

Important Concepts [EC2] :

It mainly consists in the capability of

- Renting virtual machines (EC2)
- Storing data on virtual drives (EBS)
- Distributing load across machines (ELB)
- Scaling the services using an auto-scaling group (ASG)

EC2 User Data

It is possible to bootstrap our instances using an EC2 User data script.

bootstrapping means launching commands when a machine starts

That script is only run once at the instance first start

EC2 user data is used to automate boot tasks such as:

- Installing updates
- Installing software
- Downloading common files from the internet
- Anything you can think of
- The EC2 User Data Script runs with the root user

Instance Types:

m5.2xlarge

- m: instance class
- 5: generation (AWS improves them over time)
- 2xlarge: size within the instance class

- 1) **General Purpose** = E.g t2 micro that gives a balance between CPU , Memory and Networking

Use Cases: Web Servers & Code repositories

- 2) **Compute Optimized**

- **Great for compute-intensive tasks that require high performance processors:**
- Batch processing workloads
- Media transcoding
- High performance web servers
- High performance computing (HPC)

- Scientific modeling & machine learning • Dedicated gaming servers

3) Memory Optimized

Fast performance for workloads that process large data sets in memory

- Use cases:
- High performance, relational/non-relational databases
- Distributed web scale cache stores
- In-memory databases optimized for BI (business intelligence)
- Applications performing real-time processing of big unstructured data

4) Storage Optimized

Great for storage-intensive tasks that require high, sequential read and write access to large data sets on local storage

Use cases:

- High frequency online transaction processing (OLTP) systems
- Relational & NoSQL databases
- Cache for in-memory databases (for example, Redis)
- Data warehousing applications • Distributed file systems

. General Purpose Instances:

- These instances are suitable for a wide range of workloads.
- Remember them as "Balanced" or "Versatile."
- T2: Suitable for low to moderate workloads, great for small applications, websites, and development environments.
- T3: The next generation of T2 instances, offering improved performance.
- M5: Balanced memory and CPU, ideal for general-purpose applications like web servers, app servers, and small to medium databases.
- M6g: Similar to M5 but powered by AWS Graviton2 processors, offering cost savings and better performance for certain workloads.

2. Compute-Optimized Instances:

- These instances are optimized for compute-bound applications.
- Remember them as "High CPU."
- C5: Ideal for compute-intensive workloads such as high-performance web servers, scientific modeling, batch processing, and machine learning.
- C6g: Similar to C5 but powered by AWS Graviton2 processors.

3. Memory-Optimized Instances:

- These instances are optimized for memory-intensive applications.
- Remember them as "High Memory."
- R5: Suitable for memory-intensive applications like in-memory caches, real-time big data analytics, and high-performance databases.
- R6g: Similar to R5 but powered by AWS Graviton2 processors.

4. Storage-Optimized Instances:

- These instances are optimized for storage-intensive applications.
- Remember them as "High Storage."
- I3: Suitable for high I/O applications such as NoSQL databases, data warehousing, and Elasticsearch.
- D2: Designed for Massively Parallel Processing (MPP) data warehousing, MapReduce, and Hadoop.
- H1: Optimized for dense storage, ideal for distributed file systems and big data workloads.

5. Accelerated Computing Instances:

- These instances are equipped with specialized hardware accelerators.
- Remember them as "Specialized."
- P3: Designed for machine learning, high-performance computing (HPC), and video processing.
- G4: Suitable for graphics-intensive applications such as gaming, video encoding, and graphic design.
- F1: Features FPGAs (Field Programmable Gate Arrays) for custom hardware acceleration

Security Groups:

Security Groups are the fundamental of network security in AWS

- They control how traffic is allowed into or out of our EC2 Instances.

