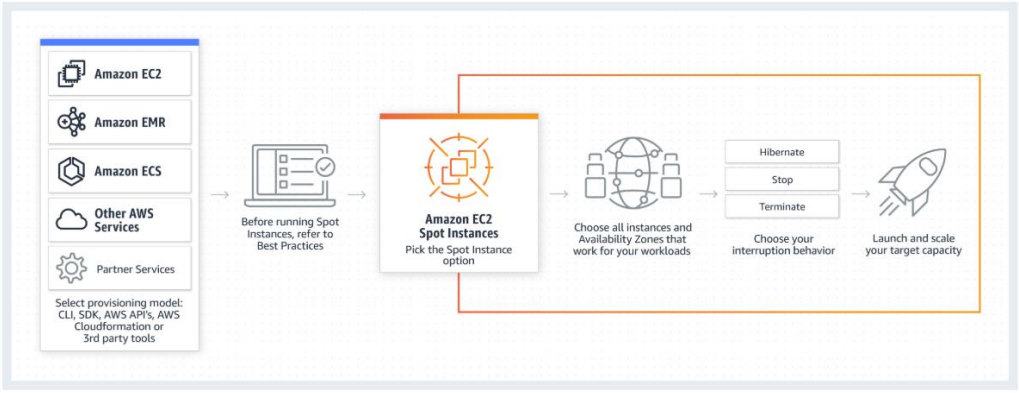
Spot Instances

Amazon EC2 Spot instances allow you to request spare Amazon EC2 computing capacity for up to 90% off the On-Demand price.

Spot instances are recommended for:

- Applications that have flexible start and end times
- Applications that are only feasible at very low compute prices
- Users with urgent computing needs for large amounts of additional capacity

How Spot Instances work?

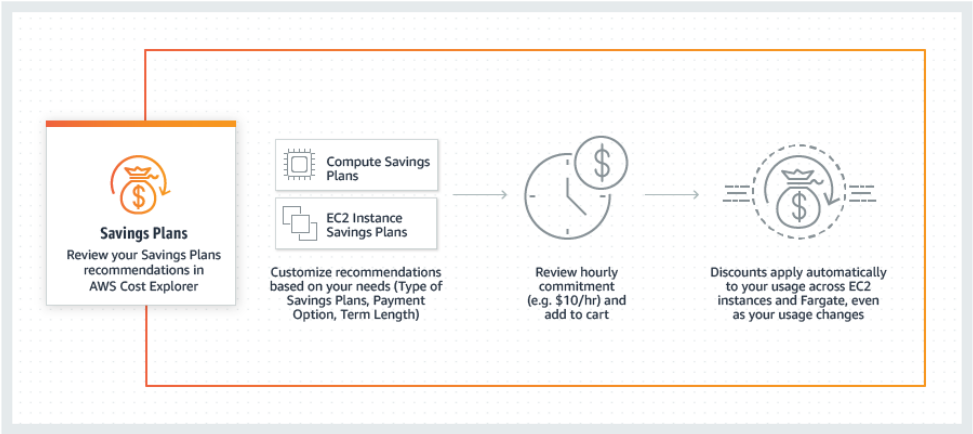


Combine Spot Instances with RIs and On-Demand Instances using EC2 Auto Scaling to optimize workload cost with performance. Click to see other the Best Practices to use Spot effectively.

Amazon Web Services customers have the ability to run Amazon Elastic MapReduce (EMR) clusters on Spot instances and significantly reduce the cost of processing vast amounts of data on managed Hadoop clusters.

Saving Plans

Savings Plans are a flexible pricing model that offer low prices on EC2 and Fargate usage, in exchange for a commitment to a consistent amount of usage (measured in $/hour) for a 1 or 3 year term.



Dedicated Host

A Dedicated Host is a physical EC2 server dedicated for your use. Dedicated Hosts can help you reduce costs by allowing you to use your existing server-bound software licenses, including Windows Server, SQL Server, and SUSE Linux Enterprise Server (subject to your license terms), and can also help you meet compliance requirements.

- Can be purchased On-Demand (hourly).

- Can be purchased as a Reservation for up to 70% off the On-Demand price.

Dedicate instance is different from **dedicated host instance**. A dedicated instance runs on physical machine and it is the only instance running on that machine. However, on restart it may be moved to a different physical machine. It also must be explicitly configured and it is not available in free tier.

Lambda

AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume.

With Lambda, you can run code for virtually any type of application or backend service - all with zero administration. Just upload your code and Lambda takes care of everything required to run and scale your code with high availability. You can set up your code to automatically trigger from other AWS services or call it directly from any web or mobile app.

AWS Lambda natively supports Java, Go, PowerShell, Node.js, C#, Python, and Ruby code, and provides a Runtime API which allows you to use any additional programming languages to author your functions.

Lambda Benefits

- No Servers to Manage
- Continuous Scaling
- Subsecond Metering (Charged for every 100 ms of code execution).
- Consistent Performance

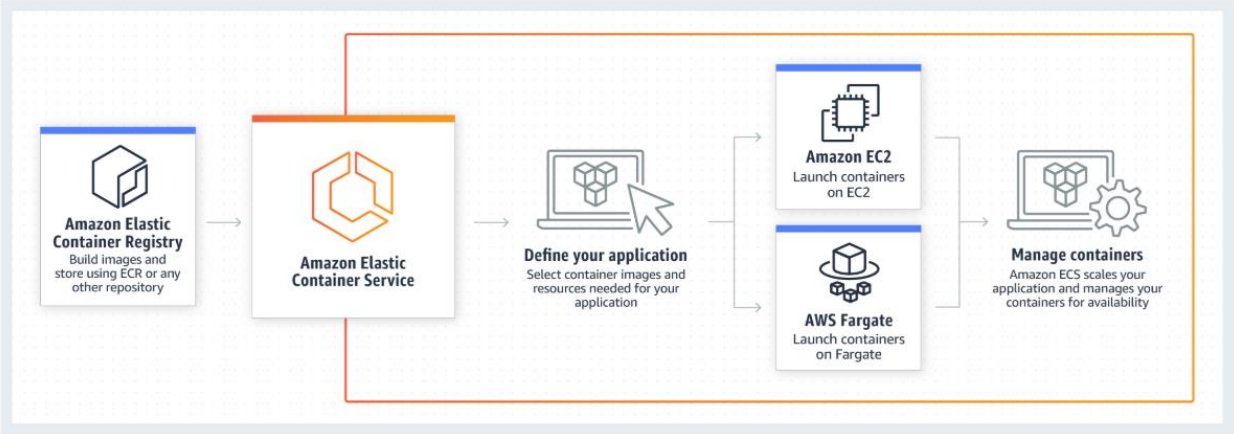
Lambda@Edge

Lambda@Edge allows you to run code across AWS locations globally without provisioning or managing servers, responding to end users at the lowest network latency. You just upload your code to AWS Lambda and configure your function to be triggered in response to Amazon CloudFront requests. The code is then ready to execute across AWS locations globally when a request for content is received, and scales with the volume of CloudFront requests globally.

Elastic Container Service (ECS) & AWS Fargate

Amazon Elastic Container Service (Amazon ECS) is a fully managed container orchestration service.

ECS is a great choice to run containers for several reasons. First, you can choose to run your ECS clusters using AWS Fargate, which is serverless compute for containers. Fargate removes the need to provision and manage servers, lets you specify and pay for resources per application, and improves security through application isolation by design. Second, ECS is used extensively within Amazon to power many services including Amazon.com's recommendation engine, ensuring ECS is tested extensively for security, reliability, and availability.



- ECS can be used to launch apps in AWS without deploying instances directly.
- A multi-tier application can use separate container for each tier.
- The concept of microservices is supported by ECS.

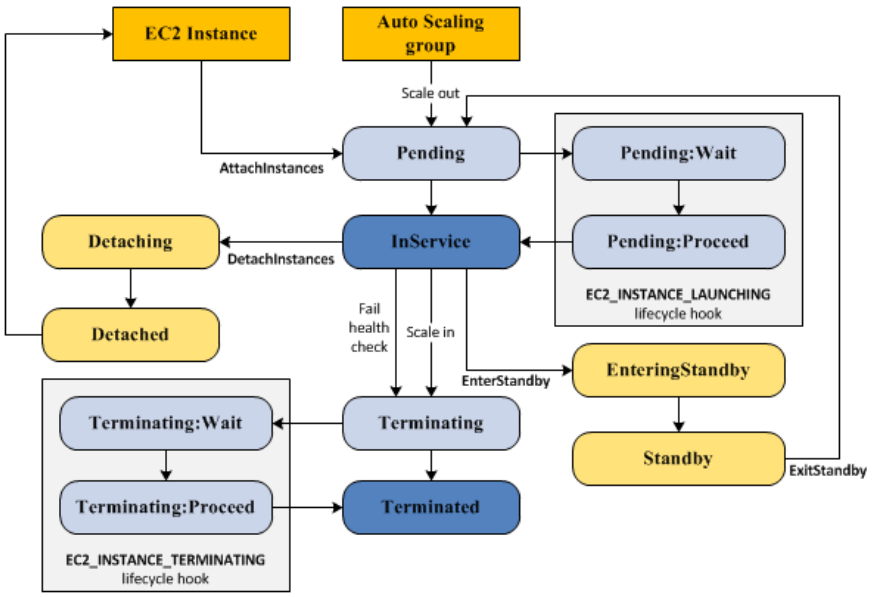
ECS Use Cases:

- **Hybrid deployment** with AWS Outposts service that allows you to manage containers on-premises.
- **Batch Processing** with AWS Batch that enables you to easily and efficiently run hundreds of thousands of batch jobs. A job is a unit of work executed by an AWS batch, or basically any script that is run in the AWS environment.
- **Machine Learning** with AWS Deep Learning Containers.
- **Web Application** By running on ECS, your web applications benefit from the performance, scale, reliability, and availability of the AWS and get out-of-the-box integration with other network and security services.

Auto Scaling

The service provides a simple, powerful user interface that lets you build scaling plans for resources including **Amazon EC2** instances and **Spot Fleets**, **Amazon ECS** tasks, **Amazon DynamoDB** tables and indexes, and **Amazon Aurora** Replicas. AWS Auto Scaling makes scaling simple with recommendations that allow you to optimize performance, costs, or balance between them

The following illustration shows the transitions between instance states in the Amazon EC2 Auto Scaling lifecycle.

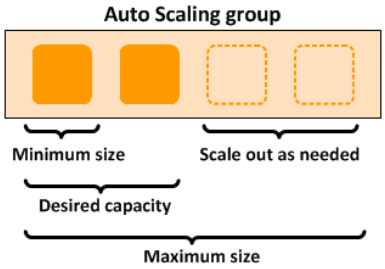


- Auto Scaling monitors the workload of instances and can add more instances or remove them as needed.
- Auto Scaling is free to use; however, you may incur costs for more instances, CloudWatch, and ELB load balancers.
- A launch configuration must be created in order to create an Auto Scaling Group. That can be done separately or as part of the Auto Scaling Group wizard.
- The four methods for creating an Auto Scaling group are:
    - Using Launch Template (Recommended)
    - Using Launch Configuration
    - Using EC2 Instance
    - Using Launch Wizard (within EC2 Console).

Auto Scaling - Amazon EC2

Launch or terminate Amazon EC2 instances in an Amazon EC2 Auto Scaling group. Amazon recommends that you create Auto Scaling groups from launch templates to ensure that you're getting the latest features from Amazon EC2.

The scaling policies that you define adjust the number of instances, within your minimum and maximum number of instances, based on the criteria that you specify.



Auto Scaling Components

- **Auto Scaling Groups**: Your EC2 instances are organized into *groups* so that they can be treated as a logical unit for the purposes of scaling and management. When you create a group, you can specify its minimum, maximum, and, desired number of EC2 instances. Auto Scaling groups can span multiple AZs.
- **Configuration Templates**: Your group uses a *launch template* or a *launch configuration* as a configuration template for its EC2 instances. You can specify information such as the AMI ID, instance type, key pair, security groups, and block device mapping for your instances.
- **Scaling Options**: Amazon EC2 Auto Scaling provides several ways for you to scale your Auto Scaling groups. For example, you can configure a group to scale based on the occurrence of specified conditions (dynamic scaling) or on a schedule.

Auto Scaling – Spot Fleet

Launch or terminate instances from an Amazon EC2 Spot Fleet, or automatically replace instances that get interrupted for price or capacity reasons.

Spot Fleet supports the following types of automatic scaling:

- **Target tracking scaling** – Increase or decrease the current capacity of the fleet based on a target value for a specific metric. This is similar to the way that your thermostat maintains the temperature of your home—you select temperature and the thermostat does the rest.
- **Step scaling** – Increase or decrease the current capacity of the fleet based on a set of scaling adjustments, known as step adjustments, that vary based on the size of the alarm breach.
- **Scheduled scaling** – Increase or decrease the current capacity of the fleet based on the date and time.

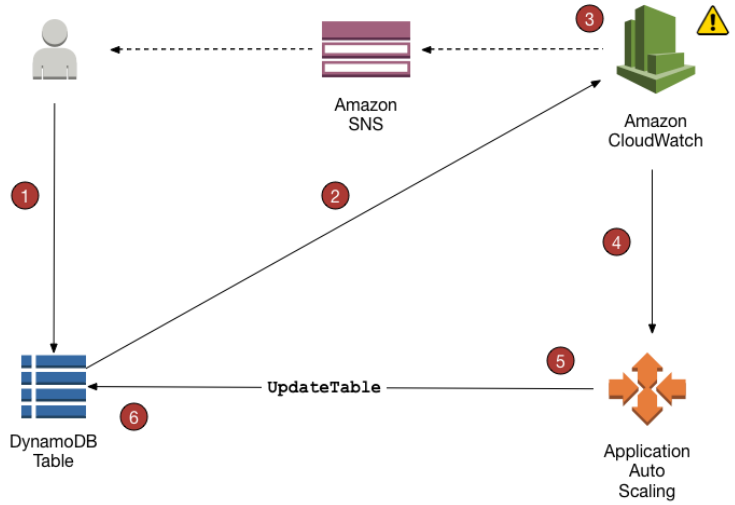
Auto Scaling - Amazon ECS

Adjust ECS service desired count up or down to respond to load variations. You can use CloudWatch metrics to scale out your service (add more tasks) to deal with high demand at peak times, and to scale in your service (run fewer tasks) to reduce costs during periods of low utilization. Amazon ECS Service Auto Scaling supports the same three type of automatic scaling: Target Tracking Scaling, Step Scaling and Scheduled Scaling.

Auto Scaling – DynamoDB

Enable a DynamoDB table or a global secondary index to increase its provisioned read and write capacity to handle sudden increases in traffic without throttling.

The following diagram provides a high-level overview of how DynamoDB auto scaling manages throughput capacity for a table.



Auto Scaling - Amazon Aurora

Dynamically adjust the number of Aurora Read Replicas provisioned for an Aurora DB cluster to handle sudden increases in active connections or workload.

Aurora Auto Scaling uses a scaling policy to adjust the number of Aurora Replicas in an Aurora DB cluster. Aurora Auto Scaling has the following components:

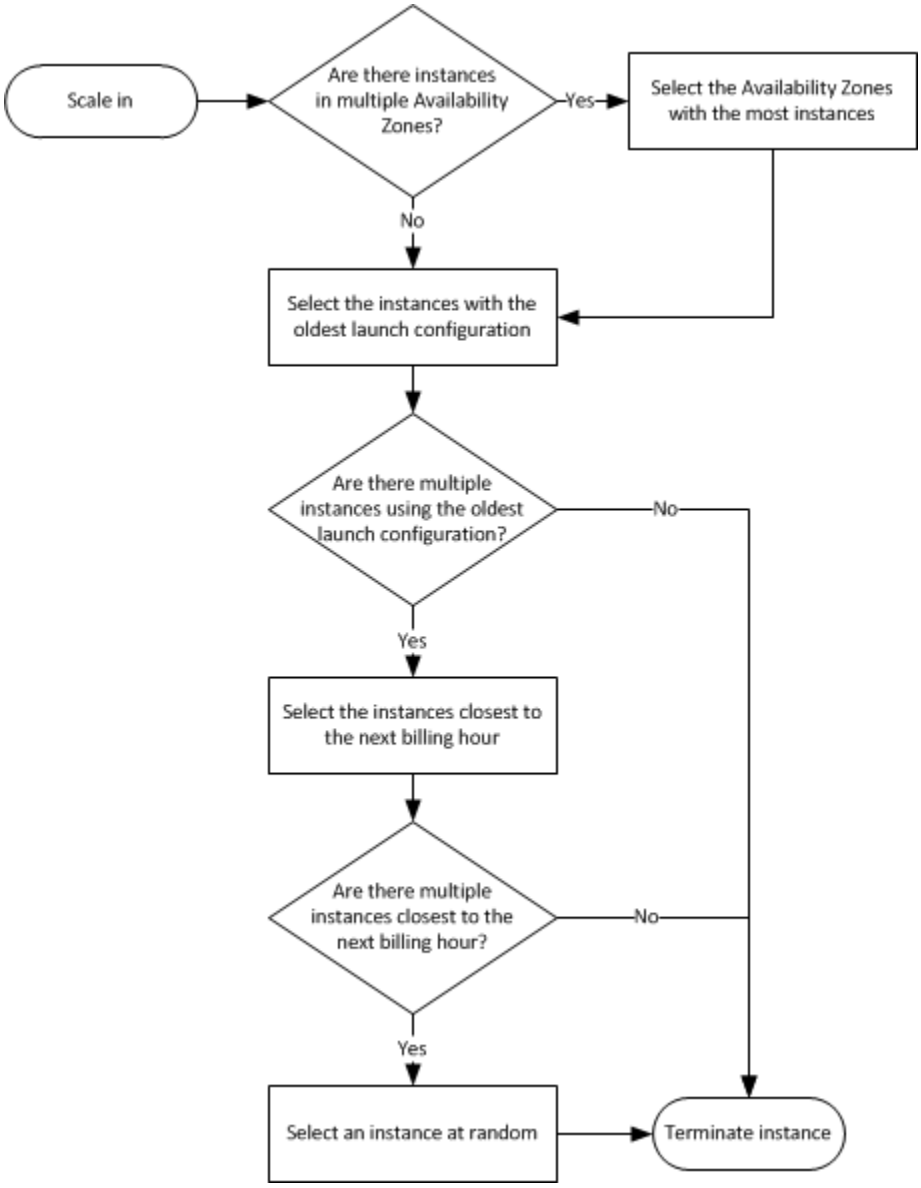
- A service-linked role
- A target metric
- Minimum and maximum capacity
- A cooldown period

Scaling In – Termination Policies Flowchart

How to decide which instance is to be scaled in (i.e.; removed)?

Look at three main components in order:

1- The AZ with the most instances.
2- The instance with the oldest launch configuration.
3- The instance closest to the next billing hour.



If you don't want to go with default termination policy described above, you can also build your custom termination policies by select one or more of the options:

- OldestInstance
- NewestInstance
- OldestLaunchConfiguration
- ClosestToNextInstanceHour

Elastic Load Balancing

- ELB is a highly available, elastic, secure, flexible, hybrid and monitored service.
- ELB types: Classic, Network and Application load balancer.
    - Classic load balancer is the legacy type and it is NOT recommended anymore. It supports SSL as well as TCP Load balancing.
    - Network load balancer is used to load balance requests at the TCP layer or Layer 4 of the OSI model.
    - Application load balancer works on Layer 7 (Application Layer) to process http/https request.
- ELB Supported Services: EC2, ECS, Auto Scaling, CloudWatch, Route 53.
- It is important to note that a load balancer CANNOT split traffic across multiple region, only across availability zones.

Elastic Beanstalk

- Using Elastic Beanstalk, you can create a server instances with the **Create New Environment** wizard.
- You cannot change the environment tier after creating an environment.
- You can use application platforms, such as .NET, Java, Node.js, Python and Ruby.
- Elastic Beanstalk stores your application files and, optionally, server log files in Amazon S3. If you are using the AWS Management Console, the AWS Toolkit for Visual Studio, or AWS Toolkit for Eclipse, an Amazon S3 bucket will be created in your account for you and the files you upload will be automatically copied from your local client to Amazon S3. Optionally, you may configure Elastic Beanstalk to copy your server log files every hour to Amazon S3. You do this by editing the environment configuration settings.

AWS Core Services - Networking

Virtual Private Cloud (VPC)

- A VPC is a logical construct in the cloud
- Connections to the VPC can be secured with VPN protocols.
- Subnets can be created within the VPC and made private or public.
- Multiple VPCs can be interconnected with VPC peering.
- All AWS accounts starts with a default VPC.
- Amazon recommends not deleting the default VPC.
- Multiple additional VPCs can be created with subnets in each.
- DHCP options for the VPC are configured in DHCP option sets.
- The DNS domain name can be configured in the option set.
- DHCP will be used to provide dynamic addresses where required within the VPC.

VPC Components

Amazon VPC comprises a variety of objects that will be familiar to customers with existing networks:

- **A Virtual Private Cloud**: A logically isolated virtual network in the AWS cloud. You define a VPC's IP address space from ranges you select.
- **Subnet**: A segment of a VPC's IP address range where you can place groups of isolated resources.
- **Internet Gateway:** The Amazon VPC side of a connection to the public Internet.
- **NAT Gateway**: A highly available, managed Network Address Translation (NAT) service for your resources in a private subnet to access the Internet.
- **Virtual private gateway**: The Amazon VPC side of a VPN connection.
- **Peering Connection**: A peering connection enables you to route traffic via private IP addresses between two peered VPCs.
- **VPC Endpoints**: Enables private connectivity to services hosted in AWS, from within your VPC without using an Internet Gateway, VPN, Network Address Translation (NAT) devices, or firewall proxies.
- **Egress-only Internet Gateway**: A stateful gateway to provide egress only access for IPv6 traffic from the VPC to the Internet.

Other Notes on VPC

Please note the following about Amazon VPC:

- You can have up to five (5) nondefault Amazon VPCs per AWS account per AWS Region.
- You can have up to four (4) secondary IP ranges per Amazon VPC.
- You can create up to two hundred (200) subnets per Amazon VPC.
- You can have up to five (5) Amazon VPC Elastic IP Addresses per AWS account per AWS Region.
- Amazon reserves the first four (4) IP addresses and the last one (1) IP address of every subnet for IP networking purposes.

Endpoints

- One VPC may need to provide access for its instances to a service in another VPC and this can be accomplished with Endpoints.
- Endpoints, in AWS, are not the same as endpoints in common local networking terminology. They are more like proxies to services.
- With Endpoints, services are specified based on region and services name.
- Endpoints allows for private connection to public services (ex: DynamoDB), without traversing the internet.
- **Gateway Endpoints** rely on creating entries in a route table and pointing them to private endpoints used for S3 or DynamoDB. **Interface Endpoints** use AWS PrivateLink and leverages the new Network Load Balancer capabilities.

  EXAM TIP: Know which services use interface endpoints and gateway endpoints. The easiest way to remember this is that Gateway Endpoints are for Amazon S3 and DynamoDB only.

VPC Peering

- Connects one VPC to another.
- VPC peering is not transitive. Meaning, if VPC1 is peered with VPC2, and VPC2 is peered with VPC3, that DOESN'T mean that VPC1 and VPC3 are peered.
- To initiate VPC, owner role is required.
- IP CIDR blocks in each VPC of the peered VPCs must NOT overlap.
- Routing tables modification may be required as well as security groups modification.
- Steps to create a VPC peering between two VPCs:
  - Create a VPC peer request in the originating VPC first.
  - Access the VPC request in the target VPC to create the peer connection.
  - If all VPCs must be connected (peered) with all other VPCs, a peer must be created between each pair of VPCs.

VPN Gateways (VPG and CGW)

- Gateways are effectively VPN endpoints that connect local networks to the VPC.
- VPG is implemented on AWS side and it is the VPN concentrator.
- Customer Gateway (CGW) is a physical or software application implemented on the customer side (on-premises).
- AWS hardware VPN, AWS Direct Connect, VPN CloudHub and Software VPN are all considered as alternative solutions to have a VPN connection with AWS.
- Split-tunnel give flexibility for routing traffic across the VPN, specifically for traffic going directly out to the internet.
- AWS has now enabled certificates for authentication to the VPN instead of pre-shared keys and other authentication methods.

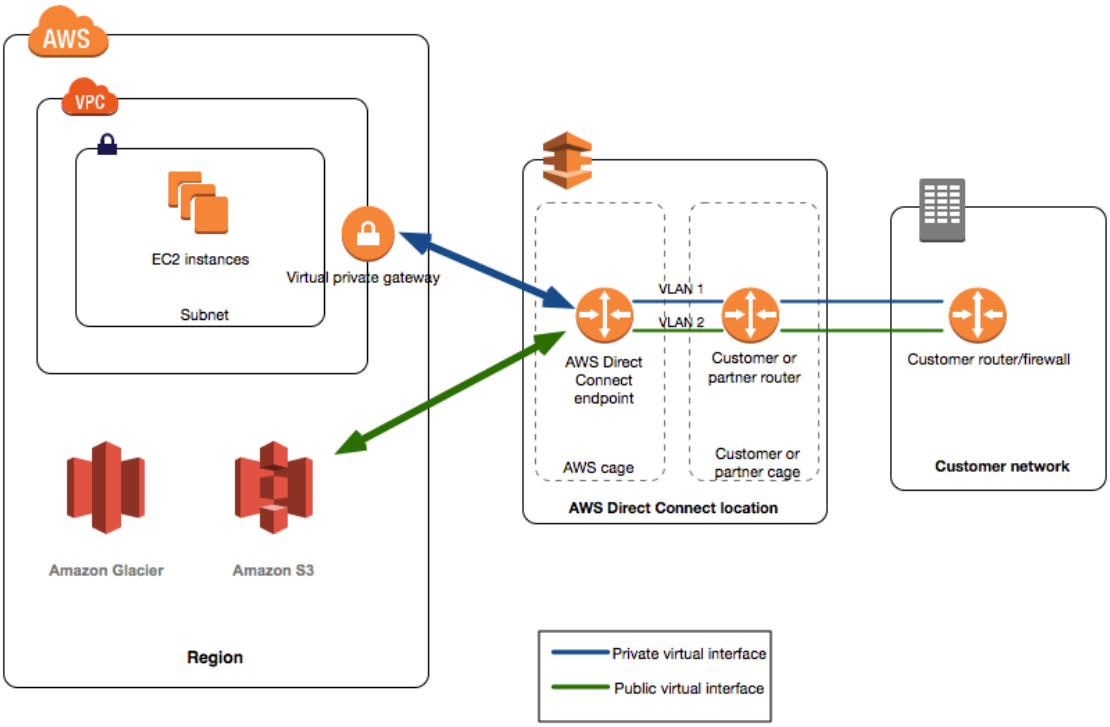
Elastic IP Addresses (EIPs) and Elastic Network Interfaces (ENI)

- EIPs can be moved between instance in the same region only.
- Elastic Network Interface (ENIs) use the EIPs
- Remember, when creating an EIP, it must be release to remove charges and it will not be released automatically.
- Multiple ENIs connected to a single instance allows dual-homing (an instance existing on more than one home = subnet). Example public and private.
- Each ENI is associated with a subnet with the VPC just as a physical network interface would be associated with a subnet on a local network.