Security groups only contain ALLOW rules

Security groups rules can reference by IP or by security group

Security groups are acting as a “firewall” on EC2 instances

They regulate:

- Access to Ports
- Authorized IP ranges – IPv4 and IPv6
- Control of inbound network (from other to the instance)
- Control of outbound network (from the instance to other)

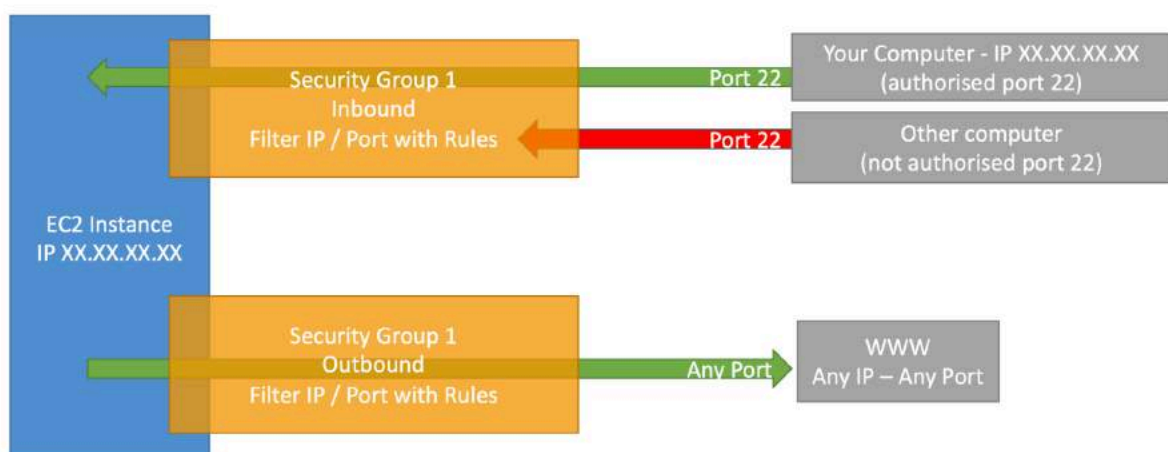
• Can be attached to multiple instances

- Locked down to a region / VPC combination
- Does live “outside” the EC2 – if traffic is blocked the EC2 instance won’t see it •

It’s good to maintain one separate security group for SSH access

- If your application is not accessible (time out), then it’s a security group issue
- If your application gives a “connection refused” error, then it’s an application error or it’s not launched
- All inbound traffic is blocked by default
- All outbound traffic is authorized by default

Security Groups Diagram



Purchasing Options

On-Demand Instances – short workload, predictable pricing, pay by second

- You go to a cloud provider, like Amazon Web Services (AWS), and ask for a computer. With On-Demand instances, it's like renting a regular car. You pay a fixed price per hour or per second, and you get to use the computer for as long as you need it. It's a straightforward, pay-as-you-go model. Just like renting a car for a day, you have the computer resources at your disposal whenever you want them.

- Pay for what you use: • Linux or Windows - billing per second, after the first minute
- All other operating systems - billing per hour
- **Has the highest cost but no upfront payment**
- No long-term commitment
- Recommended for short-term and un-interrupted workloads, where you can't predict how the application will behave

EC2 On-Demand instances allow you to pay for compute capacity on an hourly basis, with no long-term commitments or upfront payments. This provides full control over when to launch, terminate, stop or reboot instances.

Some key things to note about On-Demand instances:

You only pay for the hours that your instances are running, with a 60-second minimum.

Pricing is fixed per instance type and region, as listed on the AWS pricing page.

Recommended for short-term, irregular workloads that cannot be interrupted.

For long-term or steady-state usage, other options like Reserved Instances or Savings Plans provide better cost savings.

Full control over instance configuration, launch/management without restrictions of traditional hosting.

Pay only for actual usage without fixed upfront costs or time commitments of traditional hosting plans

Reserved (1 & 3 years) • Reserved Instances – long workloads

Up to 72% discount compared to On-demand

You reserve a specific instance attributes (Instance Type, Region, Tenancy, OS)

Reservation Period – 1 year (+discount) or 3 years (+++discount)

Payment Options – No Upfront (+), Partial Upfront (++), All Upfront (+++)

Reserved Instance's Scope – Regional or Zonal (reserve capacity in an AZ)

[Recommended for steady-state usage applications \(think database\)](#)

You can buy and sell in the Reserved Instance Marketplace

Comparing it

Regular Daily Rentals (On-Demand Instances):

- Every day, you go to a car rental agency and rent a car for that day. You pay the regular daily rate, and you have a car whenever you need it. This is like using On-Demand instances in the cloud—you get computing power instantly, and you pay for it each time you use it.