AWS Direct Connect

AWS Direct connect is an alternative to VPN using above-mentioned gateways. Direct Connect bypasses traditional Internet Service Provider (ISP) Internet connection and connects straight into AWS (through AWS Partners Network APN).

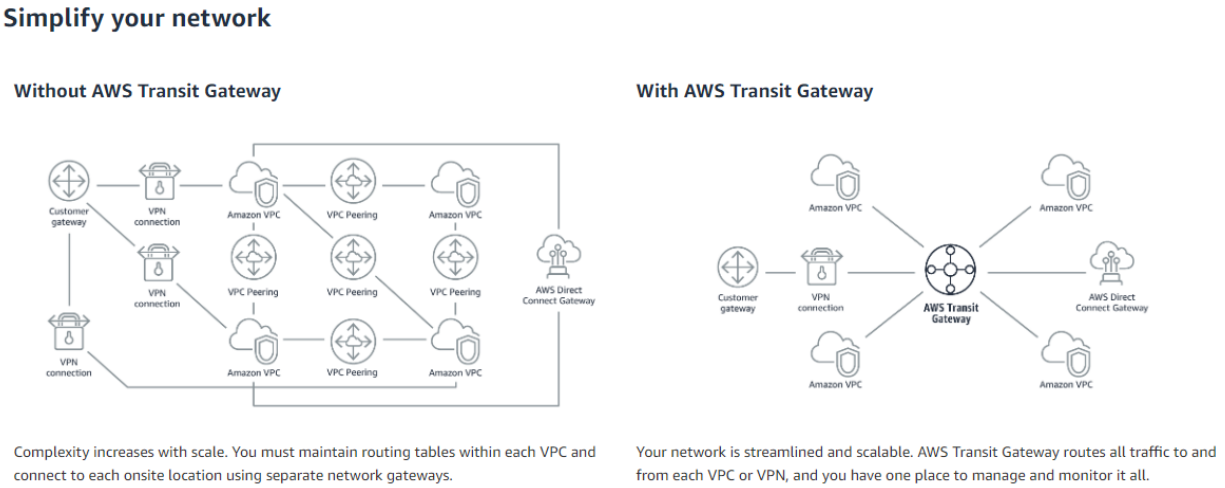
The following diagram shows how AWS Direct Connect interfaces with your network.



AWS Transit Gateway

To resolve the complexity of VPC peering for multiple VPC, AWS came up with AWS Transit Gateway in the late 2018s. AWS Transit Gateway connects VPCs and on-premises networks through a central hub. This simplifies your network and puts an end to complex peering relationships. It acts as a cloud router – each new connection is only made once.

As you expand globally, inter-Region peering connects AWS Transit Gateways together using the AWS global network. Your data is automatically encrypted, and never travels over the public internet. And, because of its central position, AWS Transit Gateway Network Manager has a unique view over your entire network, even connecting to Software-Defined Wide Area Network (SD-WAN) devices.



Route 53

Route 53 is more than just a basic DNS service. Route 53 focuses on 4 distinct areas:

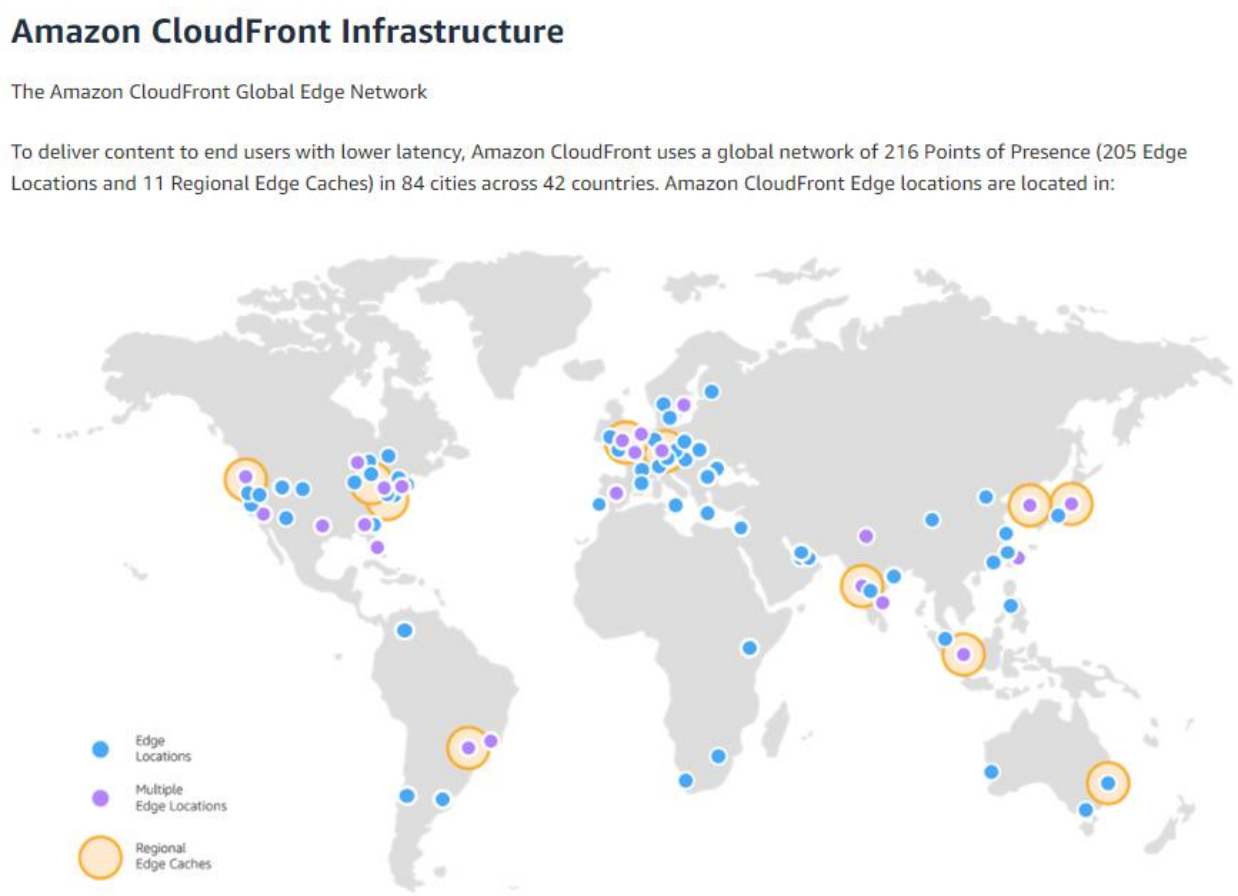
- Domain Registration
- DNS management
- Availability monitoring (health checks)
- Traffic management via routing policies

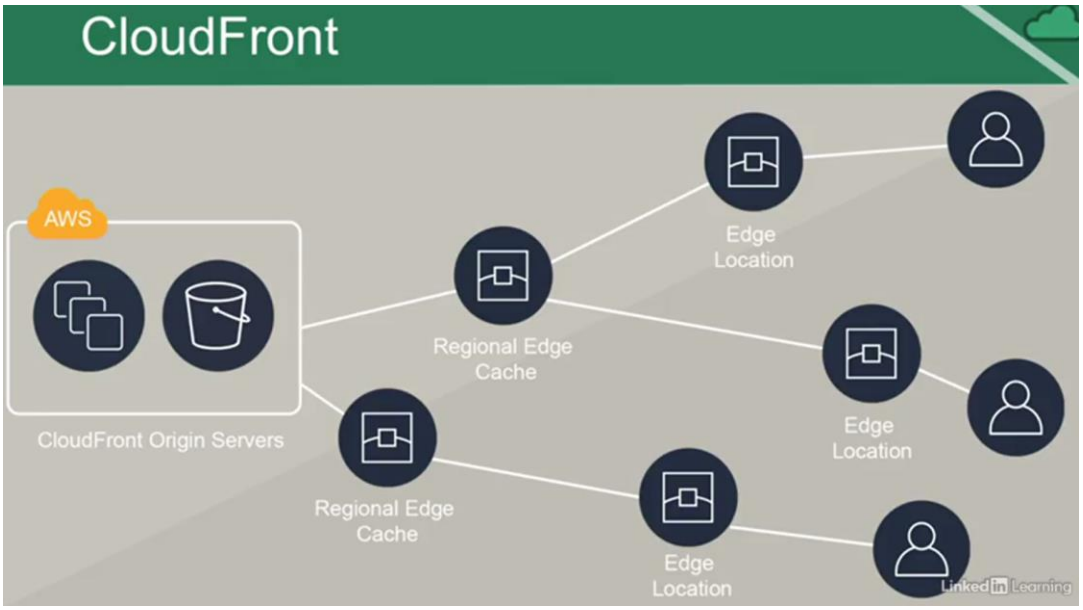
Traffic management uses the following routing policies to manage traffic:

- Weighted Routed
- Latency Routing
- Failover Routing
- Geolocation Routing
- Multivalue Answer (based on combined health checks)

CloudFront

Cloudfront is Amazon's Content Delivery Network (CDN) solution that distributes content in edge locations all over the globe. The below map give an idea about the 216 points of presence around the world.





CloudFront use cases:

- Accelerate Static Website Content Delivery
- Serve Video On-Demand or Live Streaming Video
- Encrypt Specific Fields Throughout System Processing
- Customize at the Edge (ex: translation)
- Serve Private Content by using Lambda@Edge Customizations

CloudFront Content Source examples:

- S3 buckets
- MediaPackage channel: that package the media based on consumer's device be it a smart phone, tablet, PC or smart TV so that the consumer gets the media in the right format.
- HTTP/HTTPS Server on EC2 Instance or Elastic load balancer.

AWS Global Accelerator

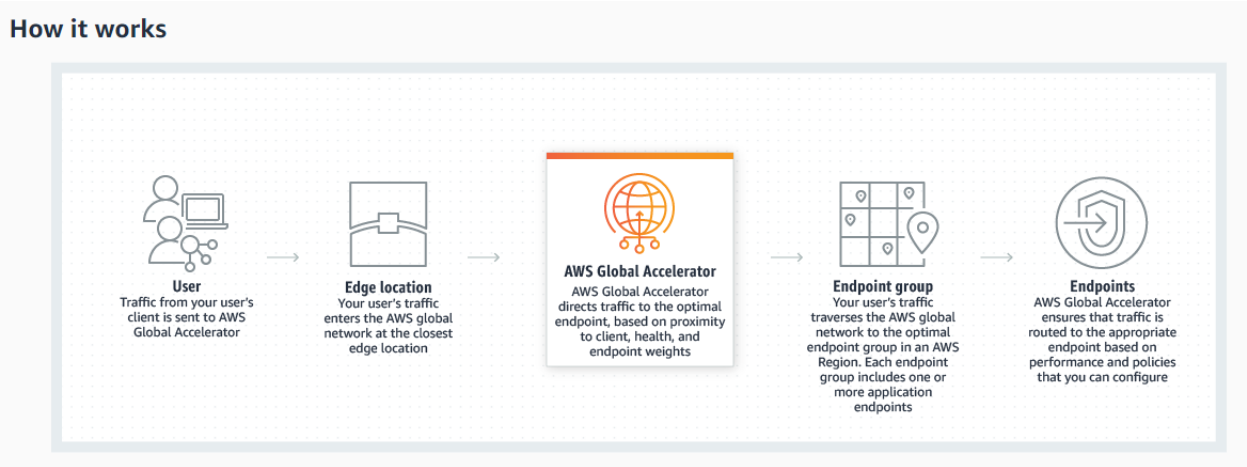
AWS Global Accelerator is a networking service that helps you improve the availability and performance of the applications that you offer to your global users.

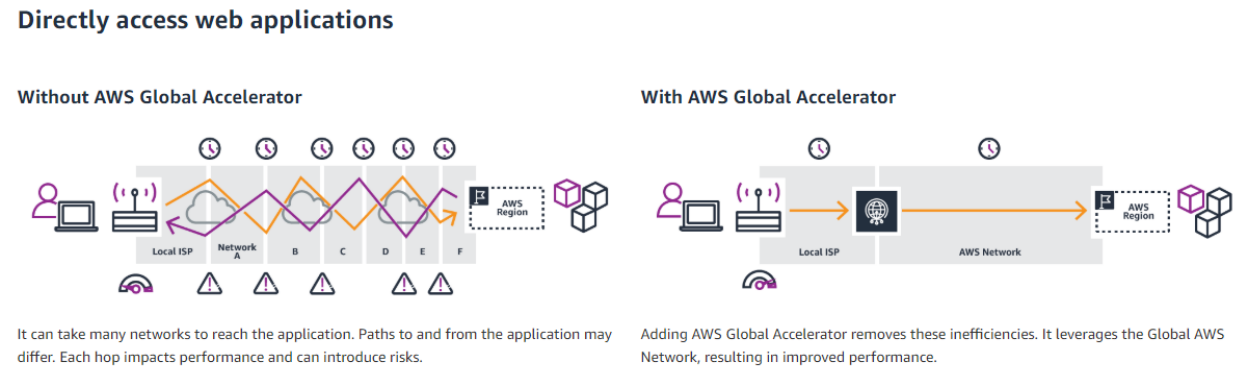
- AWS Global Accelerator provides static IP addresses that provide a fixed entry point to your applications and eliminate the complexity of managing specific IP addresses for different AWS Regions and Availability Zones.
- AWS Global Accelerator always routes user traffic to the optimal endpoint based on performance, reacting instantly to changes in application health, your user's location, and policies that you configure.
- You can test the performance benefits from your location with a speed comparison tool
- AWS Global Accelerator's static IP addresses make it easy to move endpoints between Availability Zones or AWS Regions without needing to update your DNS configuration or change client-facing applications. You can use static IP addresses from the Amazon IP address pool or you **can bring your own IP addresses (BYOIP)** to AWS Global Accelerator.

By using AWS Global Accelerator, you can:

- Associate the static IP addresses provided by AWS Global Accelerator to regional AWS resources or endpoints, such as Network Load Balancers, Application Load Balancers, EC2 Instances, and Elastic IP addresses. The IP addresses are anycast from AWS edge locations so they provide onboarding to the AWS global network close to your users.
- Easily move endpoints between Availability Zones or AWS Regions without needing to update your DNS configuration or change client-facing applications.
- Dial traffic up or down for a specific AWS Region by configuring a traffic dial percentage for your endpoint groups. This is especially useful for testing performance and releasing updates.

- Control the proportion of traffic directed to each endpoint within an endpoint group by assigning weights across the endpoints.





NAT Devices and NAT Gateways

- NAT stands for Network Address Translation it translates between private IP address and public IP addresses.
- An EIP is associated with the NAT instance for the public-facing side.
- Instances in the private subnet of the VPC use the NAT to connect to the internet.
- NAT can be implemented using a dedicated NAT instance or using an AWS NAT Gateway.

Flow Logs

- Equivalent to Netflows in a traditional network.
- Flow logs allow you to log traffic passing through your AWS network.
- Flow logs can be created on network interfaces, VPCs, and on subnets.
- Flow logs store the logs in S3 and CloudWatch. Thus, using flow logs incurs cost.

Simple Storage Service (S3)

Amazon S3 is object storage built to store and retrieve any amount of data from anywhere on the Internet. It's a simple storage service that offers an extremely durable, highly available, and infinitely scalable data storage infrastructure at very low costs.

Main Features

- Prefixes and Delimiters are used to create a folders-like structures.
- Storage Classes (Standard, S3-IA, S-RR, Glacier), choosing the right class is based on performance and cost.
- Object Lifecycle Management
- Encryption
- Versioning (Once enabled it can't be disabled, it can only be suspended).
- MFA Delete
- Mutli-part upload.
- Range GETs
- Cross-Region Replication (starts replicating from the point of enabling it. It doesn't replicate old objects.)
- Logging, not-enabled by default.
- Events notifications

General Notes

- S3 Bucket names must be globally unique across all Amazon S3 (not just your account).
- Intelligent Tiering using lifecycle rules to automate data transition.
- Object locking to enable WORM (write once read many). Can only be enabled at time of bucket creation.
- Batch operations create jobs to enable automatic actions.

Amazon S3 Access Points

Q: What is Amazon S3 Access Points?

Today, customers manage access to their S3 buckets using a single bucket policy that controls access for hundreds of applications with different permission levels.

Amazon S3 Access Points simplifies managing data access at scale for applications using shared data sets on S3. With S3 Access Points, you can now easily create hundreds of access points per bucket, representing a new way of provisioning access to shared data sets. Access Points provide a customized path into a bucket, with a unique hostname and access policy that enforces the specific permissions and network controls for any request made through the access point.

Q: How do S3 Access Points work?

Each S3 Access Point is configured with an access policy specific to a use case or application, and a bucket can have hundreds of access points. For example, you can create an access point for your S3 bucket that grants access for groups of users or applications for your data lake. An Access Point could support a single user or application, or groups of users or applications, allowing separate management of each access point. Each access point is associated with a single bucket and contains a network origin control, and a Block Public Access control. For example, you can create an access point with a network origin control that only permits storage access from your Virtual Private Cloud, a logically isolated section of the AWS Cloud. You can also create an access point with the access point policy configured to only allow access to objects with a defined prefix, such as “finance”.

Because each access point contains a unique DNS name, you can now address existing and new buckets with any name of your choice that is unique within the AWS account and region. Using access points that are restricted to a VPC, you can now have an easy, auditable way to make sure S3 data stays within your VPC. Additionally, you can now use AWS Service Control Policies to require any new access point in their organization to be restricted to VPC only access.

Glacier

- S3 stores objects in buckets, while Glacier stores archives in vaults.
- There are three retrieval methods where the user have to balance cost of retrieval vs. waiting time:
    - **Expedited access** provides the files within **3-5 minutes**.
    - **Standard access** provides the files within **3-5 hours**.
    - **Bulk access** provides the files within **5-12 hours**
- A Single AWS account can create up to 1,000 vaults per region.
- Only empty vaults can be deleted.
- Glacier Supports multipart upload of archives, so a large archive is not required to be uploaded in a single action.
- Glacier is the most cost-effective storage class among all S3-storage-classes. However, the cost might go up and suppress the other classes if the customer retrieves more the 5% of the stored data monthly.

Performance across the S3 Storage Classes

	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier	S3 Glacier Deep Archive
Designed for durability	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
Designed for availability	99.99%	99.9%	99.9%	99.5%	99.99%	99.99%
Availability SLA	99.9%	99%	99%	99%	99.9%	99.9%
Availability Zones	≥3	≥3	≥3	1	≥3	≥3
Minimum capacity charge per object	N/A	N/A	128KB	128KB	40KB	40KB
Minimum storage duration charge	N/A	30 days	30 days	30 days	90 days	180 days
Retrieval fee	N/A	N/A	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved
First byte latency	milliseconds	milliseconds	milliseconds	milliseconds	select minutes or hours	select hours
Storage type	Object	Object	Object	Object	Object	Object
Lifecycle transitions	Yes	Yes	Yes	Yes	Yes	Yes

Elastic Block Store (EBS)

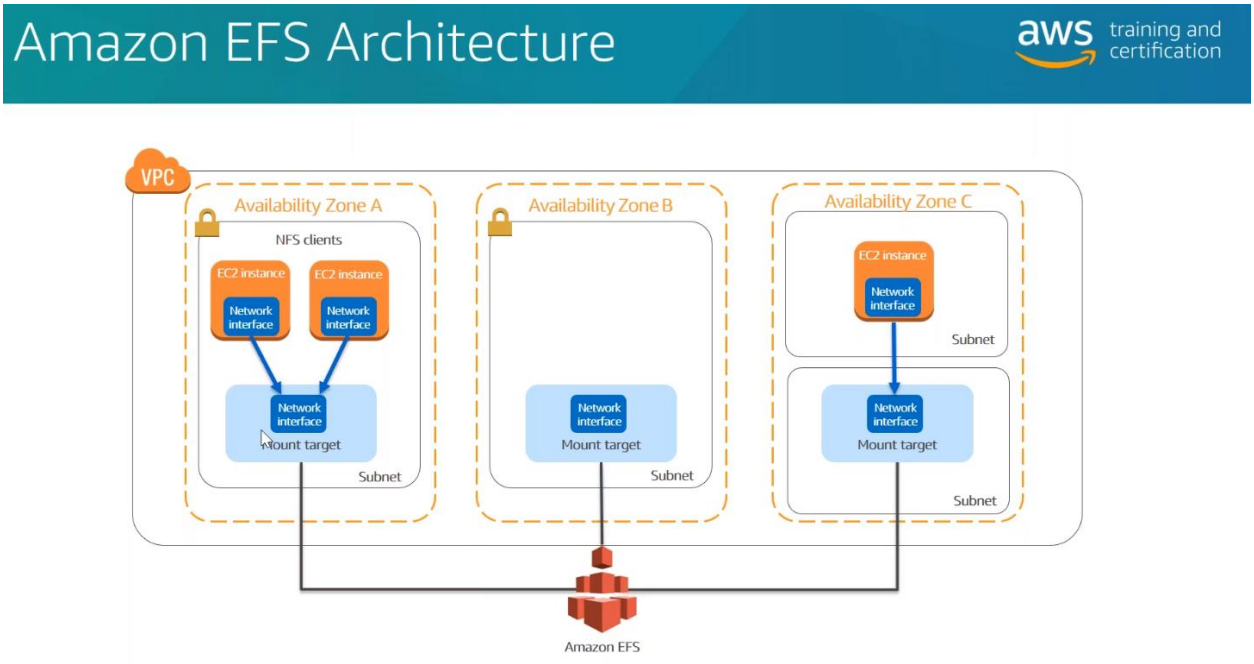
- Used for durable storage in EC2 instances. It can be connected to one-instance only at a time.
- Block-level storage from one AWS service to another.
- You can attach Elastic Block Store to your EC2 instance just as you would use hard drives or flash drives with your physical server. There are currently four EBS volume types: General Purpose SSD (gp2), Provisioned IOPS SSD (io1), Throughput Optimized HDD (st1), Cold HDD (Sc1).

	General Purpose SSD (gp2)	Provisioned IOPS SSD (io1)	Throughput Optimized HDD (st1)	Cold HDD (sc1)
Use Cases	Boot volumes, low-latency interactive applications, dev, test.	I/O intensive NoSQL and relational databases.	Big data, data warehouses, log processing.	Large volumes of data that is infrequently accessed
Volume Size	1 GiB - 16 TiB	4 GiB - 16 TiB	500 GiB - 16 TiB	500 GiB - 16 TiB
Max. IOPS/Volume	10,000	32,000	500	250
Max. Throughput/Volume	160 MiB/s	500 MiB/s	500 MiB/s	250 MiB/s
Dominant Performance Attribute	IOPS	IOPS	MiB/s	MiB/s

- **PRO TIP**: It is very important to know that if you use SSD storage as your EBS and you want to take advantage of the performance capabilities, you will have to use an EBS-optimized EC2 instance. If you don't, you will be paying for SSD but not getting the SSD performance. So make sure you are using the right kind of instance.
- To protect EBS data:
    - Use snapshots for backup and migration.
    - Volume recovery by attaching volumes from one instance to another.
    - Use encryption methods.

Elastic File System (EFS)

- EFS is like NAS within the cloud for the cloud.
- EFS is shareable (used by multiple instances at the same time).
- Hierarchical by nature
- Accessed using NFSv4
- EFS is NOT supported on Windows instances. Only Linux.



Amazon FSx

A cost-optimized, fully managed File System that allows creation of file systems either windows-based file system or the open-source file system Lustre (the high performance file system).

When setting up FSx, adding a file system name is optional, but makes it easier to manage.

FSx can be deployed in one or more AZs.

In FSx, windows authentication through Active Directory can be self-managed or managed by AWS.

Storage Gateway

The storage gateway connects on-premises software appliances with cloud-based storage.

File Gateway

Store files as objects in Amazon S3, with a local cache for low-latency access to your most recently used data.

Uses Network File system NFS and supports windows.

Volume Gateway

Block storage in Amazon S3 with point-in-time backups as Amazon EBS snapshots.

Uses SCSI over IP

Comes is two flavors: Cached Volumes and Stored Volumes

Tape Gateway

- Backup your data to Amazon S3 and archive in Amazon Glacier using your existing tape-based processes.
- Tape Gateway can be used with systems that only allow for backing up to tapes.
- Tape Gateway can be configured as public or VPC (private).
- A Virtual Tape Library (VTL) is a library of backup “tapes” that are actually just objects stored in an S3 bucket, usually in a S3 class called Glacier Deep Archive.

AWS Core Services - Databases

AWS supports both hosted databases (managed by AWS) and custom databases (running on EC2 instance).

AWS Relational Database Service (RDS) provide managed database services for relational. While Dynamo DB is the managed non-relational database solution from AWS.

When using a custom instance, you must install the instance and then install the appropriate database service on that instance.

Relational Database Service (RDS)

I am not a SQL guy, so I would like to start with some Relational Databases terminology.

- **Rows** may be called *tuples*
- **Columns** may be called *attributes* or *properties*
- **Tables** may be called *relations*, *entities*, *objects*
- **Views** and **results** are generated from SQL queries.
- Each table has a **Primary key** used to uniquely identify each record in that table.
- Tables are linked based on primary keys and **Foreign key.** A **Foreign key** is a primary key in the "other" table, in the context of linking current table (primary key) and other tables (foreign key).
- **Normalization** is the process for evaluating and correcting structures. It determines the best assignments of attributes to entities. Normalization work through a series of stages called normal forms.

RDS Hosting Methods

There are two methods of database hosting when it comes to relational databases: 1- Using EC2 Instance-based, or 2- AWS Service-Based

- EC2 Instance-based hosting requires more effort, it will basically involve launching the instance, installing database service, allowing appropriate ports in security groups then connecting to database.
- While in AWS managed service-based you only have to launch the database and connect to it.

Instance-Based Benefits	Service-Based Consideration
Complete control	Less control
Manual performance management	Automatic performance management
Manual updates	Automatic updates

RDS Databases

AWS RDS support the following Relational databases:

- **MySQL**: MySQL is designed for Online Transaction Processing OLTP. It offers two storage engines: MyISAM and InnoDB. However, InnoDB is the only one compatible with RDS-managed automatic backups.
- **Oracle:** Is one of the most widely developed relational database management systems. In AWS, Oracle is the only RDS that allow for BYOL (bring your own license).
- **PostgreSQL**: PostgreSQL advertises itself as the most Oracle-compatible open source database. It is a good choice when you want to use applications developed for Oracle but you want to keep your costs down.
- **Aurora**, **Aurora MySQL** and **Aurora PostgreSQL** will be discussed in details below.
- **Microsoft SQL**: RDS offers multiple editions of Microsoft SQL Server (2012, 2014, 2016, and 2017) including Express, Web, Standard and Enterprise
- **MariaDB**: MariaDB is a drop-in binary replacement for MySQL. MariaDB supports the XtraDB and InnoDB storage engines, but AWS recommends using the latter for maximum compatibility with RDS.

Aurora

- A Relational DB created by Amazon and built for the cloud.
- Optimized for Online Transaction Processing OLTP.
- It has very fast writes.
- MySQL-compatible database system.
- Increased performance by five times over MySQL and three times over PostgreSQL.
- Aurora databases starts initially at 10 GB, and can scale up in 10 GB increments.
- Maximum database size is 64 TB
- Maximum compute resources: 32 CPUs and 244 GiB RAM. Memory and compute resources are modified by changing DB instance class.
- Aurora is highly available by default: 2 DB copies in each AZ in a given region. That's a minimum of 6 copies per region.
- Write capability continues with up to two copies lost.
- Read capability continues with up to three copies lost.
- Automated backups are always enabled on Amazon Aurora DB Instances. Backups do not impact database performance. And administrator can take DB snapshots if needed.
- Amazon Aurora Global Database is a feature that allows a single Amazon Aurora database to span multiple AWS regions.
- AuroraDB is the only RDS that cannot be implemented under Free Tier.

- Replicas: Aurora supports up to 15 replicas with automatic failover, versus 5 read replicas limit for MySQL with no automatic failover. However, with Amazon Aurora MySQL, you can also create cross-region MySQL Read Replicas based on MySQL's binlog-based replication engine. The below table compares the main features:

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
Replica location	In-region	Cross-region
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

- If needed, a read replica can be promoted to be the new primary instance from the RDS console.
- **Amazon Aurora Multi-Master** is a new feature of the Aurora MySQL-compatible edition that adds the ability to scale out write performance across multiple Availability Zones, allowing applications to direct read/write workloads to multiple instances in a database cluster and operate with higher availability.
- **In-transit encryption**: Amazon Aurora uses SSL (AES-256) to secure the connection between the database instance and the application.
- **Encryption at rest**: Amazon Aurora allows you to encrypt your databases using keys you manage through AWS Key Management Service (KMS). On a database instance running with Amazon Aurora encryption, data stored at rest in the underlying storage is encrypted, as are its automated backups, snapshots, and replicas in the same cluster. However, it is important to know that encryption must be done at creation time. Encrypting an existing unencrypted Aurora instance is not currently supported.
- **Amazon Aurora Serverless** is an on-demand, autoscaling configuration for the MySQL-compatible and PostgreSQL-compatible editions of Amazon Aurora. An Aurora Serverless DB cluster automatically starts up, shuts down, and scales capacity up or down based on your application's needs. Aurora Serverless provides a relatively simple, cost-effective option for infrequent, intermittent, or unpredictable workloads.

Read Replicas

Scaling horizontally entails creating additional database instances called read replicas. All database engines except for Oracle and Microsoft SQL Server support read replicas. Aurora exclusively supports a special type of read replica called Aurora replica.

You can have up to 5 read replicas for RDS databases and up to 15 Aurora replicas.

In the event of the master instance failing, you can promote a read replica to master.

Mutli-AZ Deployment

While read replicas are meant for enhancing performance, Multi-AZ deployment is intended for high availability and disaster recovery.

- You can deploy multiple database instances in different availability zones using what RDS calls a multi AZ deployment.
- If using BYOL for Oracle, you must possess a license for both primary and standby instances.
- For Oracle, Microsoft SQL and PostgreSQL multi-AZ deployment all instances must reside in the same region.
- For MySQL and Maria DB you can create a multi-AZ read replica in a different region.
- Amazon Aurora handles multi-AZ a bit differently. An Amazon Aurora cluster consists of a primary Instance. Aurora give you a cluster endpoint that always points to primary instance. An Aurora cluster also may include Aurora replicas. The primary and all replicas share a single cluster volume, which is synchronously replicated across three AZs and can expand to 64 TB. In the event the primary instance fails, one of two things will happen.
    - If no Aurora replicas exist, Aurora will create a new primary instance.
    - If an Aurora replica does exist, Aurora will promote the replica to primary.

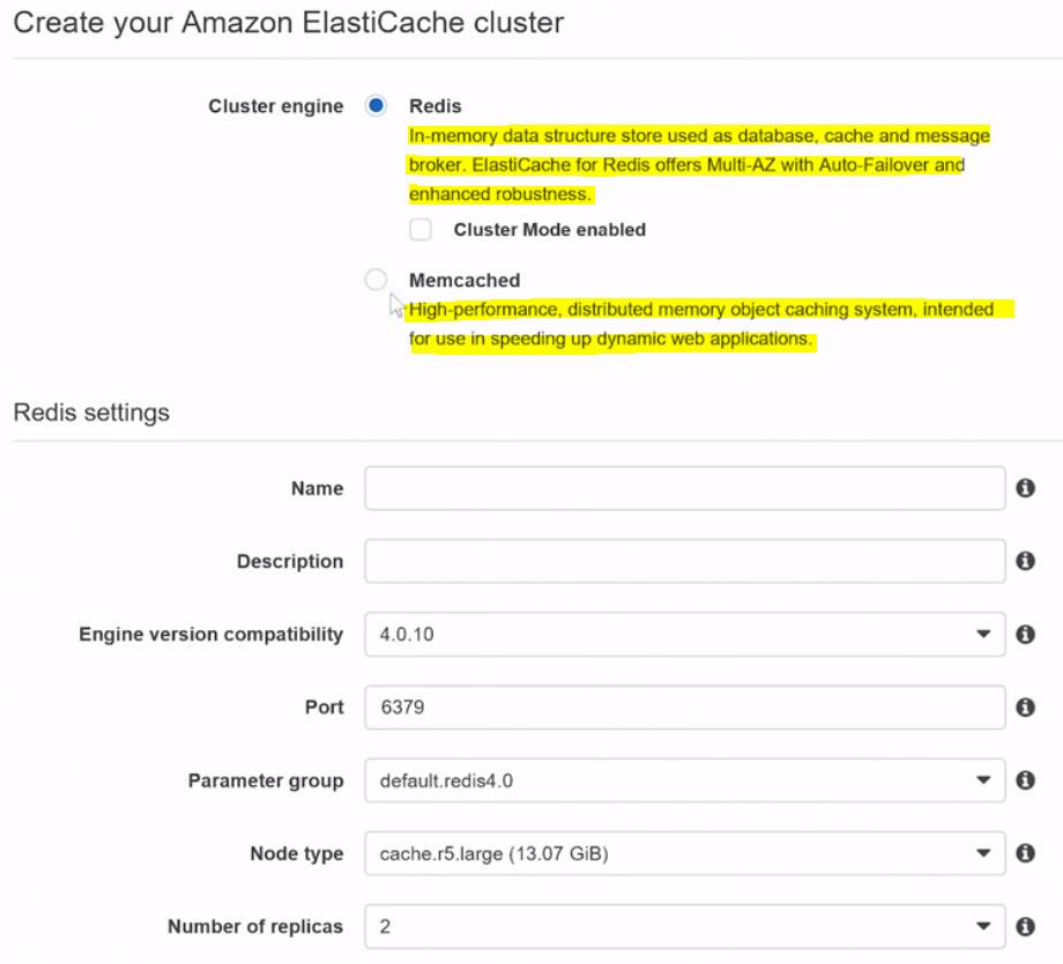
RDS Databases Backup/Restore

- By default, AWS backs up the database on a regular basis using snapshots.
- Backup retention periods can be set as low as 0 days (meaning no backup) and to a maximum of 35 days.
- You can also backup a database by copying all data to a local file, but this is not a common solution.
- You can create a snapshot manually any time you desire.
- When restoring a database to a point-in-time, you are actually creating a new db instance from a snapshot. And because you are creating a new database during a restore process, you can encrypt the database during restoration if you haven't done that with the original database.

ElastiCache

Fully managed in-memory data store, compatible with Redis or Memcached. Power real-time applications with sub-millisecond latency.

- In-memory caching for databases
- Memcached is the simplest mode for implementation, and provides higher performance.
- When regulatory compliance is required, Redis can be used.
- Pick an instance type that has the amount of memory needed.



Redshift (Data Warehouse DB)

- Used for Online Analytical Processing (OLAP).
- Large, central repository for data.
- Super fast read operations because it uses massively Parallel processing (MPP), data compression, and columnar data stores.
- Data aggregated from one or more sources.
- **Single Node** supports up to 160 GB. For more than 160 GB you will have to use a Multiple Node.
- A **Multiple node** consists of a **Leader node** (handles connections and queries) and a **compute node** (store data and execute queries and calculations).
- Compute Nodes are divided into two categories:
    - **Dense Compute**: can store up to 326 TB of data on magnetic storage.
    - **Dense Storage**: can store up to 2 PB of data on fast SSDs.
- Redshift supports SSL transit encryption and AES-256 storage encryption at rest using KMS.
- Redshift operates in one AZ, but snapshots can be restored to new AZs.
- Amazon Redshift Spectrum allows you to directly run SQL queries against exabytes of unstructured data in Amazon S3. No loading or transformation is required.

DynamoDB

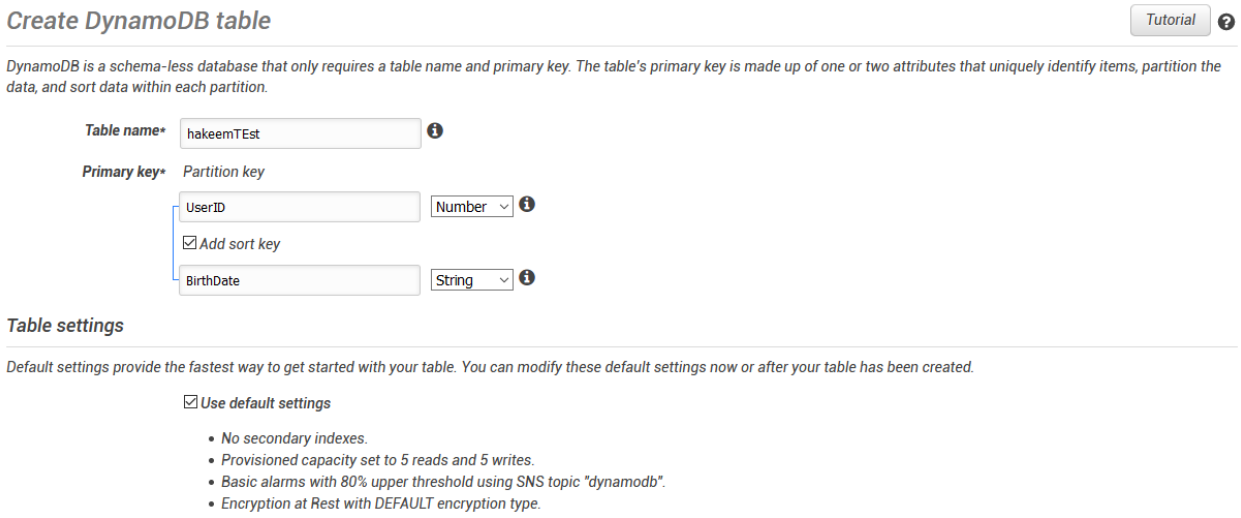
Around Christmas time of 2004, Amazon.com had a problem; they could not keep up with the load of too many shoppers. To resolve the problem, they decided to come up with a better database solution. That was DynamoDB, which eventually made its way into AWS ecosystem.

Amazon DynamoDB is a fully managed NoSQL database (non-relational database) service that provides fast and predictable performance with seamless scalability.

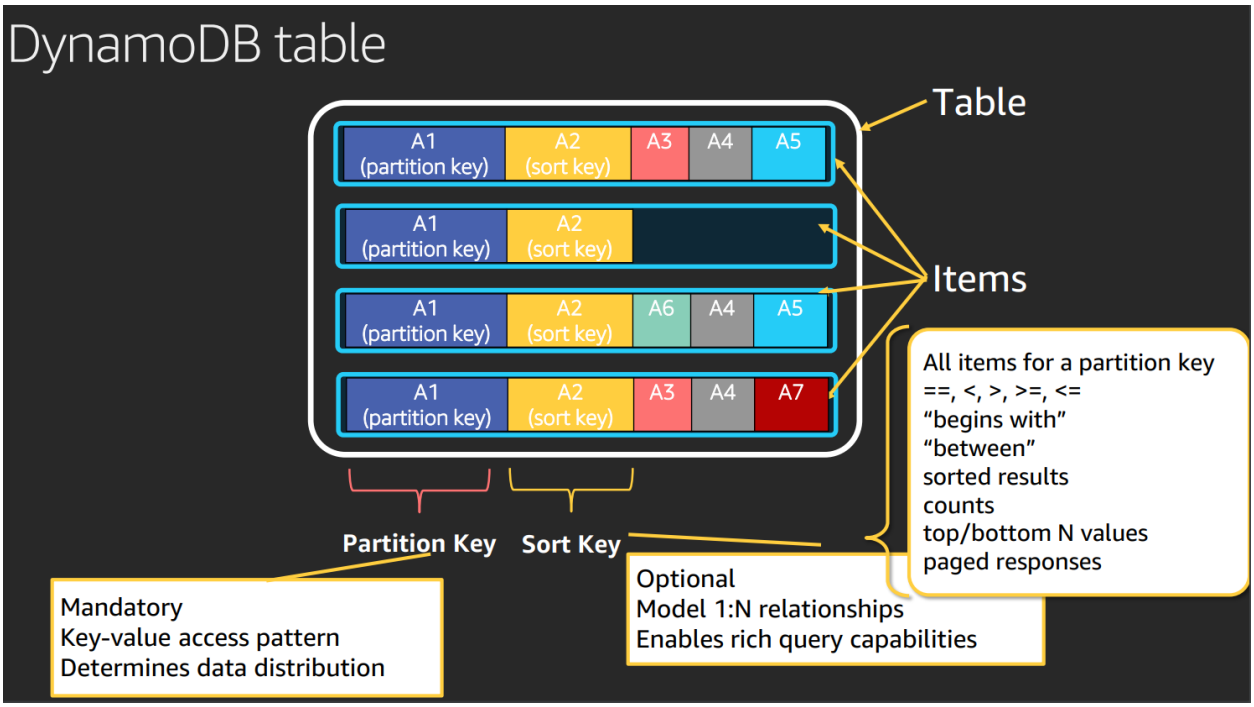
DynamoDB is all about small fast transactions.

Tables are the fundamental data structures in relational databases and in Amazon DynamoDB. A relational database management system (RDBMS) requires you to define the table's schema when you create it. In contrast, DynamoDB tables are schemaless—other than the primary key, you do not need to define any extra attributes or data types when you create a table. With DynamoDB you never create a database, you only create tables.

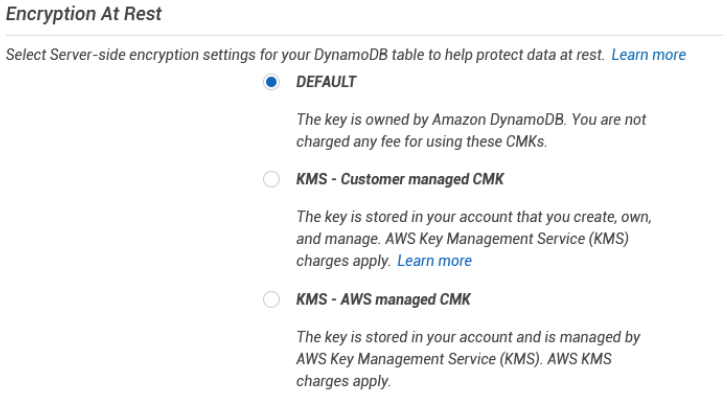
A primary key can be a combination of two values: a *partition key* and a *sort* (or range), and it would be called a composite primary key, versus a simple primary key that contains a single value that's a *partition key* (may also be called *hash key*).



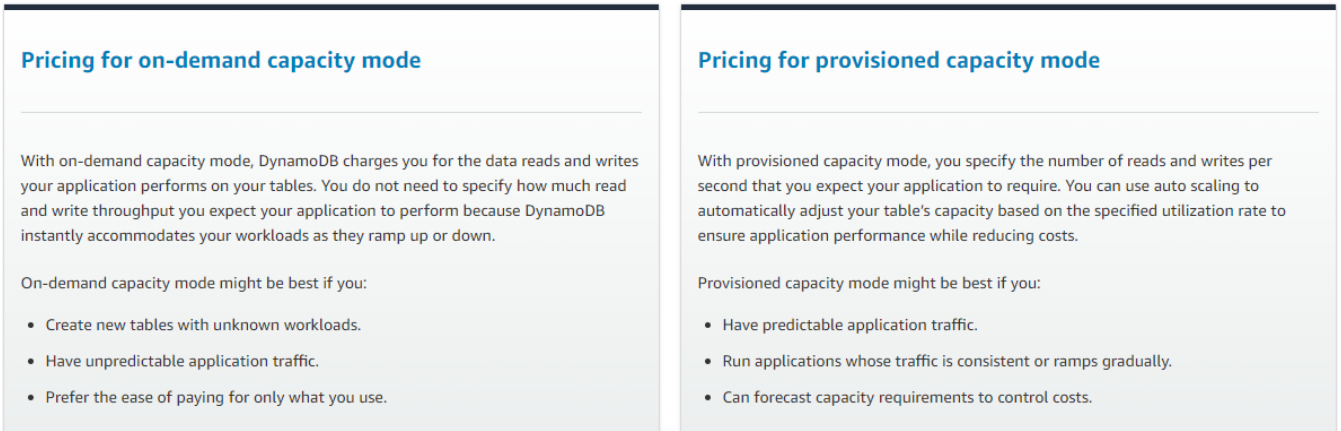
As a quick note, a good DynamoDB architecture avoids hot partitions by making the partitions keys as unique as possible.



- DynamoDB provides millisecond latency at any scale. It can handle thousands of reads and writes per second.
- Stored on SSD and spread across 3 distinct datacenters.
- DynamoDB have two read consistency types:
    - *Eventually consistent* reads (default) can have few seconds delays in replicating data
    - *Strongly consistent* reads replicates within milliseconds, so it always gives the most up-to-date data.
- Bills for storage and throughput (read and write units).
- DynamoDB automatically scales throughput capacity to meet workload demands, and partitions and repartitions your data as your table size grows. Also, DynamoDB synchronously replicates data across three facilities in an AWS Region, giving you high availability and data durability.
- Maximum throughput per DynamoDB table is practically unlimited.
- The smallest provisioned throughput you can request is 1 write capacity unit and 1 read capacity unit for both auto scaling and manual throughput provisioning. Such provisioning falls within the free tier which allows for 25 units of write capacity and 25 units of read capacity. The free tier applies at the account level, not the table level. In other words, if you add up the provisioned capacity of all your tables, and if the total capacity is no more than 25 units of write capacity and 25 units of read capacity, your provisioned capacity would fall into the free tier.
- You can create on-demand backups for your Amazon DynamoDB tables or enable continuous backups with point-in-time recovery.
- DynamoDB encrypts data at rest by default. You can use the default encryption, the AWS owned customer master key (CMK), or the AWS managed CMK to encrypt all your data. DynamoDB now has added support to enable you to switch encryption keys, between the AWS owned CMK and AWS managed CMK, without having to make any code or application modifications to encrypt your data.



- DynamoDB provides two different operations to let you read data from a table:
    - A scan list
    - A query return
- Amazon.com uses DynamoDB, it particularly useful in places where products are suggested to users instantly based on their current session browsing behavior.
- **DynamoDB Streams** help you to keep a list of item level changes or provide a list of item level changes that have taken place in the last 24hrs. Amazon DynamoDB is integrated with AWS Lambda so that you can create triggers—pieces of code that automatically respond to events in DynamoDB Streams.
- If you enable **DynamoDB Streams** on a table, you can associate the stream ARN with a Lambda function that you write. Immediately after an item in the table is modified, a new record appears in the table's stream. AWS Lambda polls the stream and invokes your Lambda function synchronously when it detects new stream records.
- An event source mapping identifies a poll-based event source for a Lambda function. It can be either an Amazon Kinesis or DynamoDB Streams. Event sources maintain the mapping configuration except for stream-based services (e.g. DynamoDB, Kinesis) for which the configuration is made on the Lambda side and Lambda performs the polling.
- Amazon DynamoDB Accelerator (DAX) is a fully managed, highly available, in-memory cache for DynamoDB that delivers up to a 10x performance improvement – from milliseconds to microseconds – even at millions of requests per second. You can enable DAX for a DynamoDB database with a few clicks.



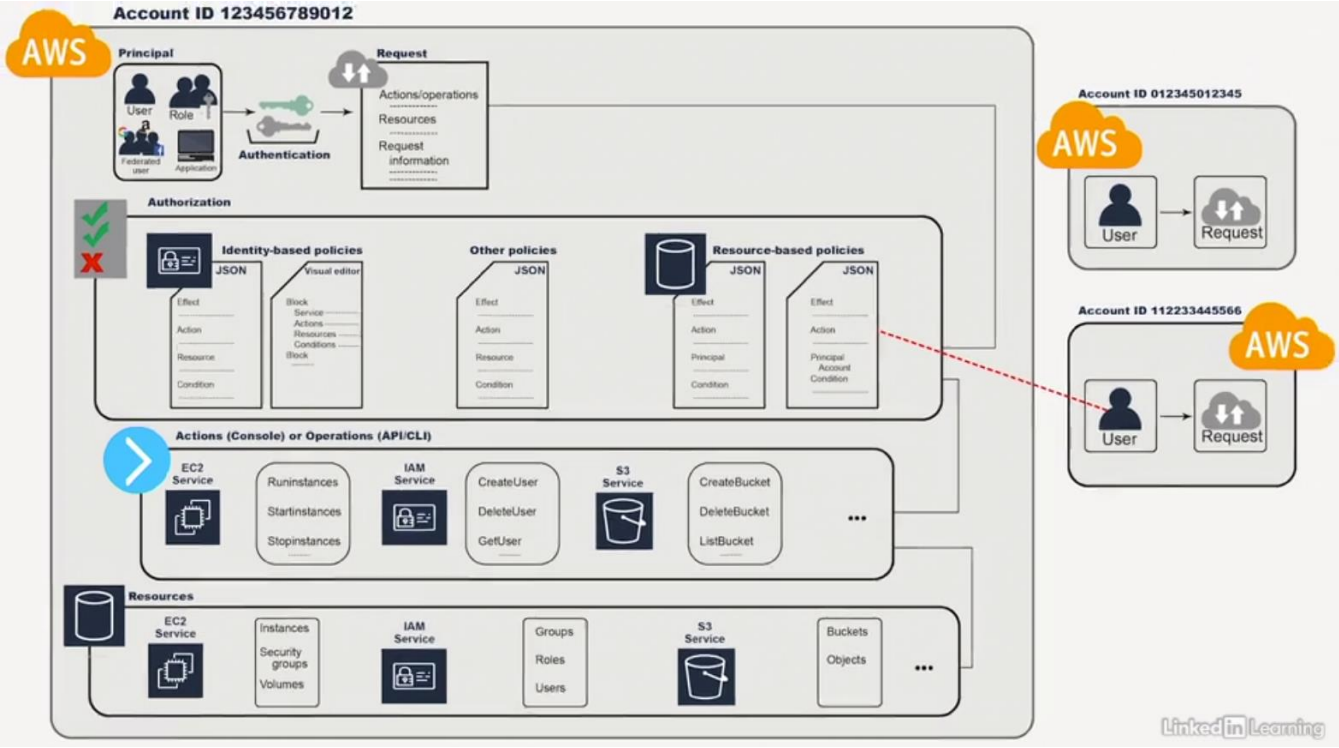
- DynamoDB best practices include:
  - Keep item sizes small.
  - If you are storing serial data in DynamoDB that will require actions based on data/time use separate tables for days, weeks, months.
  - Store more frequently and less frequently accessed data in separate tables.
  - If possible compress larger attribute values.
  - Store objects larger than 400KB in S3 and use pointers (S3 Object ID) in DynamoDB.

Identity and Access Management (IAM)

It is important to understand the whole hierarchy model or architecture of IAM.

You have **Principals** (*Users/Groups*), that are given **Authorization** (through *Identity-based or resource-based Policies*) in order to perform **Actions** or **Operations** on **Resources**.

The below illustration shows this hierarchy.



- Entities that can perform an action in AWS are called principals or identities and they are:
  - Users (*Person or Service*)
  - Groups
  - Roles
- Roles are an identity granted permission. Roles are not permanently assigned; they are assumable by any entity with a need for it.
- Roles are compatible with federated users.
- The root user is the email address used when creating the AWS subscription. As a best practice, it is always recommended NOT to use the AWS root user account for everyday tasks. However, there are certain tasks that only the root user can do, examples:
  - Modifying the root user itself.
  - Changing AWS support plan.
  - Closing an AWS account.
  - Creating a CloudFront key pair.

    - Enabling Multi-Factor Authentication (MFA) on an S3 bucket.
    - Restore permissions for other IAM users.
- Console access is authenticated through username and password. While CLI and API is authenticated through access keys and secret keys.

Key Management Service (KMS)

AWS KMS is a managed service that enables you to easily create and control the keys used for cryptographic operations. The service provides a highly available key generation, storage, management, and auditing solution for you to encrypt or digitally sign data within your own applications or control the encryption of data across AWS services.

KMS is a global service and not tied to a region, however, asymmetric keys are currently only available in Northern Virginia, Oregon, Sydney, Ireland, and Tokyo.