Monthly Subscription (Reserved Instances):

- Instead of renting a car every day, you decide to subscribe to a monthly plan with a car rental agency. By committing to renting a car for the entire month, the rental agency gives you a significant discount compared to the regular daily rate. This way, you save money because you've made a longer-term commitment.

Auction Bidding (Spot Instances):

- Alternatively, you find out about an auction where you can bid for a car rental. You bid a certain price, and if your bid is accepted, you get to rent a car at a lower cost. However, there's a chance someone else might bid higher and take the car away from you.

Now, let's relate this to EC2 Reserved Instances:

- **On-Demand Instances (Regular Daily Rentals):** These are like renting computing power on the go. You pay each time you use it, with no upfront commitment.
- **Reserved Instances (Monthly Subscription):** With Reserved Instances, you commit to using a specific type of computing power (an instance type) for a one- or three-year term. In return for this commitment, AWS gives you a significant discount compared to On-Demand prices. It's a way to save money by planning for a more extended period.

So, in essence, EC2 Reserved Instances are like subscribing to a monthly plan for computing power in the cloud. You commit to using it for a longer term, and in return, you enjoy cost savings compared to the pay-as-you-go On-Demand pricing.

Convertible Reserved Instance

- Can change the EC2 instance type, instance family, OS, scope and tenancy • Up to 66% Discount

Regular Reserved Instances (Monthly Subscription):

Regular Reserved Instances:

- **Analogy:** Imagine you subscribe to a monthly car rental plan where you commit to renting a specific type of car for the entire month, and you get a discounted rate.
- **In Cloud Computing:** This is like committing to using a certain type of computing instance (Reserved Instance) for a one- or three-year term, with a significant discount compared to On-Demand prices.

Convertible Reserved Instances (Upgrade or Change Your Car):

Convertible Reserved Instances:

- **Analogy:** Now, let's say you subscribe to a monthly car rental plan, but you have the option to change your car type halfway through your subscription without losing the discount. For example, you start with a sedan but can switch to an SUV if your needs change.
- **In Cloud Computing:** Convertible Reserved Instances allow you to change the instance type within the same family (e.g., from a general-purpose instance to a

memory-optimized instance) without losing the Reserved Instance discount. It provides flexibility to adapt to changing requirements.

Comparison:

- **Regular Reserved Instances:**
 - Like a fixed-term subscription for a specific car type.
 - Provides a significant discount, but the instance type is fixed for the term.
- **Convertible Reserved Instances:**
 - Like a subscription plan where you can switch to a different type of car within the same rental service without losing the discount.
 - Offers flexibility to adapt to changing needs without losing the Reserved Instance benefits.

Key Points:

- **Flexibility with Convertible Reserved Instances:**
 - You can switch to a different instance type within the same family if your computing requirements change.
- **Benefits of Regular Reserved Instances:**
 - You still get the cost savings of a Reserved Instance but with added flexibility to adjust to evolving workloads.

In summary, Convertible Reserved Instances are like a subscription plan where you can change the type of computing instance within the same category without losing the benefits of the discount. It offers a balance between cost savings and the ability to adapt to changing requirements.

Savings Plans (1 & 3 years) –commitment to an amount of usage, long workload

No Commitment (Pay-as-You-Go, On-Demand Instances):

No Commitment (Pay-as-You-Go):

- Analogy: Imagine going to a gym without a membership. You pay for each visit separately. Some days you may go a lot, and on other days, you may skip.

- In Cloud Computing: This is similar to using On-Demand instances in the cloud. You pay for the computing resources you use without any upfront commitment.

1-Year Savings Plan (Gym Membership for a Year):

1-Year Savings Plan:

- Analogy: Now, think about getting a gym membership for a year. You commit to the gym for a year, and in return, you get a discounted rate compared to paying for each visit separately. It's a good deal if you plan to go regularly.
- In Cloud Computing: A 1-Year Savings Plan is like committing to using a certain amount of computing power for one year, and you get a significant discount compared to On-Demand prices.

3-Year Savings Plan (Long-Term Gym Membership):

3-Year Savings Plan:

- Analogy: Consider signing up for a three-year gym membership. It's a more extended commitment, but you get an even more substantial discount compared to the one-year plan. This is ideal if you know you'll be using the gym for the long term.
- In Cloud Computing: A 3-Year Savings Plan is similar to committing to using a specific amount of computing power for three years, and you get the maximum discount compared to On-Demand prices.