Difference Between Symmetric and Asymmetric Encryption:
The basic difference between these two types of encryption is that symmetric encryption uses one key for both encryption and decryption, and the asymmetric encryption uses public key for encryption and a private key for decryption

Features:

You can perform the following key management functions:

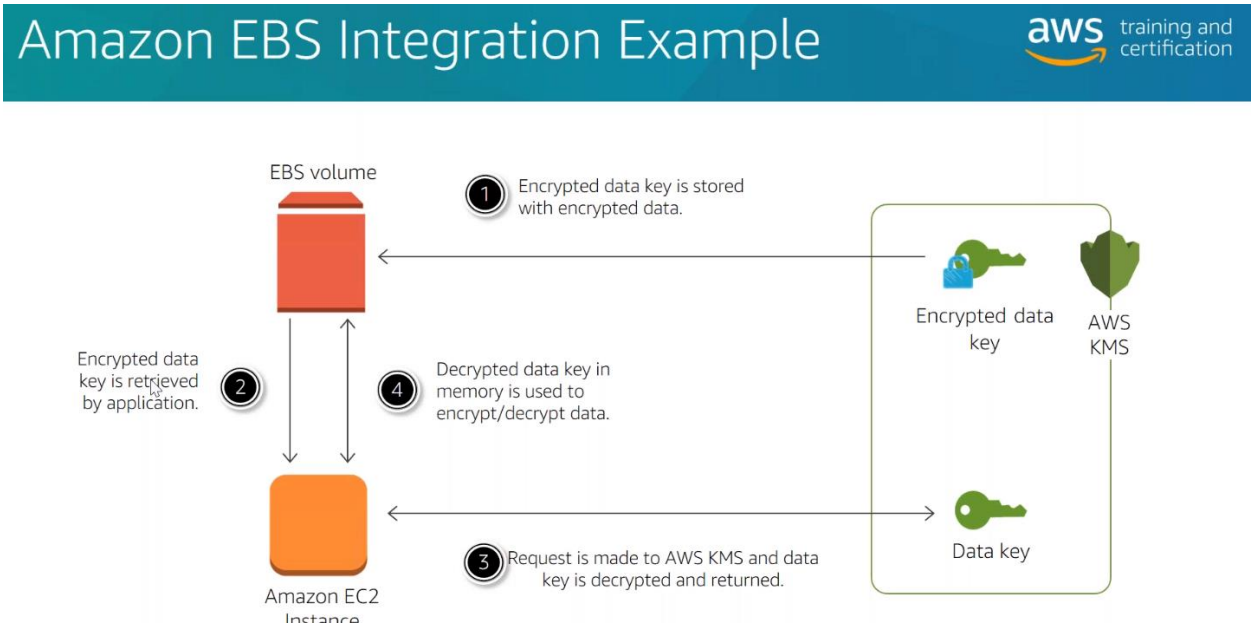
- Create symmetric and asymmetric keys where the key material is only ever used within the service
- Create symmetric keys where the key material is generated and used within a custom key store under your control. **Note that the use of custom key stores requires CloudHSM resources to be available in your account**.
- Import your own symmetric key material for use within the service
- Create both symmetric and asymmetric data key pairs for local use within your applications
- Define which IAM users and roles can manage keys
- Define which IAM users and roles can use keys to encrypt and decrypt data
- Choose to have keys that were generated by the service to be automatically rotated on an annual basis
- Temporarily disable keys so they cannot be used by anyone
- Re-enable disabled keys
- Schedule the deletion of keys that you no longer use
- Audit use of keys by inspecting logs in AWS CloudTrail

How does AWS KMS work?

You start using the service by requesting the creation of a CMK (Customer Master Key). You control the lifecycle of the CMK as well as who can use or manage it. The key material for a CMK is generated within hardware security modules (HSMs) managed by AWS KMS. Alternatively, you can import key material from your own key management infrastructure and associate it with a CMK. You can also have the key material generated and used in an AWS CloudHSM cluster as a part of the custom key store feature in AWS KMS.

Once you have created a CMK using any of the three supported options, you can submit data directly to the service AWS KMS to be signed, verified, encrypted, or decrypted using these CMK. You set usage policies on these keys that determine which users can perform which actions under which conditions.

The below illustrates a KMS integration with EBS volume as an example:



CloudHSM

The AWS CloudHSM service helps you meet corporate, contractual, and regulatory compliance requirements for data security by using dedicated Hardware Security Module (HSM) instances within the AWS cloud. AWS and AWS Marketplace partners offer a variety of solutions for protecting sensitive data within the AWS platform, but for some applications and data subject to contractual or regulatory mandates for managing cryptographic keys, additional protection may be necessary. CloudHSM complements existing data protection solutions and allows you to protect your encryption keys within HSMs that are designed and validated to government standards for secure key management. CloudHSM allows you to securely generate, store, and manage cryptographic keys used for data encryption in a way that keys are accessible only by you.

A Hardware Security Module (HSM) provides secure key storage and cryptographic operations within a tamper-resistant hardware device. HSMs are designed to securely store cryptographic key material and use the key material without exposing it outside the cryptographic boundary of the hardware.

Directory Service

AWS Directory Service is a managed service offering, providing directories that contain information about your organization, including users, groups, computers, and other resources. As a managed offering, AWS Directory Service is designed to reduce management tasks, thereby allowing you to focus more of your time and resources on your business. There is no need to build out your own complex, highly-available directory topology because each directory is deployed across multiple Availability Zones, and monitoring automatically detects and replaces domain controllers that fail. In addition, data replication and automated daily snapshots are configured for you. There is no software to install and AWS handles all of the patching and software updates.

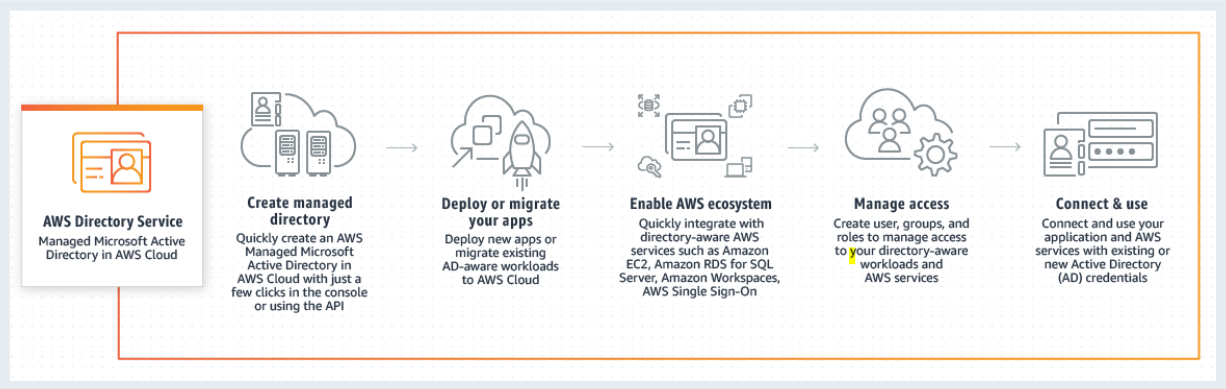
Directory Types:

- **AWS Managed Microsoft AD:** With AWS Managed Microsoft AD, you can easily enable your Active Directory-aware workloads and AWS resources to use managed actual Microsoft Active Directory in the AWS Cloud. Workload examples include Amazon EC2, Amazon RDS for SQL Server, custom .NET applications, and AWS Enterprise IT applications such as Amazon WorkSpaces.

- **AD Connector:** AD Connector is a proxy for redirecting directory requests to your existing Microsoft Active Directory without caching any information in the cloud. AD Connector comes in two sizes, small and large. A small AD Connector is designed for smaller organizations of up to 500 users. A large AD Connector can support larger organizations of up to 5,000 users.

- **Amazon Cognito User Pools:** With user pools, you can add user registration and sign-in features to your apps. Users can sign in with an email address, phone number, or user name rather than use an external identity provider like Facebook or Google. You can also create custom registration fields and store that metadata in your user directory. You can verify email addresses and phone numbers, recover passwords, and enable multi-factor authentication (MFA) with just a few lines of code.

How it works



Uses Cases:

- Use Active Directory Group Policy objects (GPOs)
- Highly available Active Directory in the AWS Cloud
- Single sign-on (SSO) with Active Directory credentials.
- Seamlessly Domain Join Amazon EC2 Instances from Multiple Accounts & VPCs

NACLs and Security Groups

NACLs

- Network Access Control Lists are applied on subnets not instances.
- Processing in NACLs is stateless
- Supports both allow and deny rules.
- Rule number defines precedence, lowest numbered rules first. First match applies.

Security groups

- Security groups acts like a firewall.
- Assigned to an instance in a VPC
- Applied to instances and NOT to subnets.
- Define allow traffic (No deny rules, only allow) flows for ingress and egress.
- Stateful firewall.
- By Default, no inbound traffic is allowed without request.
- By default, all outbound traffic is allowed.
- By Default, security are only bound to the primary network interface. However, they can be bound to other network interfaces, including ENIs.

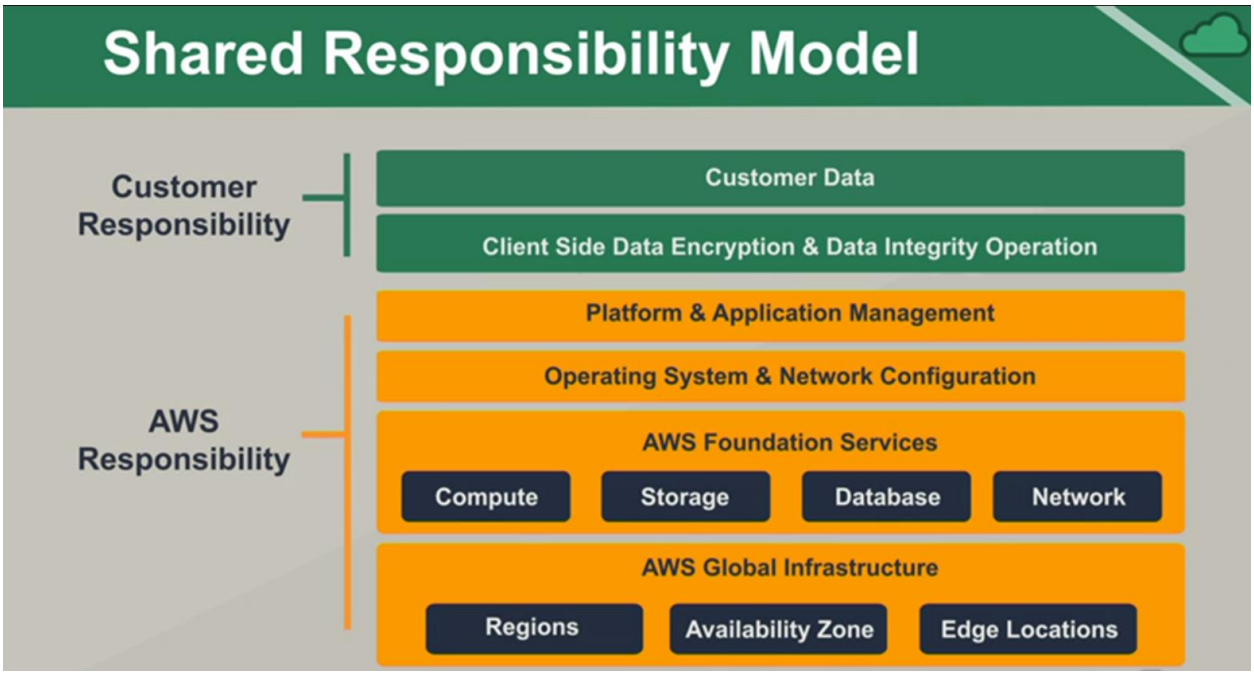
Security Groups vs. ACLs

Security Group	Network ACL
Operates at the instance level	Operates at the subnet level
Supports allow rules only	Supports allow and deny rules
Is stateful: Return traffic is automatically allowed, regardless of any rules	Is stateless: Return traffic must be explicitly allowed by rules
We evaluate all rules before deciding whether to allow traffic	We process rules in number order when deciding whether to allow traffic
Applies to an instance only if someone specifies the security group when launching the instance, or associates the security group with the instance later on	Automatically applies to all instances in the subnets it's associated with (therefore, you don't have to rely on users to specify the security groups)

AWS Shared Responsibility Model

Amazon is responsible for the security OF the cloud.

Customer is responsible for the security IN the cloud.



As a best practice, always apply the principle of least privilege.

Web Application Firewall (WAF) & Shield

- WAF is an http/https firewall that controls access to HTTP and HTTPS servers based on requests or source IPs.
- WAF works with CloudFront and/or Elastic load balancer, S3 bucket or other http/https content source.
- WAF can be **open-to-closed** (allow all requests except the ones that you specify), or **closed-to-open** (block all requests except the one that you specify).
- WAF allows monitoring for requests that match specified parameters and it can integrate with CloudWatch, CloudTrail and even SNS to receive alerts discoveries done by WAF.
- WAF will return HTTP 403 error for the source if the request is denied for any reason.
- AWS Shield is a managed DDoS protection solution.
- AWS Shield standard is automatically enabled for your account at no additional cost when you use services like Elastic Load Balancing (ELB), Application Load Balancer, Amazon CloudFront and Amazon Route 53.

Cognito

Congito is a user identity and data synchronization service that provides AWS users with single-sign-on capability (SSO) using public identity providers such as Google, Facebook and amazon.com as well as private identity providers such as Active Directory with SAML.

Cognito allows for profile management of federated identities without the need to create them in IAM. It can also scale to millions of users. This is important in situation where you're creating say a gaming app and when launching it many of users registered using their existing facebook or google accounts.

Cognito is based on open standards such as:

- OAuth 2.0
- SAML 2.0
- OpenID Connect

Cognito controls access to AWS resources by defining roles and then mapping users to those roles.

Other Security Services

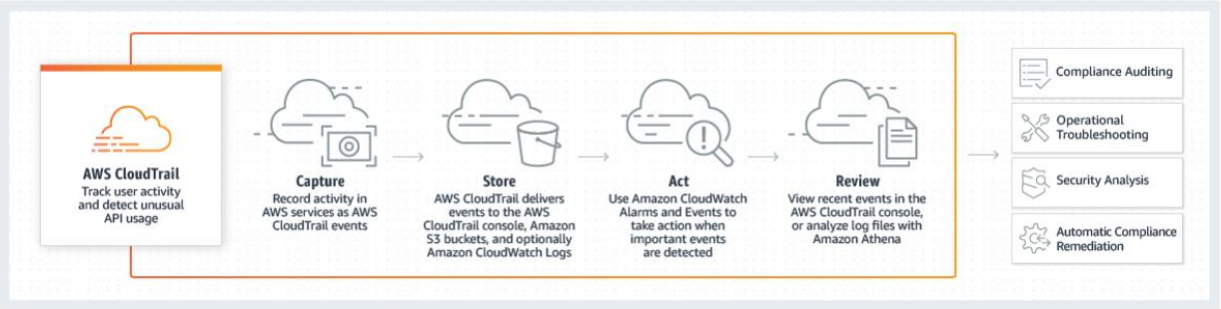
- AWS Security Hub is a paid subscription that runs automatic checks to scan for compliance with regulations and laws.
- Amazon Guard Duty is an intrusion detection system (IDS), Amazon Inspector performs vulnerability analysis and Amazon Macie provide S3 bucket policy compliance scans.
- AWS Organization provides a centralized management interface for billing and account management at no additional charge.
- Service Control Policies (SCP) in AWS organizations enable fine-grained permission controls.

AWS Core Services - Applications Deployment and Management

CloudTrail

AWS CloudTrail is a service that enables governance, compliance, operational auditing, and risk auditing of your AWS account. With CloudTrail, you can log, continuously monitor, and retain account activity related to actions across your AWS infrastructure. CloudTrail provides event history of your AWS account activity, including actions taken through the AWS Management Console, AWS SDKs, command line tools, and other AWS services. This event history simplifies security analysis, resource change tracking, and troubleshooting. In addition, you can use CloudTrail to detect unusual activity in your AWS accounts. These capabilities help simplify operational analysis and troubleshooting.

How it works



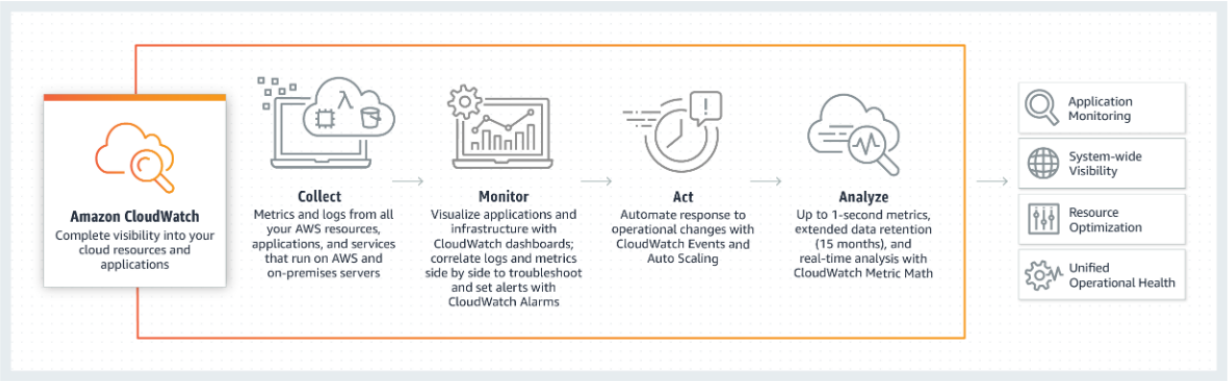
- **Free Tier**: AWS CloudTrail allows you to view and download the last 90 days of your account activity for create, modify, and delete operations of supported services free of charge.
- **Logging**: CloudTrail keeps detailed event logs of every action that occurs against your AWS resources (IP of the requester, data and time, Region, API action performed, service…).
- **Management Events vs. Data Events**: CloudTrail also classifies events along two other dimensions: Management events and Data events. **Management events** are operations that a user or service attempts to execute against an AWS resource. While **Data events** consist of S3 object-level activity and Lambda function executions.
- **Trail**: When you open an AWS account, CloudTrail begins logging all of your management events automatically. The **event history** log doesn't record data events. If you need to store more than 90 days of event history, you need to create a *Trail*. A **Trail** is a configuration that directs CloudTrail to record specified events in log files and deliver them to an S3 bucket. A trail can log events either from a single region or all regions. You can choose to log management events, data events, or both. You can also choose whether to log read-only or write-only events, or both.
- **Log Encryption:** By default, AWS CloudTrail encrypts all log files delivered to your specified Amazon S3 bucket using Amazon S3 server-side encryption (SSE). Optionally, add a layer of security to your CloudTrail log files by encrypting the log files with your AWS Key Management Service (AWS KMS) key. Amazon S3 automatically decrypts your log files if you have decrypt permissions.
- **Log File Integrity Validation** is an *optional* feature that uses cryptographic hash calculations to provide assurance that no CloudTrail log files are surreptitiously modified or deleted.

CloudWatch

Amazon CloudWatch is a monitoring service for AWS cloud resources and the applications you run on AWS. You can use Amazon CloudWatch to collect and track metrics, collect and monitor log files, and set alarms. Amazon CloudWatch can monitor on-premises resources and AWS resources such as Amazon EC2 instances, Amazon DynamoDB tables, and Amazon RDS DB instances, as well as custom metrics generated by your applications and services, and any log files your applications generate. You can use Amazon CloudWatch to gain system-wide visibility into resource utilization, application performance, and operational health. You can use these insights to react and keep your application running smoothly.

How it works

CloudWatch collects monitoring and operational data in the form of logs, metrics, and events, and visualizes it using automated dashboards so you can get a unified view of your AWS resources, applications, and services that run in AWS and on-premises. You can correlate your metrics and logs to better understand the health and performance of your resources. You can also create alarms based on metric value thresholds you specify, or that can watch for anomalous metric behavior based on machine learning algorithms. To take action quickly, you can set up automated actions to notify you if an alarm is triggered and automatically start auto scaling, for example, to help reduce mean-time-to-resolution. You can also dive deep and analyze your metrics, logs, and traces, to better understand how to improve application performance.



Main Features

- Easily collect and store logs.
- Contains built-in metrics and allows collection of custom metrics.
- Unified operational view with dashboards.
- Set high resolution alarms.
- Application insights for .NET and SQL server applications.
- Integrates with Auto Scaling, you can set alarm that trigger automated Auto Scaling action.
- Automate response to operational changes with CloudWatch Events.
- Log analytics, granular data and extended retention.

Free tier

You can get started with Amazon CloudWatch for free. Most AWS Services (EC2, S3, Kinesis, etc.) vend metrics automatically for free to CloudWatch. Many applications should be able to operate within these free tier limits. You can learn more about AWS Free Tier here.

Metrics	Basic Monitoring Metrics (at 5-minute frequency) 10 Detailed Monitoring Metrics (at 1-minute frequency) 1 Million API requests (not applicable to GetMetricData and GetMetricWidgetImage)
Dashboard	3 Dashboards for up to 50 metrics per month
Alarms	10 Alarm metrics (not applicable to high-resolution alarms)
Logs	5GB Data (ingestion, archive storage, and data scanned by Logs Insights queries)
Events	All events except custom events are included
Contributor Insights	1 Contributor Insights rule per month The first one million log events that match the rule per month
Synthetics	100 canary runs per month

- CloudWatch organizes metrics into namespaces. You can think of a namespace as a container for metrics. And you can create custom namespaces for custom metrics.
- A Metric functions as a variable and contains a time-ordered set of data points.
- A dimension is a name-value pair that distinguishes metrics with the same name and namespace from one another

Basic and Detailed Monitoring

- **Basic Monitoring** sends metrics to CloudWatch every **five minutes**.
- **Detailed Monitoring** sends metrics to CloudWatch every **minute**.

Regular and High-Resolution Metrics

- **Regular resolution metrics** have a timestamp resolution of no less than one minute.
- Metrics with a resolution of less than a minute are considered **High resolution metrics** and can reach down to 1-second resolution.

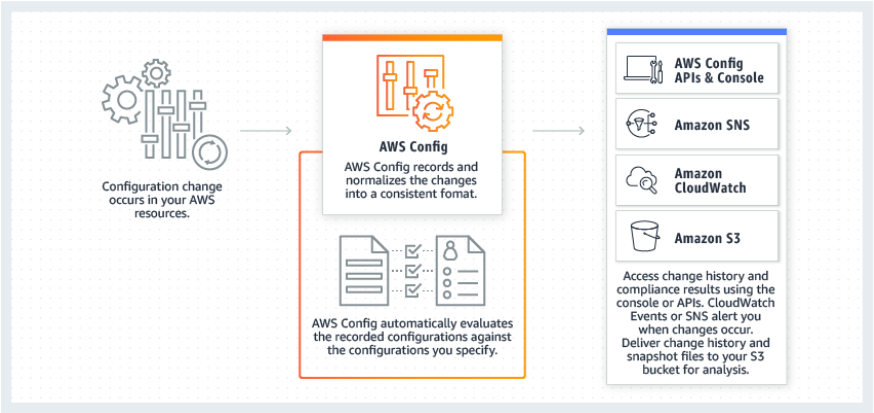
What is the retention period of all metrics?

CloudWatch launched High Resolution Custom Metrics on July 26, 2017. This enables you to publish and store custom metrics down to 1-second resolution. Extended retention of metrics was launched on November 1, 2016, and enabled storage of all metrics for customers from the previous 14 days to 15 months. CloudWatch retains metric data as follows:

- Data points with a period of **less than 60 seconds** are available for **3 hours**. These data points are high-resolution custom metrics.
- Data points with a period of 60 seconds (**1 minute**) are available for **15 days**
- Data points with a period of 300 seconds (**5 minute**) are available for **63 days**
- Data points with a period of 3600 seconds (**1 hour**) are available for 455 days (**15 months**)

AWS Config

AWS Config is a fully managed service that provides you with an AWS resource inventory, configuration history, and configuration change notifications to enable security and governance. You can aslo discover existing AWS resources, export a complete inventory of your AWS resources with all configuration details, and determine how a resource was configured at any point in time. These capabilities enable compliance auditing, security analysis, resource change tracking, and troubleshooting.



Benefits:

- **Continuous monitoring and assessment**: With AWS Config, you are able to continuously monitor and record configuration changes of your AWS resources. It also allows you to continuously audit and assess the overall compliance of your AWS resource configurations with your organization's policies and guidelines.
- **Configuration History**: Config also enables you to inventory your AWS resources, the configurations of your AWS resources, as well as software configurations within EC2 instances at any point in time.
- **Change Management**: With AWS Config, you are able to track the relationships among resources and review resource dependencies prior to making changes. Once a change occurs, you are able to quickly review the history of the resource's configuration and determine what the resource's configuration looked like at any point in the past. Config provides you with information to assess how a change to a resource configuration would affect your other resources, which minimizes the impact of change-related incidents.

AWS Systems Manager

Systems Manager Services is a collection of tools for monitoring and managing the resources running in the AWS cloud and on-premises infrastructure.

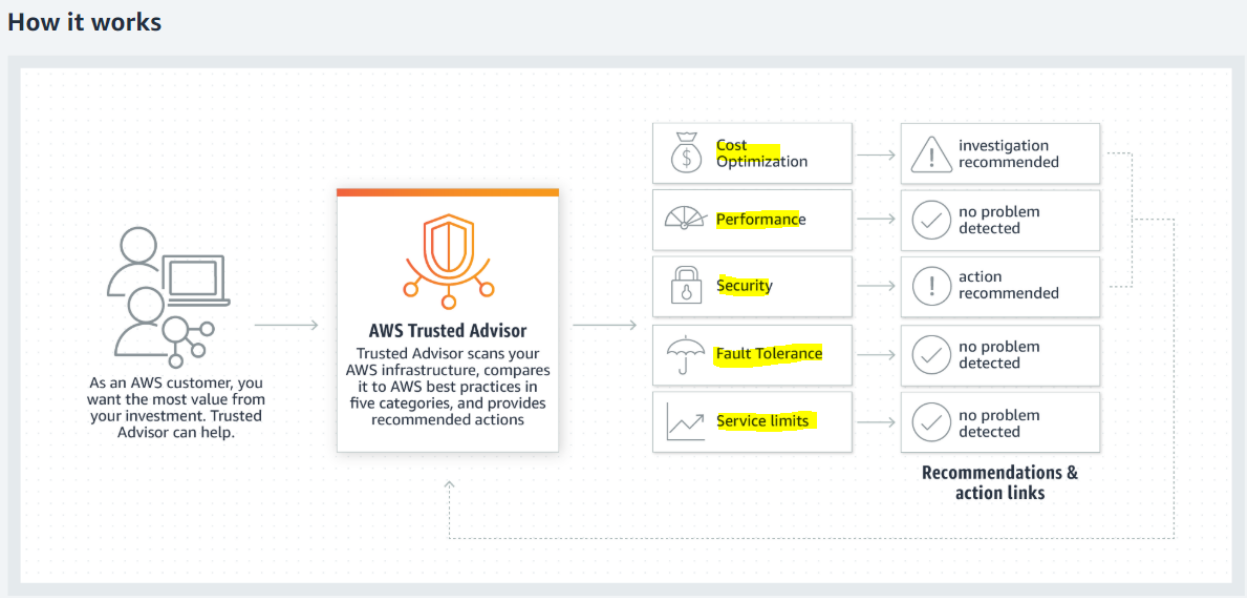
Systems Manager lets you automatically or manually perform actions against your AWS resources and on-premises.

AWS Systems Manager is, in-a-way, equivalent to Microsoft SCCM solution we use in a traditional on-premises environment.

Trusted Advisor

AWS Trusted Advisor is an online tool that provides you real time guidance to help you provision your resources following AWS best practices. Trusted Advisor checks help optimize your AWS infrastructure, increase security and performance, reduce your overall costs, and monitor service limits.

AWS Basic Support and AWS Developer Support customers get access to 6 security checks (S3 Bucket Permissions, Security Groups - Specific Ports Unrestricted, IAM use, MFA on Root Account, EBS Public Snapshots, RDS Public Snapshots) and 50 service limit checks. AWS Business Support and AWS Enterprise Support customers get access to all 115 Trusted Advisor checks (14 cost optimization, 17 security, 24 fault tolerance, 10 performance, and 50 service limits) and recommendations. For a complete list of checks and descriptions, explore Trusted Advisor Best Practices.



CloudFormation

AWS CloudFormation is a service that gives developers and businesses an easy way to create a collection of related AWS and third party resources and provision them in an orderly and predictable fashion.

AWS CloudFormation introduces two concepts:

- **The Template**: a JSON or YAML-format, text-based file that describes all the AWS resources you need to deploy to run your application and
- **The Stack**: the set of AWS resources that are created and managed as a single unit when AWS CloudFormation instantiates a template.

The GUI interface doesn't have it all, so you will need to learn some JSON and YAML in order to fully use CloudFormation.

Why use CloudFormation?

- Rapid deployment
- Mirror existing internal architectures
- Take advantage of templates created by others (example wordpress template done by bitnami).
- Mirror or duplicate your cloud architecture with CloudFormer.

OpsWorks

AWS OpsWorks is a configuration management service that provides managed instances of Chef and Puppet. Chef and Puppet are automation platforms that allow you to use code to automate the configurations of your servers. OpsWorks lets you use Chef and Puppet to automate how servers are configured, deployed, and managed across your Amazon EC2 instances or on-premises compute environments. OpsWorks has three offerings, **AWS Opsworks for Chef Automate**, **AWS OpsWorks for Puppet Enterprise**, and **AWS OpsWorks Stacks**.

- OpsWorks can be used to configure code-based deployments including: instance deployment, service deployment or application deployment.
- OpsWorks stacks include layers of services and runtime environments.
- Chef Automate users cookbooks of recipes to launch solutions.
- Puppet uses master servers with pre-configured modules.

As an architect, OpsWorks tools will come in handy in situations where a company want to migrate an entire datacenter into the cloud, where OpsWorks is used to manage base-line configuration of new instances instead of configuring every single instance alone.

Kinesis

Amazon Kinesis makes it easy to collect, process, and analyze real-time, streaming data so you can get timely insights and react quickly to new information.

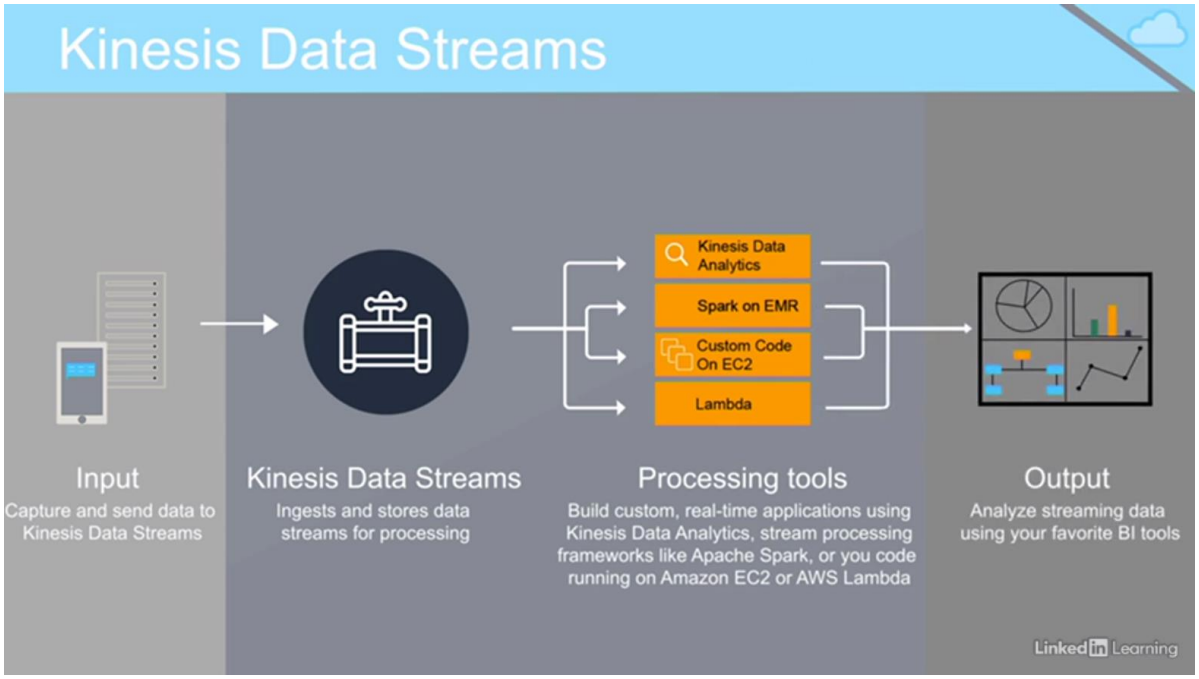
Kinesis is a fully managed service used for the processing of streaming data *example social media data related to your business.*

Kinesis provides real-time data analytics and can enable the use of multi-tier, decoupled applications.

No custom coding required, just configure producers, consumers and focus on your analysis.

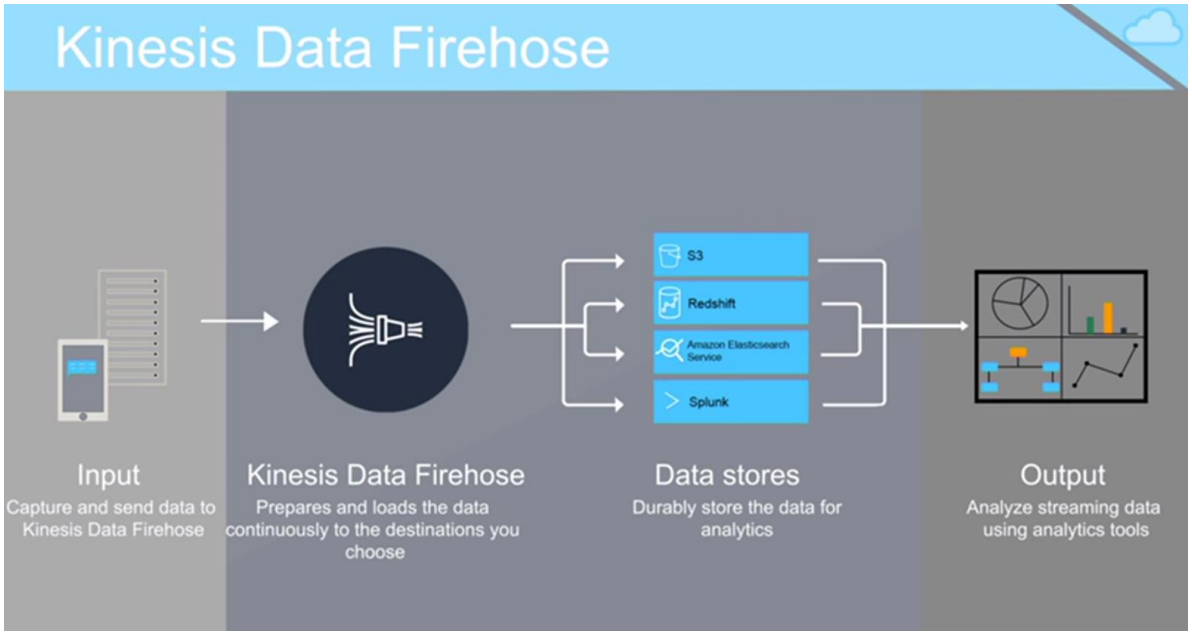
Kinesis has the following operating modes:

- **Kinesis Data Streams**: scalable and durable real-time data streaming service that can continuously capture gigabytes of data per second from hundreds of thousands of sources.



Kinesis Data streams provides streaming data to processing tools like Kinesis Data Analytics, Lambda, and custom code on EC2 instances.

- **Kinesis Data Firehose**: is the easiest way to capture, transform, and load data streams into AWS data stores for near real-time analytics with existing business intelligence tools.

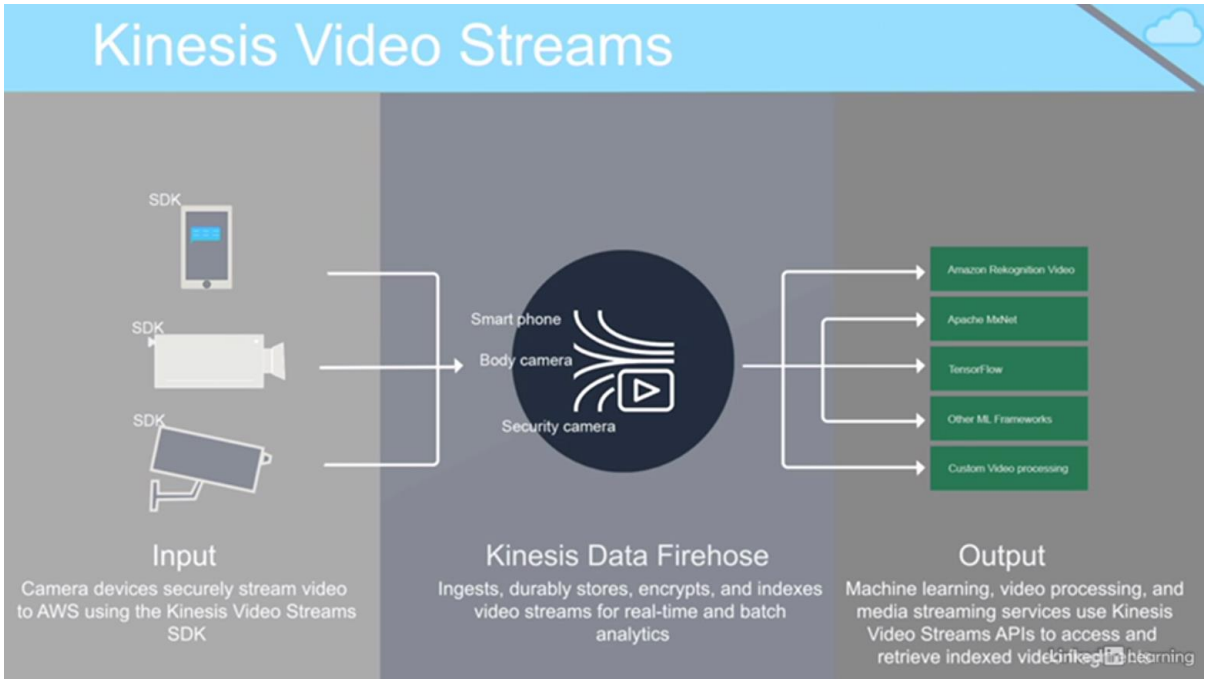


Kinesis Data Firehose can prepare data from data streams and place it into S3 buckets, Redshift, Elasticsearch, and Splunk.

Kinesis Data Streams vs. Kinesis Data Firehose

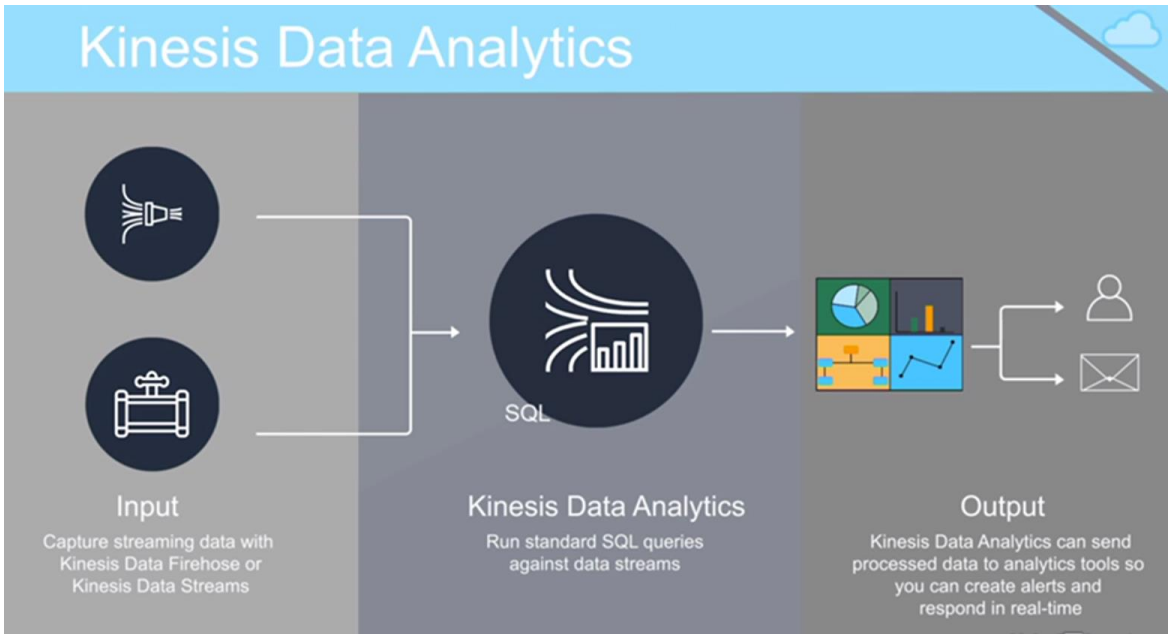
The difference here that's key to understand for the exam and for selecting which one you want is with **Kinesis Data Streams**, we have this data that's brought in and stored temporarily by Kinesis Data Streams. And it's held there for us until we can pull it out. With **Kinesis Data Firehose**, things are a little different now. We're just dumping the data as soon as it comes in, it's like a firehose, right? So we're throwing that data right out to the destination.

- **Kinesis Video Streams**: securely stream video from connected devices to AWS for analytics, machine learning (ML), and other processing.



Kinesis video Streams can output video stream to the Amazon Recognition Video, TensorFlow, Apache mxNet and even custom video processing applications.
A good usage case is streaming your CCTV surveillance feed right into the cloud where it would be analyzed and processed and send you alerts or whatever.

- **Kinesis Data Analytics**: is the easiest way to process data streams in real time with SQL or Java without having to learn new programming languages or processing frameworks.



- As you may have noticed from the above illustration, Kinesis Data Analytics uses Kinesis Data Streams or Kinesis Data Firehose as input.
- Kinesis Data Analytics is used when you need to analyze real-time data in real-time.
- It is based on Standard SQL queries or Java.
- After processing, Kinesis Data Analytics can send the result to other analytics tools and real-time responses.

Elastic MapReduce (EMR)

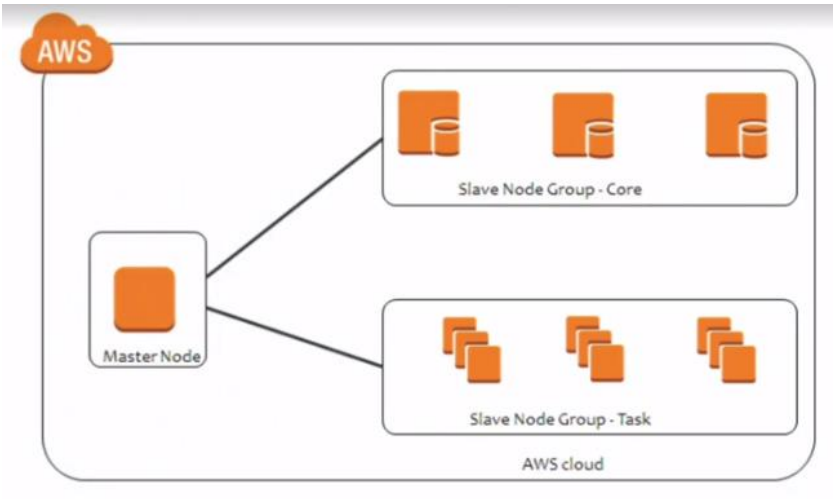
Amazon Elastic MapReduce (Amazon EMR) is a web service that makes it easy to quickly and cost-effectively process vast amounts of data by distributing processing across clusters.

Amazon EMR uses Hadoop, an open source framework, to distribute your data and processing across a resizable cluster of Amazon EC2 instances. Amazon EMR is used in a variety of applications, including log analysis, web indexing, data warehousing, machine learning, financial analysis, scientific simulation, and bioinformatics.

EMR pulls data from S3 buckets and uses EC2 instances for processing, where the user defines the number of needed clusters.

With EMR we have three different kinds of cluster nodes:

1- **Master Node**: Coordinates the job distribution across core and task nodes.
2- **Core Nodes**: Runs tasks assigned by the master node and stores data in the cluster.
3- **Task Nodes**: Runs only tasks that do NOT store data.

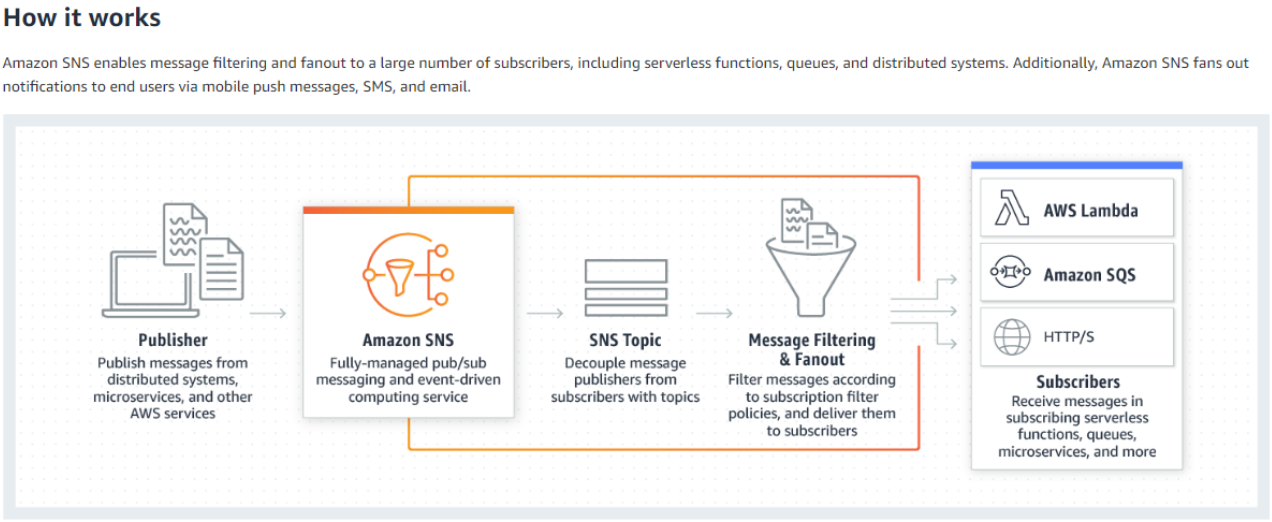


AWS Core Services - Application Integration

Simple Notification Service (SNS)

Amazon Simple Notification Service (Amazon SNS) is a web service that makes it easy to set up, operate, and send notifications from the cloud. It provides developers with a highly scalable, flexible, and cost-effective capability to publish messages from an application and immediately deliver them to subscribers or other applications. It is designed to make web-scale computing easier for developers. Amazon SNS follows the "publish-subscribe" (pub-sub) messaging model, with notifications being delivered to clients using a "push" mechanism that eliminates the need to periodically check or "poll" for new information and updates.

Simply put, SNS is "Paging-in-the-cloud"



- Publishers use SNS to push messages to a topic. Publisher Example: CloudWatch, Cost Explorer etc.
- A "topic" is simply a placeholder to stick some messages. Topic Example: Admin alerts, budget alerts, performance alert etc.
- Subscribers are the clients receiving all the notifications broadcasted to the topic.
- Publishers and subscribers aren't "aware" of each other.
- SNS is stored across multiple AZ, so it is redundant by nature.
- SNS has several delivery options to subscribers:
    - HTTP/HTTPS (to input into certain website for example to store notification in a log format), Email Notifications, SMS, Lambda function, SQS.
- SNS messages can be up to 256KB of data for all above delivery options except with SMS where it has a special constraint of 140 bytes for a single SMS. Larger SMS messages are sent as multiple transmission with a maximum aggregate size of 1600 bytes.

Simple Queue Service (SQS)

Amazon Simple Queue Service (SQS) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications.

Amazon's SQS is the managed equivalent of Microsoft's SQL server Service Broker.

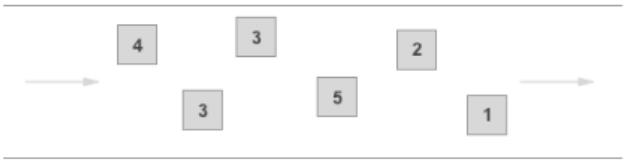
- SQS Messages are basically outputs from processes and input to other processes.
- In **SQS Standard** messages are processed asynchronously, meaning that they are non-linear, they don't have to be processed in the same order they come in.
- SQS is by default redundant across multiple AZs and it scales automatically with the changes in load.
- Messages can be up to 256 KB in size. Messages are usually queued until processed with a retention period of 14 days.
- SQS offers two types of message queues: **Standard Queues** and **FIFO Queues:**

Standard Queues

Unlimited Throughput: Standard queues support a nearly unlimited number of transactions per second (TPS) per API action.

At-Least-Once Delivery: A message is delivered at least once, but occasionally more than one copy of a message is delivered.

Best-Effort Ordering: Occasionally, messages might be delivered in an order different from which they were sent.



You can use standard message queues in many scenarios, as long as your application can process messages that arrive more than once and out of order, for example:

- Decouple live user requests from intensive background work: Let users upload media while resizing or encoding it.
- Allocate tasks to multiple worker nodes: Process a high number of credit card validation requests.
- Batch messages for future processing: Schedule multiple entries to be added to a database.

FIFO Queues

High Throughput: By default, FIFO queues support up to 300 messages per second (300 send, receive, or delete operations per second). When you batch 10 messages per operation (maximum), FIFO queues can support up to 3,000 messages per second. To request a quota increase, file a support request.

Exactly-Once Processing: A message is delivered once and remains available until a consumer processes and deletes it. Duplicates aren't introduced into the queue.

First-In-First-Out Delivery: The order in which messages are sent and received is strictly preserved (i.e. First-In-First-Out).

FIFO queues are designed to enhance messaging between applications when the order of operations and events is critical, or where duplicates can't be tolerated, for example:

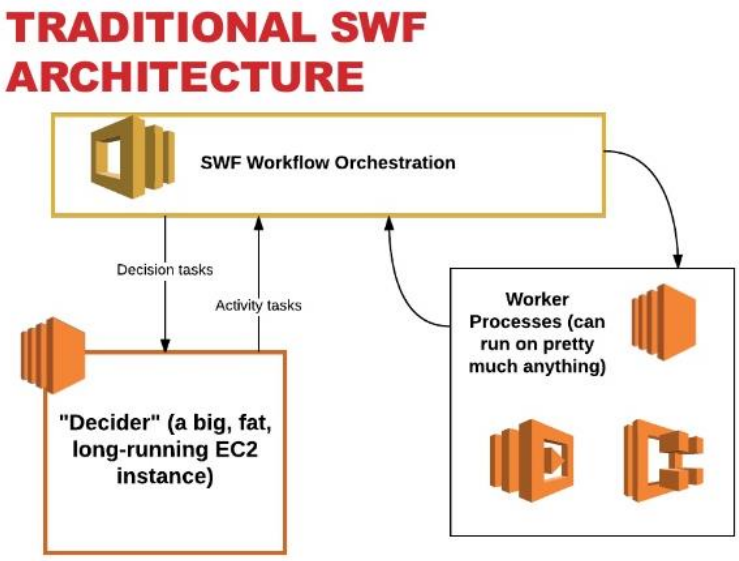
- Ensure that user-entered commands are executed in the right order.
- Display the correct product price by sending price modifications in the right order.
- Prevent a student from enrolling in a course before registering for an account.

Simple WorkFlow (SWF)

Amazon Simple Workflow Service (SWF) is a web service that makes it easy to coordinate work across distributed application components. Amazon SWF enables applications for a range of use cases, including media processing, web application back-ends, business process workflows, and analytics pipelines, to be designed as a coordination of tasks. Tasks represent invocations of various processing steps in an application which can be performed by executable code, web service calls, human actions, and scripts.

The coordination of tasks involves managing execution dependencies, scheduling, and concurrency in accordance with the logical flow of the application. With Amazon SWF, developers get full control over implementing processing steps and coordinating the tasks that drive them, without worrying about underlying complexities such as tracking their progress and keeping their state. Amazon SWF also provides the AWS Flow Framework to help developers use asynchronous programming in the development of their applications. By using Amazon SWF, developers benefit from ease of programming and have the ability to improve their applications' resource usage, latencies, and throughputs.