Key Points:

- Flexibility: Just like with gym memberships, the longer the commitment, the more you save per unit (visit or compute usage). However, it also means a more extended commitment.
- Upfront Payment: Savings Plans require an upfront commitment, meaning you commit to a certain amount of usage, and in return, you get a lower rate. This can result in significant cost savings compared to Pay-as-You-Go pricing.
- Adaptability: Savings Plans provide flexibility. If your usage goes beyond the committed amount, you still get the Savings Plan rate for those additional resources.

So, in cloud computing terms, Savings Plans offer a way to commit to a certain amount of computing resources for a specific period (1 or 3 years) in exchange for a significant discount compared to paying for resources on a pay-as-you-go basis. It's a way to get cost savings with a longer-term commitment.

An EC2 Savings Plan is a pricing model that offers significant savings on EC2 instance usage in exchange for a commitment to a consistent amount of usage over a term of 1 or 3 years. There are two types of Savings Plans:

Compute Savings Plans - Provide the most flexibility by automatically applying discounts to EC2, Lambda, and Fargate usage regardless of instance type, size, region, OS, etc. You can change workloads freely between services and continue getting discounts. Savings of up to 66%.

EC2 Instance Savings Plans - Offer the lowest prices by applying discounts specifically to usage of a particular instance family in a region. Savings of up to 72%. You have flexibility to change instance sizes and types within the committed family and region.

Savings Plans remove the limitations of Reserved Instances by allowing workload changes without impacting discounts, and committing to a usage amount instead of specific instances. For more details on Savings Plans like pricing, signup process, and how discounts are applied, refer to the AWS documentation or Savings Plans product page.

Get a discount based on long-term usage (up to 72% - same as RIs)

Commit to a certain type of usage (\$10/hour for 1 or 3 years)

Usage beyond EC2 Savings Plans is billed at the On-Demand price

Locked to a specific instance family & AWS region (e.g., M5 in us-east-1)

Flexible across:

- Instance Size (e.g., m5.xlarge, m5.2xlarge)

- OS (e.g., Linux, Windows)
- Tenancy (Host, Dedicated, Default)

The main differences between EC2 Reserved Instances and EC2 Savings Plans are:

Reserved Instances require you to commit to specific EC2 instance attributes like instance type, platform, region etc. for 1 or 3 years. Savings Plans do not require such commitments and provide savings automatically across different instances, regions, AZs etc.

With Reserved Instances, you must perform instance exchanges to change instance types which can impact discounts. Savings Plans provide savings flexibly without any limitations on changing workloads.

Reserved Instances offer capacity reservations in addition to pricing discounts. Savings Plans only provide pricing discounts and do not reserve capacity.

Savings Plans provide savings of up to 66% for Compute Savings Plans and up to 72% for EC2 Instance Savings Plans, similar to discounts offered by Reserved Instances.

Savings Plans automatically apply discounts on EC2 instances as well as services like Fargate and Lambda usage, providing more flexibility than Reserved Instances.

Spot Instances – short workloads, cheap, can lose instances (less reliable)

An EC2 Spot Instance is a type of Amazon EC2 instance that allows you to request unused Amazon EC2 computing capacity for up to a 90% discount compared to On-Demand instances. Here are the key aspects of EC2 Spot Instances:

You can bid on spare Amazon EC2 computing capacity and launch Spot Instances when your bid exceeds the current Spot Price.

Spot Instances are available as long as your maximum bid exceeds the current Spot Price. If the Spot Price increases above your bid, your Spot Instance will be interrupted with a two-minute warning.

You only pay for the Spot Instances that are running, giving you flexibility to optimize your costs for non-critical, fault-tolerant workloads like big data, container workloads, CI/CD, HPC, etc.

Spot Instances are available in all AWS regions and for most EC2 instance types.

You can use Spot Fleets to automatically launch and maintain Spot Instances across different instance types, availability zones to meet your target capacity. This provides cost savings and replaces interrupted instances.