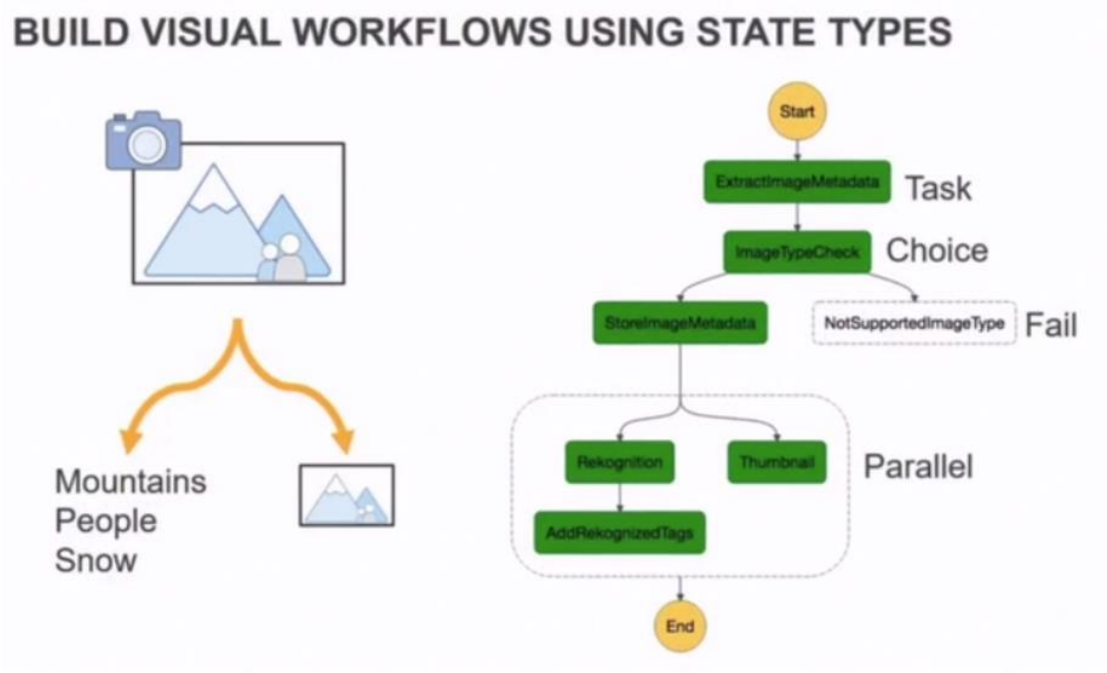
- SWF defines the sequence of events required to achieve a workflow.
- Workflow is a set of activities that result in a desired objective, and it includes the logic required to complete a process.
- The logic that controls the activities, where the decider function determines the best workflow.
- SWF is used in decoupled applications.
- SWF operates in a domain. SWF domain is a created logical boundary in SWF to constrain the scope of the activities.



Step Functions

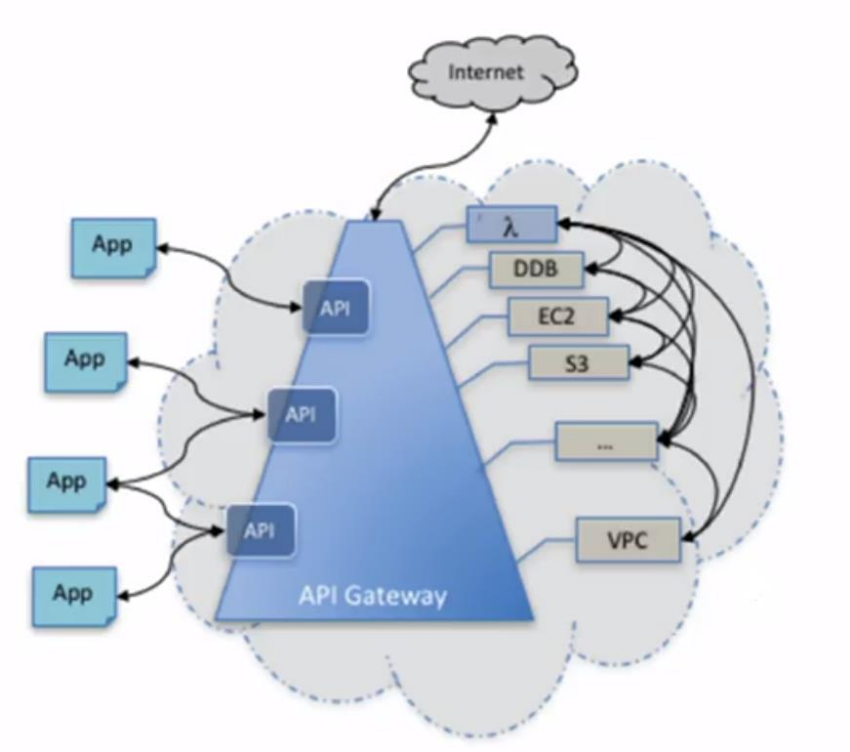
Step Functions is a reliable way to coordinate components and step through the functions of your application. Step Functions provides a graphical console to arrange and visualize the components of your application as a series of steps. This makes it simple to build and run multi-step applications.

- Step functions are positioned to replace SWF. AWS recommends using them instead of SWF workflows.
- Step functions use **state machines** instead of decider, activity tasks and worker tasks in SWF.
- Step functions contains tasks and choices. A **task** is a single unit of work and **choices** provides branching logic as shown in the below diagram.
- Step functions now supports nested workflows.



API Gateway

Amazon API Gateway helps developers to create and manage APIs to back-end systems running on Amazon EC2, AWS Lambda, or any publicly addressable web service. With Amazon API Gateway, you can generate custom client SDKs for your APIs, to connect your back-end systems to mobile, web, and server applications or services.



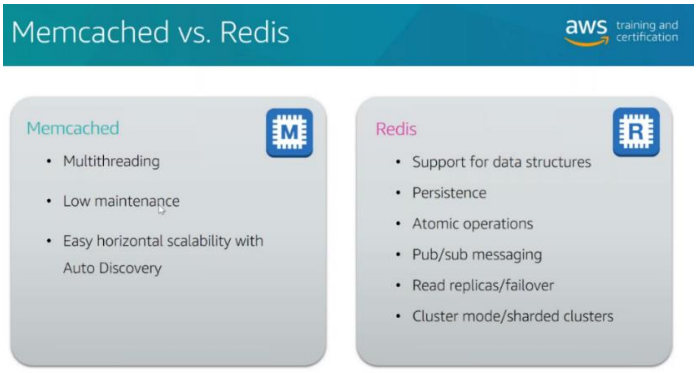
- You can create, publish, maintain, monitor and secure APIs with the API Gateway.
- API Gateway APIs interact with AWS services, external web services and data stored in AWS.

Useful Comparisons

CloudFront vs. ElastiCache

ElastiCache	CloudFront
ElastiCache uses redis and memcached to improve the performance of web applications by allowing you to retrieve information from fast, managed, in-memory data stores, instead of relying entirely on slower disk-based databases. **Always remember that ElastiCache sits in front of a Database**.	CloudFront is a global content delivery network (CDN) service that accelerates delivery of your websites, APIs, video content or other web assets.
The in-memory caching provided by ElastiCache can be used to significantly improve latency and throughput for many read-heavy application workloads (such as social networking, gaming, media sharing and Q&A portals) or compute-intensive workloads (such as a recommendation engine). In-memory caching improves application performance by storing critical pieces of data in memory for low-latency access. Cached information may include the results of **I/O-intensive database queries** or the results of computationally-intensive calculations. Just to confuse things a little more -- ElastiCache manages Redis like a relational database. Redis ElastiCache clusters are managed as stateful entities that include failovers in a similar same way as you would with Amazon RDS M-Z replication.	Amazon CloudFront supports all files that can be served over HTTP. This includes dynamic web pages, such as HTML or PHP pages, any popular static files that are a part of your web application, such as website images, audio, video, media files or software downloads. For on-demand media files, you can also choose to stream your content using RTMP delivery. Amazon CloudFront also supports delivery of live media over HTTP. Also, you can place CloudFront in front of dynamic content, such as web apps. CloudFront only caches rendered page output. In web apps, games, and mobile apps, it's very common to have thousands of fragments of data, which are reused in multiple sections of the app. CloudFront is a valuable component of scaling a website especially for geo-location workloads and queries, but it does not obviate the need for application caching.

Memcached vs. Redis



CloudTrail vs. CloudWatch vs. AWS Config

CloudTrail	CloudWatch	AWS Config
Definition CloudTrail keeps detailed logs of every read or write action that occurs against your AWS resources giving you a trail that includes what happened, who did it, when and even their IP address.	**Definition** CloudWatch Collects numeric performance metrics from AWS and non-AWS resources such as on-premises servers. It also collects and stores log files from these resources and lets you search them easily, and it provides alarms that can send you a notification or take action when a metric crosses a threshold.	**Definition** AWS Config tracks how your AWS resources are configured and how they change over time. You can view how your resources are related to one another and how they were configured at any time in the past. You can also compare your resource configurations against a baseline that you define and have AWS Config alert you when a resource fails out of compliance.
Main Features • Always on, enabled automatically from day one. • Keeps data for logs for the last 90 days (default). • Search and download event history • Multi-region configuration • Log file integrity validation. • Log file encryption • Stores data events and management events • Integrates with CloudWatch, Lambda,	**Main Features** • Easily collect and store logs • Built-in Metrics • Custom Metrics • Collect and aggregate container metrics and logs • Unified operational view dashboard • High resolution alarms • Logs and Metrics • Act: AutoScaling, Automate responses to operational changes • Custom metrics • Log analytics • Integrate with IAM for better compliance and security.	**Main Features** • Configuration history of AWS resources • Configuration history of software • Resource Relationship Tracking • Configurable and customizable rules • Conformance packs • Mutli-account, multi-region data aggregation • Extensibility • Configuration and snapshots • Cloud Governance dashboard • Integrates with third-party ITSM/ITOM software, Cloudtrail, AWS System Manager, AWS Organizations
Pricing Free Tier: You can view, filter, and download the most recent 90 days of your account activity for all management events in supported AWS services free of charge. **Paid Tier charges** Once a CloudTrail trail is set up, Amazon S3 charges apply based on your usage, since AWS CloudTrail delivers logs to an S3 bucket. Typical Amazon S3 charges are less than $3 per month for most accounts.	**Pricing Free Tier:** • Basic Monitoring Metrics (at 5-minute frequency) • 10 Detailed Monitoring Metrics (at 1-minute frequency) • 1 Million API requests (not applicable to GetMetricData and GetMetricWidgetImage) • 3 Dashboards • 10 Alarm • Log size up to 5 GB **Paid tier** charges per metric, alamrs, events and dashboards	**Pricing Free Tier:** Not Eligible for free tier. With AWS Config, you are charged based on the number of configuration items recorded, the number of active AWS Config rule evaluations and the number of conformance pack evaluations in your account. You pay $0.003 per configuration item recorded in your AWS account per AWS Region.

SQS vs. SNS

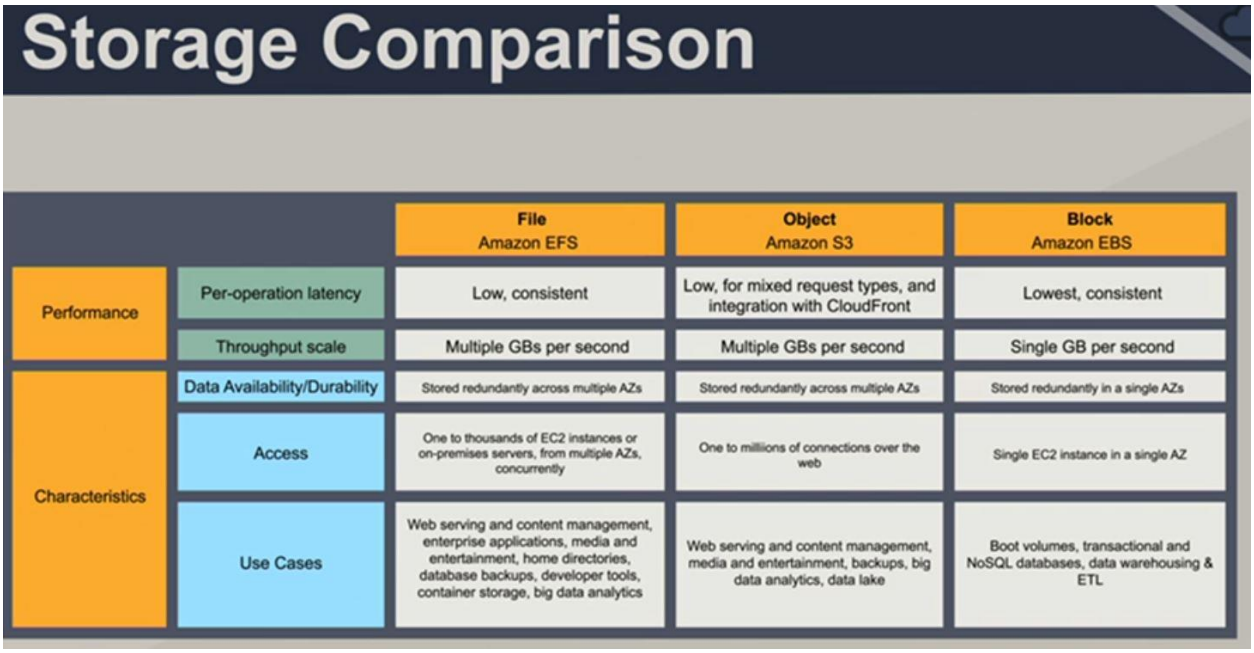
Parameter	SQS	SNS
Description	message queue service (poll)	pub-sub messaging service (push)
General features	Message retention, message locking, Dead Letter Queues (DLQ)	Integration with SMS, HTTP, Email
Send message to multiple functions	No	Yes
Max message size	256 KB	256 KB
Batch messages publish	Yes	No
Messages per second	Standard queues support a nearly unlimited number of transactions	Depends on the region (30,000 in us-east-1). Can be increased as it is a soft limit.
Monthly pricing 100 messages/seconds each message is 128KB	Requests: 100msgs x 60s x 60m x 24h x 30d x $0.4/mil = **$103.6** Data: 100msgs x 128KB x 60s x 60m x 24h x 30d = 31,640GB * $0.09/GB = **$2847**	Requests: 100msgs x 60s x 60m x 24h x 30d x $0.5/mil = **$129.6** Data: 100msgs x 128KB x 60s x 60m x 24h x 30d = 31,640GB * $0.09/GB = **$2847**

NACLs vs. Security Groups

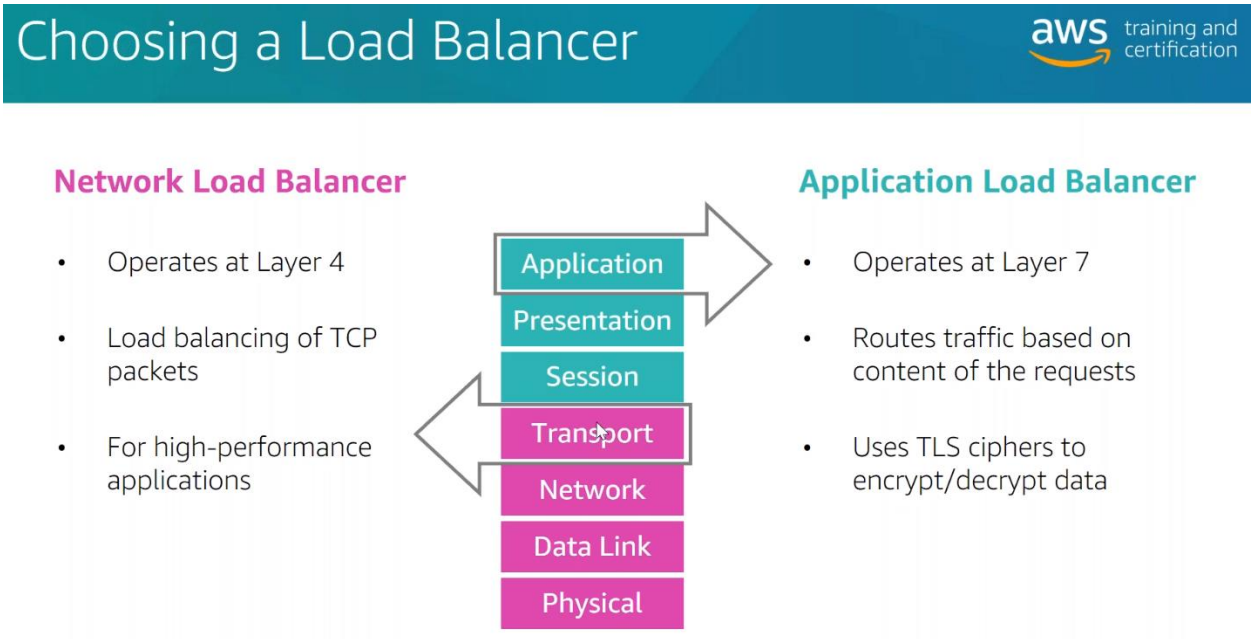
I am adding this one more time in this section, just to highlight its importance for the exam.

	Security Group	Network ACLs
Application	Associated to an elastic network interface	Associated to a subnet
Rules	Supports Allow rules only	Supports Allow rules and Deny rules
State	Stateful	Stateless
Rules Logic	All rules are evaluated before deciding whether to allow traffic	All rules are processed in order when deciding whether to allow traffic
Configuration	Needs to be manually configured on the ENI to allow traffic	Configured at the subnet level and automatically applied to all instances
Default Access	Deny all ingress traffic/ Allow all egress traffic	Allow all ingress/egress traffic

Storage Comparison



Load Balancers



Read more on: https://aws.amazon.com/elasticloadbalancing/features/?nc=sn&loc=2

Cloud HSM vs. AWS KMS

Security Controls/Features	AWS CloudHSM	AWS KMS
Tenancy	Single-tenant	Multi-tenant
Compliance	FIPS 140-2 Level 3	FIPS 140-2 Level 2
Master Key	Managed by customer only	Managed by AWS KMS
Ciphers	Symmetric or asymmetric encryption	Symmetric encryption only
High Availability	Managed by customer	Managed by AWS
Regional Availability	Select regions	All regions
Pricing	Hourly	By API calls and keys

How does AWS KMS compare to AWS CloudHSM?

AWS CloudHSM provides you with a FIPS 140-2 Level 3 overall validated single-tenant HSM cluster in your Amazon Virtual Private Cloud (VPC) to store and use your keys. You have exclusive control over how your keys are used via an authentication mechanism independent from AWS. You interact with keys in your AWS CloudHSM cluster similar to the way you interact with your applications running in Amazon EC2. You can use AWS CloudHSM to support a variety of use cases, such as Digital Rights Management (DRM), Public Key Infrastructure (PKI), document signing, and cryptographic functions using PKCS#11, Java JCE, or Microsoft CNG interfaces.

AWS KMS allows you to create and control the encryption keys used by your applications and supported AWS services in multiple regions around the world from a single console. The service uses an FIPS HSM that has been validated under FIPS 140-2, or are in the process of being validated, to protect the security of your keys. Centralized management of all your keys in AWS KMS lets you enforce who can use your keys under which conditions, when they get rotated, and who can manage them. AWS KMS integration with AWS CloudTrail gives you the ability to audit the use of your keys to support your regulatory and compliance activities. You interact with AWS KMS from your applications using the AWS SDK if you want to call the service APIs directly, via other AWS services that are integrated with AWS KMS or by using the AWS Encryption SDK if you want to perform client-side encryption.

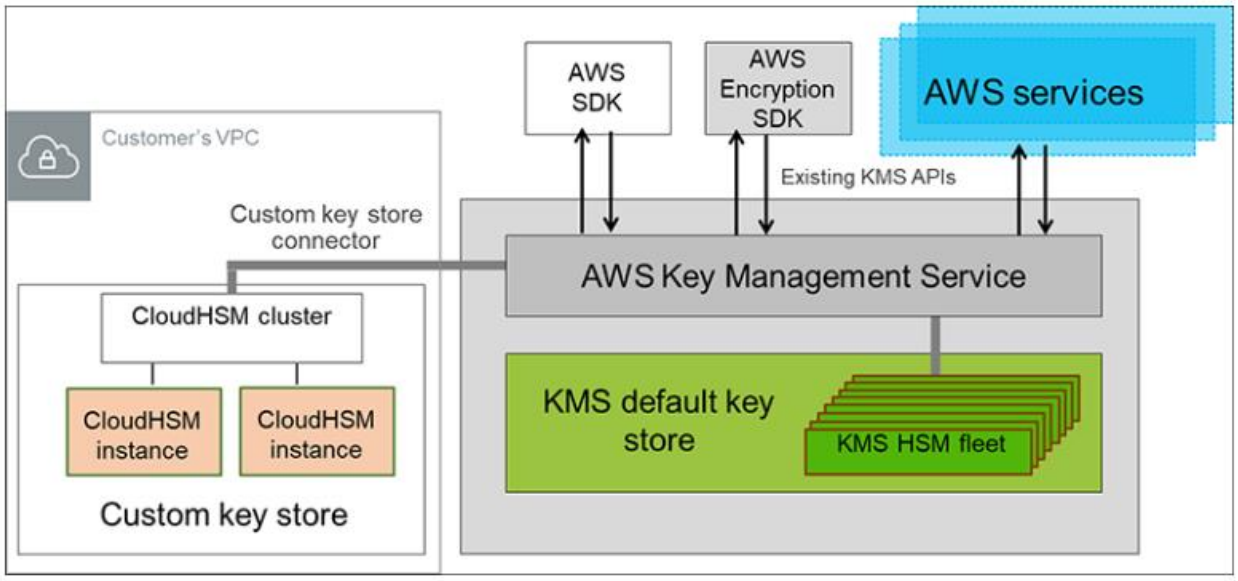
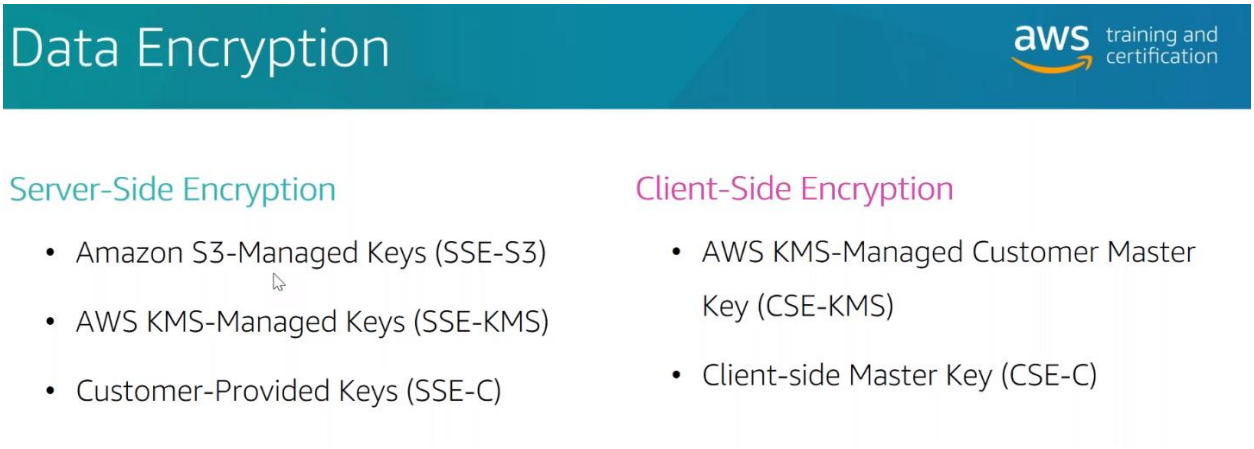


Figure 1: A cluster of two CloudHSM instances is connected to KMS to create a customer controlled key store

Server-Side Encryption vs. Client-Side Encryption



Launch Template vs. Launch Configuration

Both Launch Templates and Launch configurations can be used to create an Auto Scaling group.

Launch template is similar to launch configuration, which usually Auto Scaling group uses to launch EC2 instances. However, defining a launch template instead of a launch configuration allows you to have multiple versions of a template.

AWS recommend that we should use launch templates instead of launch configurations to ensure that we can leverage the latest features of Amazon EC2, such as T2 Unlimited instances.

Reference: https://docs.aws.amazon.com/autoscaling/ec2/userguide/LaunchTemplates.html

CloudFormation vs. Elastic Beanstalk

How is AWS CloudFormation different from AWS Elastic Beanstalk?

These services are designed to complement each other. AWS Elastic Beanstalk provides an environment to easily deploy and run applications in the cloud. It is integrated with developer tools and provides a one-stop experience for you to manage the lifecycle of your applications. AWS CloudFormation is a convenient provisioning mechanism for a broad range of AWS and third party resources. It supports the infrastructure needs of many different types of applications such as existing enterprise applications, legacy applications, applications built using a variety of AWS resources and container-based solutions (including those built using AWS Elastic Beanstalk).

AWS CloudFormation supports Elastic Beanstalk application environments as one of the AWS resource types. This allows you, for example, to create and manage an AWS Elastic Beanstalk–hosted application along with an RDS database to store the application data. In addition to RDS instances, any other supported AWS resource can be added to the group as well.

CodeCommit Differencing vs. S3 Versiomning

How does AWS CodeCommit compare to a versioned S3 bucket?

AWS CodeCommit is designed for collaborative software development. It manages batches of changes across multiple files, offers parallel branching, and includes version differencing (“diffing”). In comparison, Amazon S3 versioning supports recovering past versions of individual files but doesn’t support tracking batched changes that span multiple files or other features needed for collaborative software development.

S3 Bucket Vs. S3 Access Point

What is the difference between a bucket and an access point?

A bucket is the logical storage container for your objects while an access point provides access to the bucket and its contents. An access point is a separate Amazon resource created for a bucket with an Amazon Resource Name (ARN), hostname (in the format of https://[access_point_name]-[account ID].s3-accesspoint.[region].amazonaws.com), an access control policy, and a network origin control.

Multi-AZ deployments, multi-region deployments, and read replicas

Amazon RDS Multi-AZ deployments complement multi-region deployments and read replicas. While all three features increase availability and durability by maintaining additional copies of your data, there are differences between them:

Multi-AZ deployments	Multi-Region deployments	Read replicas
Main purpose is high availability	Main purpose is disaster recovery and local performance	Main purpose is scalability
Non-Aurora: synchronous replication; Aurora: asynchronous replication	Asynchronous replication	Asynchronous replication
Non-Aurora: only the primary instance is active; Aurora: all instances are active	All regions are accessible and can be used for reads	All read replicas are accessible and can be used for readscaling
Non-Aurora: automated backups are taken from standby; Aurora: automated backups are taken from shared storage layer	Automated backups can be taken in each region	No backups configured by default
Always span at least two Availability Zones within a single region	Each region can have a Multi-AZ deployment	Can be within an Availability Zone, Cross-AZ, or Cross-Region
Non-Aurora: database engine version upgrades happen on primary; Aurora: all instances are updated together	Non-Aurora: database engine version upgrade is independent in each region; Aurora: all instances are updated together	Non-Aurora: database engine version upgrade is independent from source instance; Aurora: all instances are updated together
Automatic failover to standby (non-Aurora) or read replica (Aurora) when a problem is detected	Aurora allows promotion of a secondary region to be the master	Can be manually promoted to a standalone database instance (non-Aurora) or to be the primary instance (Aurora)

You can combine Multi-AZ deployments with other Amazon RDS features to enjoy the benefits of each. For example, you can configure a source database as Multi-AZ for high availability and create a read replica (in Single-AZ) for read scalability. Or you can use Aurora Global Database to replicate data from your Multi-AZ Aurora deployment into additional regions.

With RDS for MySQL, MariaDB, PostgreSQL, and Oracle, you can also set the read replica as Multi-AZ, allowing you to use the read replica as a DR target. When you promote the read replica to be a standalone database, it will already be Multi-AZ enabled.

Support Plans Comparison

	DEVELOPER	BUSINESS	ENTERPRISE
Use Case	Recommended if you are experimenting or testing in AWS.	Recommended if you have production workloads in AWS.	Recommended if you have business and/or mission critical workloads in AWS.
AWS Trusted Advisor Best Practice Checks	7 Core checks	Full set of checks	Full set of checks
Architectural Guidance	General	Contextual to your use-cases	Consultative review and guidance based on your applications
Technical Account Management	X	X	Designated Technical Account Manager (TAM) to proactively monitor your environment and assist with optimization.
Training	X	X	Access to online self-paced labs
Account Assistance	X	X	Concierge Support Team
Enhanced Technical Support	Business hours** email access to Cloud Support Associates. Unlimited cases / 1 primary contact	24x7 phone, email, and chat access to Cloud Support Engineers Unlimited cases / unlimited contacts (IAM supported)	24x7 phone, email, and chat access to Cloud Support Engineers Unlimited cases / unlimited contacts (IAM supported)
Programmatic Case Management	X	AWS Support API	AWS Support API
Third-Party Software Support	X	Interoperability & configuration guidance and troubleshooting	Interoperability & configuration guidance and troubleshooting
Proactive Programs	X	Access to Infrastructure Event Management for additional fee.	• Infrastructure Event Management • Well-Architected Reviews • Operations Reviews • Technical Account Manager (TAM) coordinates access to programs and other AWS experts as needed.

While on the **Basic Plan** (free), you're given access only to publicly available documentation (white papers, tutorials, and support forums).

Monthly pricing for Amazon Paid Support Plans:

Developer	**Business**	**Enterprise**
Greater of $29 or… 3% of monthly usage	Greater of $100 or… 10% of usage up to $10,000 7% of usage up to $80,000 5% of usage up to $250,000 3% of usage over $250,000	Greater of $15,000 or… 10% of usage for the first $0-$150,000 7% of usage from $150,000-$1,000,000 3% of usage over $1,000,000

- **Knowledge Center** is basically a frequently asked questions (FAQ) page that has a lot of information collected there.

Random but Important

The following information bullet-points were extracted from practice test answers to review questions.

- S3 and S3-IA have the same durability.
- The limit of individual S3 object is **5 TB**.
- AWS Console and your account will show you all S3 buckets at all times.
- AMIs are **not** cross-region.
- While S3 allows for 0-byte objects and charges as such, S3-IA charges all object as if they are at least 128 KB in size.
- EBS **root** volumes are terminated when the associated instance is terminated. However, this is only the default value. You can use the AWS CLI or console to set the root volume to persist after instance termination.
- Additional EBS volumes attached to an instance **won't** be deleted when the instance is terminated.
- EBS volumes are backed up to S3 incrementally.
- EBS volumes can only attach to a single instance at one time. EFS, on the other hand, can be used by multiple instances at the same time.
- All instances and most services in AWS provide tagging for metadata.
- Every EC2 instance provide the option to specify an availability zone, which is user configurable.
- **Spread** placement groups can be placed across multiple availability zones. **Cluster** placement groups cannot, (When used alone, placement groups usually refers to cluster placement groups).
- A placement group is concerned primarily with network throughput and reducing latency among EC2 instances within a single availability zone.
- Spread placement groups can span availability zones and support up to seven (7) instances per zone.
- Spread placement groups primarily offer reduced network lag between instances. They allow for cross-VPC spanning of instances.
- VPN connections between an on-premises site and AWS consist of a customer gateway on the customer side and a virtual private gateway on the AWS side.
- Both File and Volume gateways offer solutions for connecting to cloud-based storage.
- A volume gateway stores data at the on-premises data store and backs up to S3 asynchronously to support disaster recovery.
- A cached volume gateway stores the most commonly accessed data locally while keeping the entire dataset in S3.
- S3 does use availability zones to store objects in buckets, you don't choose the availability zone yourself. Even S3 One Zone-IA doesn't allow you to specify the AZ for use.
- S3 does not provide SSH or SFTP access, nor standard FTP access. You can access your data through the AWS console and through a REST interface via HTTP.
- S3 is built to automatically scale in times of heavy application usage.
- AWS account can create up to 100 buckets.

- Presigned URLs are not tied to specific AWS Services. They are like any regular URL, except that the creator can associate permissions and a timeout with the URL.
- In S3, all regions have eventual consistency for overwrite PUTs and DELETEs.
- New objects uploaded via PUT are subject to read after write consistency. Overwrite PUTs use the eventual consistency model.
- S3 used to have a hard limit of 100 PUTs per second, but that limit has been raised in 2016 to 3500 PUTs per second.
- S3 supports two styles of bucket URL: virtual-hosted-style and path-style URLs
    1. http://bucket.s3-aws-region.amazonaws.com ← Virtual-hosted-style
    2. https://s3-aws-region.amazonaws.com/bucket-name ← Path-Style
- MFA delete is the absolute best means of ensuring that objects are not accidentally deleted.
- MFA Delete applies to deleting objects, not buckets. Deleting the object's metadata while leaving the object intact doesn't require MFA Delete.
- All Amazon-specific request headers begin with **x-amz**.
- Once enabled, it is NOT possible to disable versioning on an S3 Bucket it can only be suspended.
- Snowball can serve as both an import and export device, both to and from S3.
- Elastic Beanstalk is focused on code deployment.
- In general, Node.js, JavaScript, Java, PHP, and Perl are pretty commonly support programming language in AWS.
- EFS, Elastic File System, provides scalable storage accessible from multiple compute instances.
- Automated backups are turned on by default.
- RDS provides for SQL interaction as well as access through the RDS web APIs. RDS instances do NOT allow access via SSH or RDP.
- RDS allows backup retention periods up to 35 days, no longer.
- AWS provides up to five read replicas for a single database instance.
- Dynamo DB does use SSD drives, and it is spread across three geographically distinct data centers.
- SES is Simple Email Service and is used for sending and receiving emails for AWS applications and services.
- Default VPC is public.
- **Five (5) VPCs** are allowed per region per account.
- You can create 200 subnets per VPC.
- You can create up to 4 secondary CIDR blocks in addition to the primary. Total 5 CIDR blocks limit.
- By default, you are allowed 5 static IP addresses per region.
- Default VPC get a subnet automatically as well as internet gateway, route table, NACL, and a security group.
- A VPC can peer with unlimited other VPCs.
- Default VPCs get a /16 CIDR block assigned to them.
- All instances in the default VPC get a public and private IP address by default.
- The default subnet is each AZ is a /20

- **Default VPC** does have an internet gateway attached to it, but **custom VPCs do NOT**.
- Instances in any non-default VPC need to be made public via an elastic OR public IP and the VPC itself needs an internet gateway to be made public.
- A VPC endpoint can connect to S3 or Dynamo DB.
- A VPC endpoint comes in two flavors: an **interface endpoint** and a **gateway endpoint**.
- Security groups only provide ALLOW rules, all other traffic is automatically denied.
- For all new AWS accounts, 20 instances are allowed per region. This limit can be raised if requested.
- Cloud watch doesn't provide memory usage by default.
- Currently, read replicas in RDS are only supported by MariaDB, MySQL and PostgreSQL.
- A multi-AZ setup is focused on disaster recovery and fault tolerance, while read replicas provide performance and scalability.
- Read replicas are updated via asynchronous replication while Multi-AZ deployment replication is done synchronously.
- Read replicas can be in a different region than the primary instance.
- You can manually promote a read replica instance to a stand-alone instance if you have to.
- For MySQL, MariaDB and Postgre, you can have up to five (5) replicas at a time for a single instance. This limit cannot be raised.
- You must turn on automatic backups for the primary database instance to enable read replicas.
- No backup is taken automatically for any EC2 instance.
- Instances can have up to 28 attachments. Assuming one of those attachments is the network interface, the other 27 may be all EBS volumes or other elements. Meaning, a single instance can support a maximum of 27 EBS volumes.
- S3 RRS (reduced redundancy storage) is no longer recommended by AWS. S3-IA one zone is the alternative.
- All S3 and S3-IA data is stored in a single region and within at least three availability zones within that region.
- Redshift is well-suited for online analytics processing (OLAP). While databases under RDS are suitable for online transactional processing (OLTP).
- An SSD volume is best for transactional workloads.
- An HDD backed volume is best for streaming workloads, where throughput needs to be maximized over IOPS.
- HDDs are not available to use as boot volumes.
- Both ALBs and ELBs offer SSL termination. While an ALB is considered a better choice when considering the management of SSL certificates.
- Route 53 supports up to 50 domain names default. But this limit can be raised if requested.
- A CloudFront distribution is a collection of edge locations across the world.
- The default TTL for edge locations is 24 hours.
- When you create a CloudFront distribution, you register a domain name for your static and dynamic content.

- RDS is NOT a valid origin server for CloudFront. An origin is the location where content is stored, and from which CloudFront gets content to serve to viewers. To specify an origin:
    1. Use the `S3OriginConfig` type to specify an Amazon S3 bucket that is NOT configured with static website hosting.
    2. Use the `CustomOriginConfig` type to specify various other kinds of content containers or HTTP servers, including:
        1. An Amazon S3 bucket that is configured with static website hosting
        2. An Elastic Load Balancing load balancer
        3. An AWS Elemental MediaPackage origin
        4. An AWS Elemental MediaStore container
        5. Any other HTTP server, running on an Amazon EC2 instance or any other kind of host.
- A CloudFront distribution is the setup including your origin servers and how the content from those server is distributed via CloudFront.
- There is NO mechanism either in the AWS Console, or the AWS CLI to interact directly with files on CloudFront distributions or edge locations.
- An RTMP distribution is the Adobe Real-Time Messaging Protocol and is suitable for using S3 buckets as an origin server to serve streaming media.
- CloudFront supports both *web distributions* and *RTMP distributions*.
- ElasticCache is ideal for high-performance and real-time processing as well as heavy-duty business intelligence.
- Consider ElasticCache as only useful for storing transient data, it's NOT a persistent store.
- ElasticCache uses shards as a grouping mechanism for individual redis nodes.
- A storage gateway using ***cached volumes*** will cache frequently accessed data while storing the entire dataset on S3 in AWS.
- A storage gateway using ***stored volumes*** will store all data locally while backing up the data to S3 in AWS as well.
- Power users can work with managed services, but they cannot create (or otherwise manage) IAM users.
- You will ALWAYS need to provide non-root sign-in URLs for new users.
- IAM changes apply immediately to all users across the system.
- Programmatic access requires an access key ID and a secret access key pair.
- There are a number of valid scaling policies for Auto Scaling:
    1. Maintain current instance levels.
    2. Manual Scaling
    3. Schedule-based scaling
    4. Demand-based scaling
- InService and Standby are valid states for an instance.
- Security groups work for launch configurations just as they do with instances: You may use as many as you like.
- All custom NACLs disallow all inbound and outbound traffic by default. While a VPC's default NACL has a default "allow all" policy.

- SSE-S3 is low cost compared to KMS.
- There are four type of encryptions for an EBS volume:
    1. Data at rest on the volume
    2. Data moving between the volume and the instance
    3. Any snapshots created from the volume
    4. Any volumes created from those snapshots
- Encryption of a volume affects snapshots of the volume and instances created from that snapshot.
- You CANNOT encrypt running RDS instance. The ONLY way to encrypt an RDS instance is to encrypt it at creation time.
- The ONLY way to encrypt an EBS volume is to encrypt it at creation time.
- You can apply snapshots across accounts, but the default permissions do not allow this. So you have to modify those permissions.
- You can only create volumes from snapshots in the same region.
- NACLs are virtual firewalls, and they operate at the subnet and VPC level. While security groups operates on the individual instance level.
- Each rule in a NACL has a number, and those rules are evaluated using those number moving from low to high.
- A subnet can only be associated to a single NACL at a time.
- A subnet CANNOT span availability zones, it can only exist within a single AZ.
- For a single VPC, you can add one or more subnet to each availability zone within that VPC.
- A VPC spans all availability zones in a region. You must always select a region to create a VPC, and you must always provide a CIDR block.
- A VPC can have a single primary CIDR block assigned to it for IPv4 addresses and an option IPv6 CIDR block. While you can add secondary IPv4 blocks, you CANNOT add additional CIDR blocks for IPv6 at this time.
- Any subnet that routes traffic through an internet gateway is a public subnet.
- A VPN-only subnet routes traffic through a virtual private gateway rather than an internet gateway.
- When you launch an instance, you must specify an availability zone.
- A VPC endpoint is for attaching to AWS services and explicitly doesn't require an internet gateway.
- A bastian host, sometimes called *jump* server, is a server whose purpose is to provide access to a private network from an external network, such as the Internet.
- Internet gateways scale horizontally, not vertically.
- Internet gateways attach to VPCs and serve multiple subnets (if needed).
- A public subnet, by definition, is a subnet with an internet gateway attached.
- Instances launched into default subnets in the default VPC can automatically reach out to the public Internet.
- An egress-only gateway is for use with IPv6 traffic only.
- An elastic network interface is virtual and can have multiple IPv4 and IPv6 addresses as well as security groups, a MAC address, and a source/destination check flag.

- An Elastic network interface can only be attached to a single instance at one time but can be moved from one instance to another.
- Elastic network interfaces don't have routing tables.
- Elastic IP addresses are specifically for avoid being tied to specific instance.
- Elastic IP addresses are, by definition, an IP address that will not change.
- You can only assign a single role to an instance.
- The well-architected framework includes five areas for **security** in the cloud:
    1. Privilege management (IAM)
    2. Detective controls
    3. Infrastructure protection
    4. Data Protection
    5. Incident Response.
- AWS infrastructure operates at the VPC layer and is almost entirely virtual.
- While AWS uses the term *managed services* in lots of areas, that term is not used in the shared responsibility model as one of the core type of services.
- By default, newly created S3 buckets are private.
- Reserved instances are locked to the region in which they are created, meaning that they can't be moved.
- Anytime you're testing a new application, on-demand instance is a good choice.
- EBS is a much better choice than EFS for a single-instance application.
- Egress data transfers always has a cost associated with it, while ingress is always free. Transferring data across regions is treated the same as transfers to the internet.
- Elastic MapReduce (EMR), is a web service targeted at processing large amounts of data. It is optimized for this task and often provides cost savings over EC2 instances running similar processes.
- QuickSight is designed for combining data sources and then performing analytics and extracting insights.
- The easiest, most cost-effective option is to migrate directly from Oracle to PostgreSQL using DMS, the Database Migration Service.
- SWF tasks are assigned once and only once.
- AWS guarantee that all SQS messages will be delivered at least once, but the message may be delivered more than once.
- SNS is the Simple Notifications Service and functions like a mailer.
- SNS sends out notifications to subscribed listeners.
- SWF pushes messages as they arrive.
- SQS holds messages until the queue is polled.
- SNS and SWF operate on a push approach.
- Both SWF and SQS deliver a message at least once, but only SWF guarantees that a message will only be delivered a single time.
- **SWF = Tasks** **SQS = Messages** **SNS = Notifications**

- AWS doesn’t support IPv6 inter-region communication. This means that for IPv6 communication to work, the two VPCs must be in the same region. Then, you must ensure that both VPCs have IPv6 addresses and that routing is setup to use those addresses.
- Kinesis is built to handle a massive data stream.
- Containers (such as Docker) allow you to reduce startup times.
- AWS does not allow vulnerability scans to be run without advance notice. There are some preapproved scans using AWS-approved tools, but in general, you'll need to contact your AWS account manager or support team in advance of running vulnerability scans.
- Only the bucket owner of an S3 bucket can completely delete a file once versioning has been enabled.
- Scaling in is the process by which an Auto Scaling group removes instances.
- Auto Scaling groups scale in using a very specific set of criteria. Highest priority
    1. Availability zone with the most instances, then
    2. The age of the launch configuration of instances
    3. The nearness of instances to the next billing hour.
- Endpoints are created within you VPC and a PrivateLink allows for secure connection between VPCs, services, and applications in AWS.
- A gig**a**byte is 1 billion bytes, a gig**i**byte is 1,073,741,824. (Meaning 1000 = 1024).

Additional AWS Services

Elastic Transcoder

Amazon Elastic Transcoder is a highly scalable, easy to use and cost effective way for developers and businesses to convert (or “transcode”) video and audio files from their source format into versions that will playback on devices like smartphones, tablets and PCs.

So instead of converting files offline and uploading to S3, Elastic Transcoder can do this conversion right in the cloud and place transcoded files in their original bucket.

Terminology

- **Jobs**: Do the transcoding
- **Pipelines**: Queues to manage jobs
- **Presets**: Settings to convert media
- **Notifications**: SNS used to notify of job status

Amazon Translate

Amazon Translate is a Neural Machine Translation (MT) service for translating text between 55 supported languages.

- On-demand language translation.
- It can integrate into applications for localization.
- Encoder reads source text and decoder outputs translated text.

Elemental MediaStore

AWS Elemental MediaStore is a video origin and storage service that offers the performance, predictable low latency, and consistency required for delivery and processing workloads like live streaming video.

The service runs with the concepts of **containers**, **folders**, **endpoints**, **objects** and **policies**. So the containers and folders give me a hierarchy within which my video is stored. The endpoint is the source or origination. And then we have objects which basically are the video files. And the actual policies that control who can access this content.

Works best for live video streams. For storage-based video S3 is just fine.

Transcribe

Speech-to-text service that support audio and video.

One of the many usage is closed caption (subtitles) for videos.

Based on machine learning.

Integrates with Translate.

Rekognition

Amazon Rekognition is a service that makes it easy to add powerful visual analysis to your applications. **Rekognition Image** lets you easily build powerful applications to search, verify, and organize millions of images. **Rekognition Video** lets you extract motion-based context from stored or live stream videos and helps you analyze them.

Image and video analysis that can recognize people, speech and objects.

It can run against S3 buckets and provide enhanced search results based on analysis.

WorkSpaces

WorkSpaces is AWS desktop-as-a-service solution.

It provides virtual desktops in AWS, available in Linux and Windows.

Persistent storage on virtual D: drive that gets backup automatically.

AppStream

Amazon AppStream 2.0 is a fully managed application streaming service that provides users with instant access to their desktop applications from anywhere. AppStream 2.0 manages the AWS resources required to host and run your applications, scales automatically, and provides access to your users on demand.

A citrix-like solution commonly used for custom-developed apps.

CloudSearch

Amazon CloudSearch is a powerful indexing engine within AWS. It is a fully-managed service in the AWS Cloud that makes it easy to set up, manage, and scale a search solution for your website or application.

Useful when you have a lot of offline data that you want to bring to a central repository and make it searchable.

ElasticSearch

Amazon Elasticsearch Service is a managed service that makes it easy to deploy, operate, and scale Elasticsearch clusters in the AWS Cloud. ElasticSearch Scales out to a larger scale than cloud search.

For example, a real estate business can use Amazon Elasticsearch Service to help its consumers find homes in their desired location, in a certain price range from among millions of real-estate properties. You get access to all of Elasticsearch's search APIs, supporting natural language search, auto-completion, faceted search, and location-aware search.

Amazon Data Pipeline

AWS Data Pipeline is a web service that helps you reliably process and move data between different AWS compute and storage services, as well as on-premises data sources, at specified intervals.

AWS Glue

AWS Glue is a fully-managed, pay-as-you-go, extract, transform, and load (ETL) service that automates the time-consuming steps of data preparation for analytics. AWS Glue automatically discovers and profiles your data via the Glue Data Catalog, recommends and generates ETL code to transform your source data into target schemas, and runs the ETL jobs on a fully managed, scale-out Apache Spark environment to load your data into its destination. It also allows you to setup, orchestrate, and monitor complex data flows.

For example, imagine you've got a table with all of your customers in it, with their customer name, address, email address, phone numbers, and so forth. And then you've got another table with all the orders they've placed with you. And you want to take this, and between the two, come up with a new set of data, a set of data that has only the customers in it in the end that have placed more than five orders from you, totalling more than $3,000 in value. And now you've got a whole new table by extracting it out of the two, and then merging it into another, so you have this specific table that's created for high targeted marketing to repeat customers who've spent a significant amount with you. That's just one example of the kind of thing you could do. Obviously, you can completely transform data as well. But this is the concept of using an ETL tool, kind of like SQL Server integration services, if you've ever used that in Microsoft SQL Server.

QuickSight

Amazon QuickSight is a fast, cloud-powered business intelligence service that makes it easy to deliver insights to everyone in your organization.

Note that Signup is required, QuickSight it doesn’t come by default with your AWS subscription.

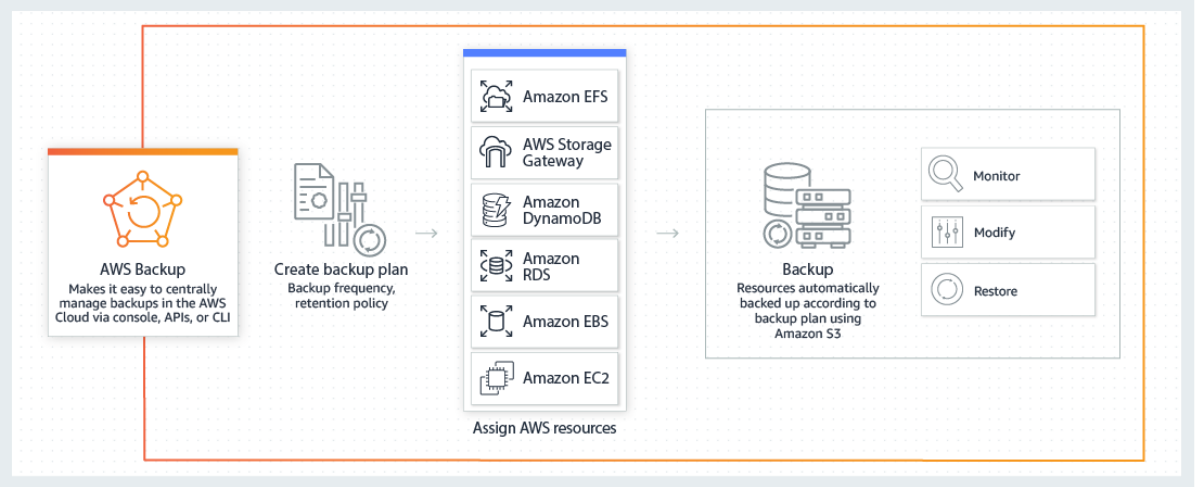
Amazon Athena

Amazon Athena is an interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL. Athena is serverless, so there is no infrastructure to setup or manage, and you can start analyzing data immediately. You don’t even need to load your data into Athena, it works directly with data stored in S3. To get started, just log into the Athena Management Console, define your schema, and start querying. Amazon Athena uses Presto with full standard SQL support and works with a variety of standard data formats, including CSV, JSON, ORC, Apache Parquet and Avro. While Amazon Athena is ideal for quick, ad-hoc querying and integrates with Amazon QuickSight for easy visualization, it can also handle complex analysis, including large joins, window functions, and arrays.

AWS Backup

Introduced in 2019, AWS Backup is a fully managed centralized backup service that makes it easy and cost-effective for you to back up your application data across AWS services in the AWS Cloud, helping you meet your business and regulatory backup compliance requirements. AWS Backup makes protecting your AWS storage volumes, databases, and file systems simple by providing a central place where you can configure and audit the AWS resources you want to backup, automate backup scheduling, set retention policies, and monitor all recent backup and restore activity.

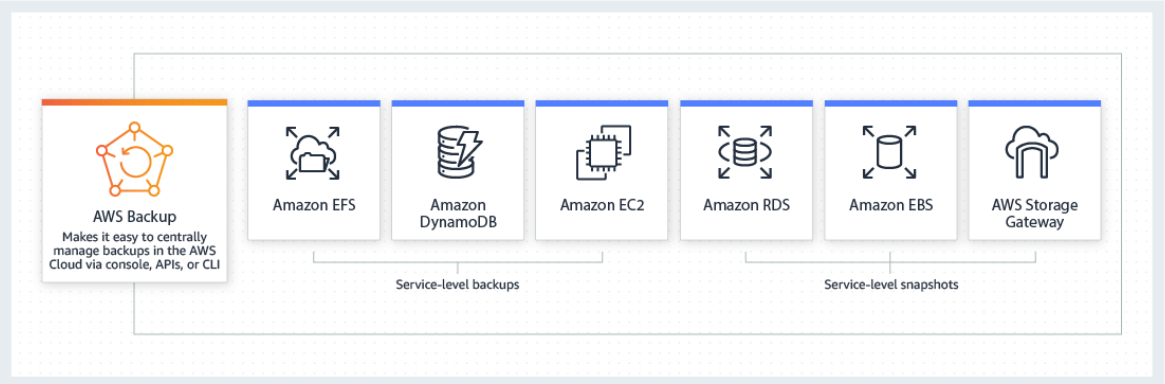
How it works



Use cases

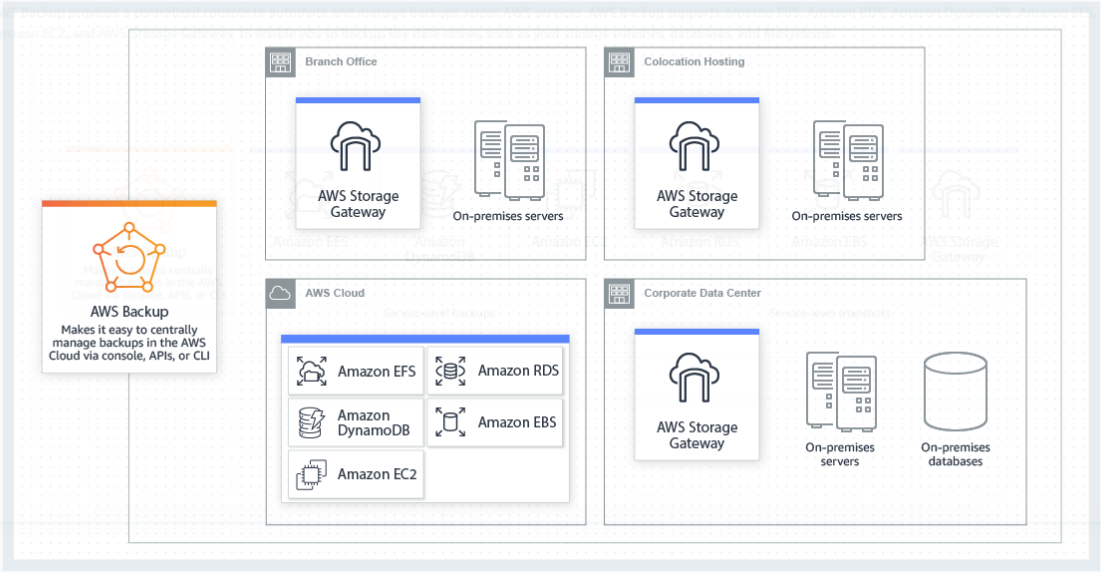
Cloud-native backup

AWS Backup provides a centralized console to automate and manage backups across AWS services. AWS Backup supports Amazon EBS, Amazon RDS, Amazon DynamoDB, Amazon EFS, Amazon EC2, and AWS Storage Gateway, to enable you to backup key data stores, such as your storage volumes, databases, and filesystems.



Hybrid backup

AWS Backup integrates with AWS Storage Gateway, a hybrid storage service that enables your on-premises applications to seamlessly use AWS cloud storage. You can use AWS Backup to back up your application data stored in AWS Storage Gateway volumes. Backups of AWS Storage Gateway volumes are securely stored in the AWS Cloud and are compatible with Amazon EBS, allowing you to restore your volumes to the AWS Cloud or to your on-premises environment. This integration also allows you to apply the same backup policies to both your AWS Cloud resources and your on-premises data stored on AWS Storage Gateway volumes.



Cost Explorer

AWS Cost Explorer give you granular detail about your spending before the end-of-month bill, along with recommendations to save money.

AWS cost explorer is disabled by default. Once enabled, it takes 24 hours to give you the initial analysis, but then provides up-to-date data.

AWS Budgets

AWS Budgets gives you the ability to set custom budgets that alert you when your costs or usage exceed (or are forecasted to exceed) your budgeted amount.

You can also use AWS Budgets to set reservation utilization or coverage targets and receive alerts when your utilization drops below the threshold you define. Reservation alerts are supported for Amazon EC2, Amazon RDS, Amazon Redshift, Amazon ElastiCache, and Amazon Elasticsearch reservations.

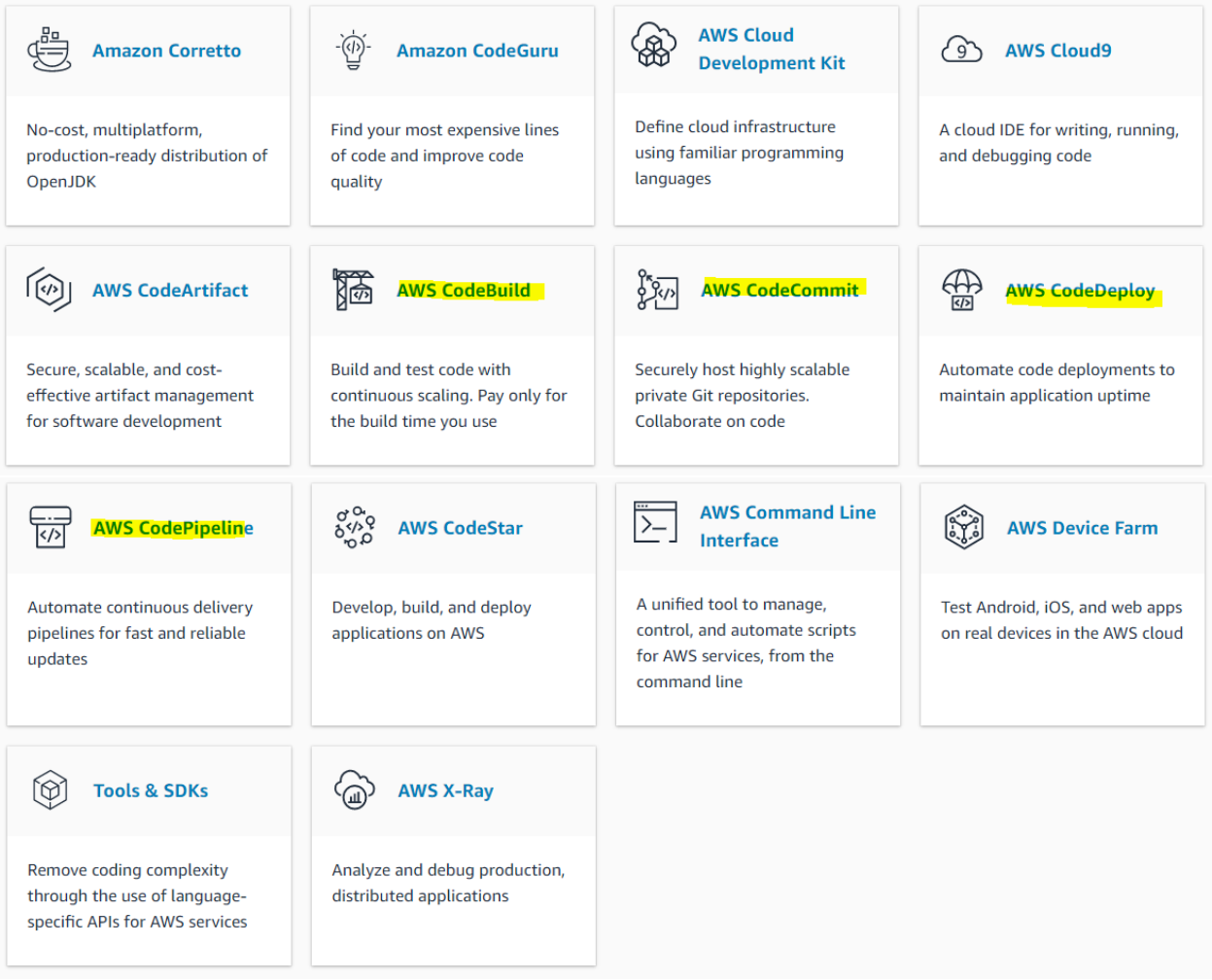
Cost & Usage Report

The AWS Cost & Usage Report contains the most comprehensive set of AWS cost and usage data available, including additional metadata about AWS services, pricing, and reservations (e.g., Amazon EC2 Reserved Instances (RIs)).

The AWS Cost & Usage Report lists AWS usage for each service category used by an account and its IAM users in hourly or daily line items, as well as any tags that you have activated for cost allocation purposes. You can also customize the AWS Cost & Usage Report to aggregate your usage data to the daily or hourly level.

Development Operations

DevOps is a separate certification track in AWS. However, as an AWS architect it is important to know the basics of DevOps and the tools used to support development operations. In this section, we are going to explore a series of development tools that integrate together in to host code, build, test and deploy applications in AWS.



CodeCommit

CodeCommit is about repositories. You've probably heard of Git repositories. If you haven't, it's basically a code management or sourcecode management solution. So you can store your sourcecode there. You can keep different versions of your sourcecode and manage it throughout its lifetime. That's the concept of CodeCommit, sourcecode management. You can really manage pretty much any kind of sourcecode in there the way you want.

AWS CodeCommit offers a number of features not offered by other Git source control systems:

- **Fully Managed** –AWS CodeCommit eliminates the need to host, maintain, backup, and scale your own source control servers.

- **Secure** –AWS CodeCommit automatically encrypts your files in transit and at rest. AWS CodeCommit is integrated with AWS Identity and Access Management (IAM), allowing you to assign user-specific permissions to your repositories.
- **Highly Available** – AWS CodeCommit is built on highly scalable, redundant, and durable AWS services such as Amazon S3 and Amazon DynamoDB.
- **Scalable** - AWS CodeCommit allows you store any number of files and there are no repository size limits.
- **Faster Development Lifecycle** - AWS CodeCommit keeps your repositories close to your build, staging, and production environments in the AWS cloud. This allows you to increase the speed and frequency of your development lifecycle.

CodeBuild

Instead of having to set up, patch, and maintain the build server software yourself, you can use CodeBuild's fully managed experience. You submit your build jobs to CodeBuild, and it runs them in temporary compute containers that are created fresh on every build and then discarded when finished. You don't need to manage build server hardware or software. CodeBuild also automatically scales to meet your build volume. It immediately processes each build you submit and can run separate builds concurrently, meaning your builds are never left waiting in a queue.

AWS CodeBuild is a fully managed continuous integration service in the cloud. CodeBuild compiles source code, runs tests, and produces packages that are ready to deploy. CodeBuild eliminates the need to provision, manage, and scale your own build servers. CodeBuild automatically scales up and down and processes multiple builds concurrently, so your builds don't have to wait in a queue. You can get started quickly by using CodeBuild prepackaged build environments, or you can use custom build environments to use your own build tools. With CodeBuild, you only pay by the minute.

You can initiate builds with AWS CodeBuild in several ways. For example, you can initiate builds in CodeBuild after connecting to AWS CodeCommit, GitHub, GitHub Enterprise, Bitbucket, or Amazon S3. You can also connect CodeBuild and your source repository with AWS CodePipeline, which automatically initiates a build every time you commit a change.

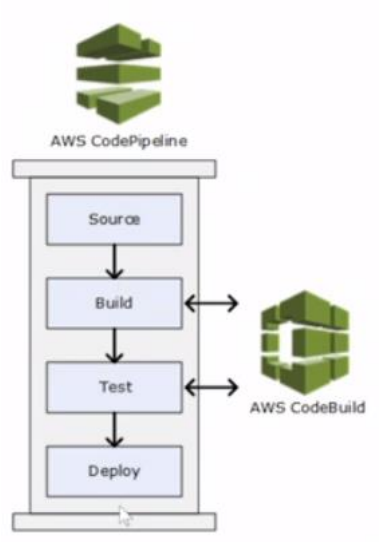
Not every language is supported; however, the widely used ones are Ruby, Python, Java, PHP, .net…etc.

CodeDeploy

AWS CodeDeploy is a service that automates code deployments to any instance, including Amazon EC2 instances and instances running on-premises. You can use AWS CodeDeploy to automate deployments, eliminating the need for error-prone manual operations, and the service scales with your infrastructure so you can easily deploy to one instance or thousands.

CodePipeline

AWS CodePipeline is a continuous delivery service that enables you to model, visualize, and automate the steps required to release your software. With AWS CodePipeline, you model the full release process for building your code, deploying to pre-production environments, testing your application and releasing it to production. AWS CodePipeline then builds, tests, and deploys your application according to the defined workflow every time there is a code change. You can integrate partner tools and your own custom tools into any stage of the release process to form an end-to-end continuous delivery solution.



Why should I use AWS CodePipeline?

By automating your build, test, and release processes, AWS CodePipeline enables you to increase the speed and quality of your software updates by running all new changes through a consistent set of quality checks.

What is continuous delivery?

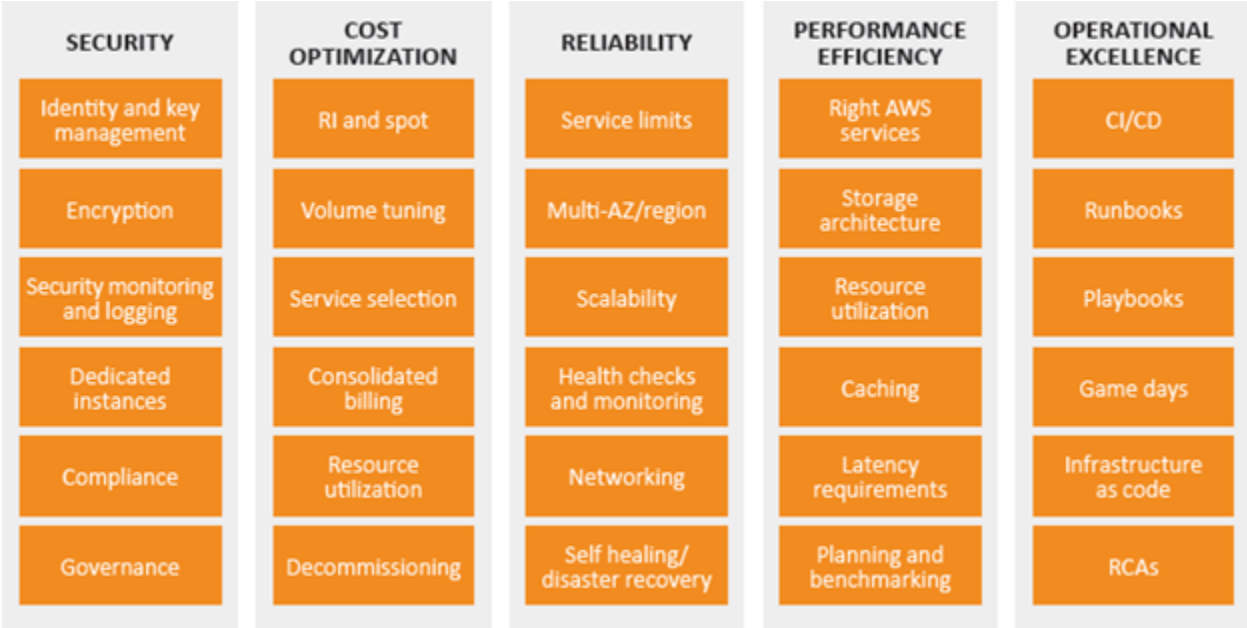
Continuous delivery is a software development practice where code changes are automatically built, tested, and prepared for a release to production. AWS CodePipeline is a service that helps you practice continuous delivery.

AWS X-RAY

There wouldn't be a good collection of developer tools if it doesn't have a debugger. A debugger is all about looking at your source code to see where there might be errors, finding problems in it, and helping you to resolve them. They should have trace capabilities. So for example you can track a variable all throughout its lifecycle and use within the application seeing what the contents of that variable actually are. And that is the job of AWS X-Ray, it is your debugger within the AWS cloud.

The Well-Architected Framework

The well-architected framework is a model for deployment of an entire solution that assists in accomplishing an end goal. It is important for the exam to read and understand the whitepaper documenting the framework (link here), this section provides some key points.



- AWS well-architected framework suggest **operational excellence**, **security**, **reliability**, **performance** and **cost** be addressed, AWS calls them the 5-pillars of the well-architected framework. Each pillar has its own whitepaper to read.
- The operational excellence process includes preparation, operation, and evolution.

- Resilient design results in reliability: the assurance that the system is there when you need it.
- Resilient design can be reached through implementation of data recovery, auto-scaling, and backups.
- To ensure reliability, test recovery procedures and implement automatic recovery whenever possible.
- Scaling horizontally, from one large system to multiple decoupled smaller systems, can increase reliability.
- Estimating capacity based on expected utilization is far better than guessing without metrics.
- Using a mutli-AZ database increases reliability and resiliency for data-centric solutions.
- Using ELB with web servers provides for improved operational performance and resiliency.

- Auto Scaling is the key to performant design in the cloud.
- Always remember that serverless architectures scales better than server-based architectures. Serverless architectures can increase performance as the process receives the performance it requires.

- With solutions in the cloud, you gain the scalability advantage it offers, with potential increased performance.
- Performance can be enhanced by deploying solutions into multiple regions, which results in the service being closer to end users.
- Using game days to experiment can increase performance. Such exercise involve testing different configurations to find the ultimate performing one.
- A primary key to performance is selecting the right class of EC2 instances.
- SNS messages can be used to automate notifications so that appropriate personnel are informed.
- Breaking S3 storage into departmental buckets can improve performance over placing all organizational data in a single bucket with folders (prefixes).

- Understand the shared responsibility model.
- Implementing IAM properly is the foundation of AWS security.
- Apply security at all layers: account, vpc, subnet, and instances….etc.
- Protect data in transit and at rest.
- Automate security best practices.
- Keep people away from data: meaning to organize access to data programmatically and not through direct access, and apply the principle of least privilege. This will protect you against accidental or intentional damage.
- Using CloudTrail, you can implement traceability so that actions are documented (logged).
- When implementing databases, the security features within the database should be used as well as AWS managed security features.
- Ensure that NACLs and security groups are configured properly on the VPC and instance network interfaces for web applications.

- Using Cost effective resources, matching supply with demand, and Optimizing over time, is what cost optimization is all about.
- AWS uses a consumption model for billing. Lower you consumption and you will lower your costs.
- The over efficiency of a solution determines the cost. Sometimes it is less expensive to decouple an application and sometimes it is not.
- Using AWS managed services is usually less expensive than implementing instances to perform the operations.

General Best Practices

- Design for Failures, examples: implement clusters in multiple availability zones, have proper backups, alternate AWS account (cold site) using CloudFormation templates.
- Implement elasticity using auto scaling, elastic load balancing, decoupled applications, run tasks in parallel.
- Learn and practice: Use AWS free tier account to play, build entire solutions, configure every option, tear down and start again.
- Try different solutions.

SAA-C02: Exam Tips

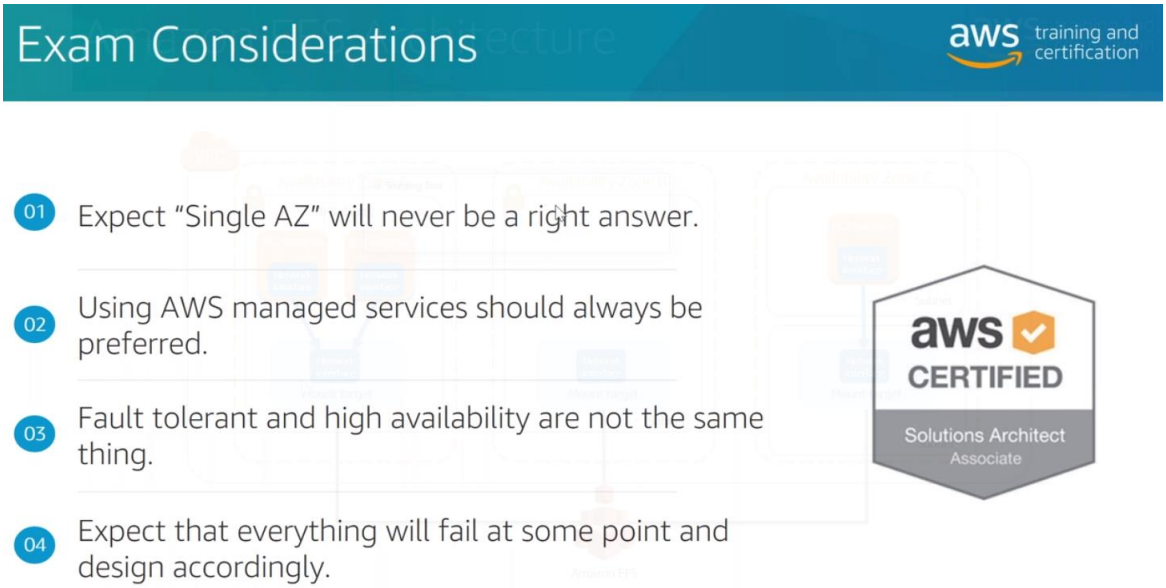
Recommended Knowledge

- Get hands-on experience designing available, cost-effective, fault-tolerant, and scalable distributed systems on AWS.
- Hands-on experience using compute, networking, storage, and database AWS services. Integration of S3 with other services is a significant part of the exam questions.
- Hands-on experience with AWS deployment and management services.
- Ability to identify and define technical requirements for an AWS-based application.
- Ability to identify which AWS services meet a given technical requirement.
- Knowledge of recommended best practices for building secure and reliable applications on the AWS platform.
- An understanding of the basic architectural principles of building in the AWS Cloud.
- An understanding of the AWS global infrastructure.
- An understanding of network technologies as they relate to AWS. VPC, subnets and routing has a significant part of the exam.
- An understanding of security features and tools that AWS provides and how they relate to traditional services.

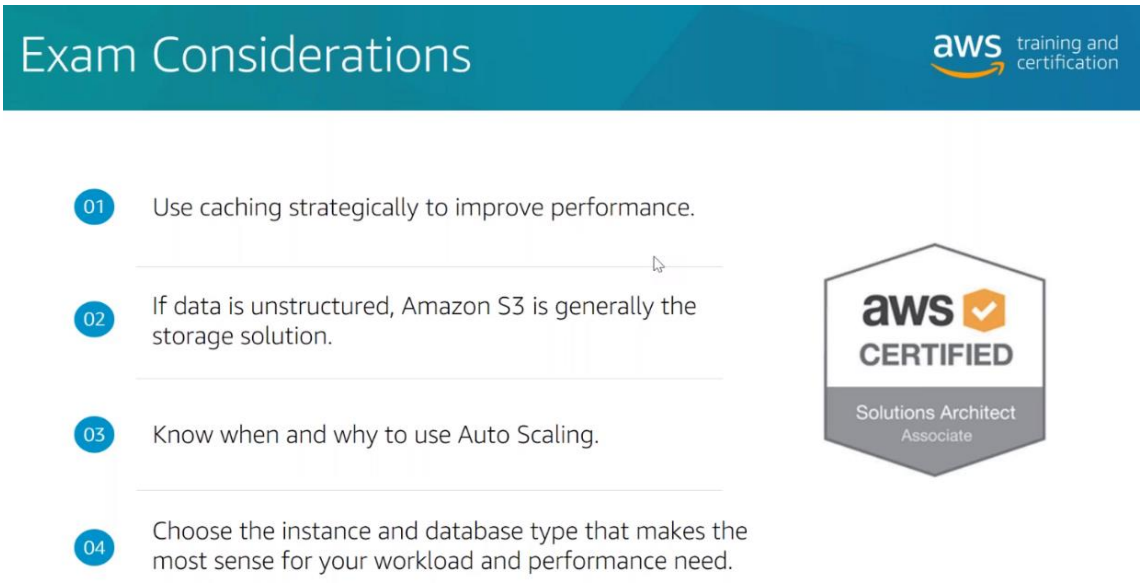
Notes from AWS Official Exam Readiness Webinar

- Exam questions used to be definitions-based, but now they are more scenario-based. Where AWS want to make sure that a solutions architect is ready to provide solutions for given scenarios.
- It is advisable to go through FAQs page of every core service: https://aws.amazon.com/faqs/ Many questions are derived from FAQs.
- Read case studies for core service on AWS websites.
- White paper are also another good source of information to prepare your for the exam: https://aws.amazon.com/whitepapers/?whitepapers-main.sort-by=item.additionalFields.sortDate&whitepapers-main.sort-order=desc
- Do some labs on: https://aws.amazon.com/training/self-paced-labs/ or https://amazon.qwiklabs.com/

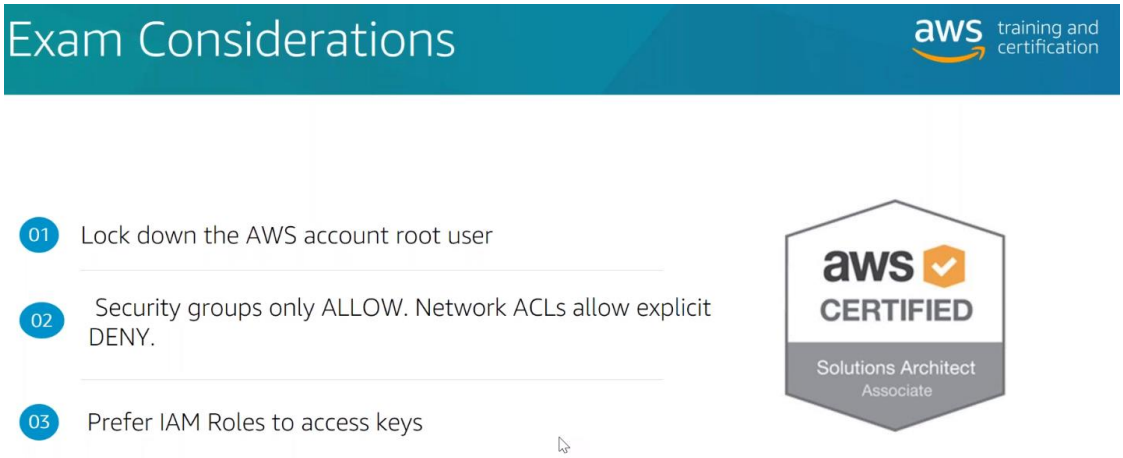
Domain 1: Designing Resilient Architecture



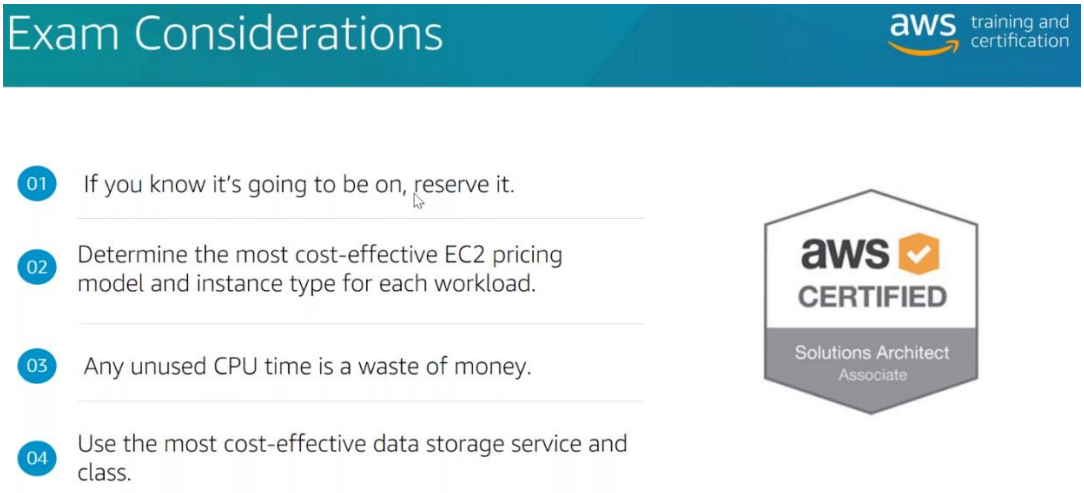
Domain 2: Performance Efficiency



Domain 3: Security



Domain 4: Cost Optimization



Useful links:

- Exam details: https://aws.amazon.com/certification/certified-solutions-architect-associate/
- Training Self-Paced Labs: https://aws.amazon.com/training/self-paced labs/
- Courses: https://aws.amazon.com/training/course-descriptions/
- Resources Whitepapers: https://aws.amazon.com/whitepapers/
- Architecture Center: https://aws.amazon.com/architecture/
- Documentation: https://docs.aws.amazon.com/index.html#lang/en_us