Can get a discount of up to 90% compared to On-demand

Instances that you can “lose” at any point of time if your max price is less than the current spot price

The MOST cost-efficient instances in AWS

Useful for workloads that are resilient to failure

- Batch jobs

- Data analysis

- Image processing

- Any distributed workloads

- Workloads with a flexible start and end time

- Not suitable for critical jobs or databases

Dedicated Hosts – book an entire physical server, control instance placement

A physical server with EC2 instance capacity fully dedicated to your use

- Allows you address compliance requirements and use your existing server- bound software licenses (per-socket, per-core, pe—VM software licenses)
- Purchasing Options:
 - On-demand – pay per second for active Dedicated Host
 - Reserved - 1 or 3 years (No Upfront, Partial Upfront, All Upfront)
 - The most expensive option
- Useful for software that have complicated licensing model (BYOL – Bring Your Own License) • Or for companies that have strong regulatory or compliance needs

If you need visibility into the physical cores and underlying network socket details for an EC2 instance because your database technology vendor bills based on these metrics, you would typically want to use **EC2 Dedicated Hosts**.

Here's why:

- **EC2 Dedicated Hosts:**
 - When you use EC2 Dedicated Hosts, you have complete control over the physical server on which your instances are running. This means you can see and manage the physical cores and sockets directly.
 - Dedicated Hosts provide visibility at the physical host level, allowing you to understand the underlying hardware specifications.
 - This option is suitable when your licensing model requires you to account for specific physical characteristics of the underlying infrastructure, such as the number of physical cores and sockets.

When you launch instances on EC2 Dedicated Hosts, you essentially have the entire physical server dedicated to your account, providing visibility and control at a granular level. This can be beneficial when dealing with licensing models that are tied to the physical characteristics of the underlying hardware.

Please note that while Dedicated Hosts offer this level of visibility, they come with a different pricing model compared to On-Demand or Reserved Instances. It's essential to carefully consider your specific requirements and licensing agreements to determine the most cost-effective and compliance-friendly option for your database deployment.

Dedicated Instances – no other customers will share your hardware

Instances run on hardware that's dedicated to you • May share hardware with other instances in same account • No control over instance placement (can move hardware after Stop / Start)

Dedicated Host:

- A Dedicated Host is a physical server that is fully dedicated for your use and is not shared with other AWS accounts.
- You have full control and ownership of the server and can use your existing licenses for software like Windows Server and SQL Server on the host.
- You can choose server configurations that support a single instance type or multiple types within the same family. The number of instances depends on the configuration.

Dedicated Instance:

- A Dedicated Instance runs within a VPC on hardware that is fully dedicated for your AWS account, but the hardware may be shared with other instances from your account.
- Dedicated Instances provide hardware isolation at the host level from other accounts, but may share hardware with non-dedicated instances from your account.
- You don't have full control of the server like with a Dedicated Host, but the instance is isolated at the host level from other accounts.

Dedicated Hosts:

Dedicated Hosts:

- **Analogy:** Imagine you have your own office building. In this building, you have specific rooms designated for your team's use only. No other team from another company can enter these rooms, and you have full control over the entire space.
- **In Cloud Computing:** A Dedicated Host is like having your own physical server in the cloud. The server is exclusively dedicated to your use, and you have complete control over the entire host. No other customers share the same physical server.

Key Points:

- **Isolation:** Complete isolation from other users. Your resources are dedicated to your workloads.
- **Control:** You have more control over the underlying physical server, including its configuration and security.

Dedicated Instances:

Dedicated Instances:

- **Analogy:** Now, think of your office building again. This time, you have specific rooms designated for your team's use, but there are other teams in the same building. Each team has its own set of rooms, and while you share the building infrastructure, you don't share the specific rooms with other teams.
- **In Cloud Computing:** A Dedicated Instance is like having virtual servers in the cloud where the underlying physical server is shared with others, but the virtual instances you run are dedicated to your account. It provides isolation at the instance level.

Key Points:

- **Instance-Level Isolation:** Your virtual instances run on a shared physical host, but they are isolated from instances belonging to other AWS accounts.
- **Flexibility:** Allows you to run multiple instances with different configurations on the same Dedicated Host.

Summary:

- **Dedicated Hosts:** Imagine having your own office building with exclusive rooms for your team only. Complete isolation and control over the entire physical server.
- **Dedicated Instances:** Picture sharing an office building with other teams, but each team has its own designated rooms. Instances are isolated at the virtual level, providing a balance between shared infrastructure and dedicated resources.

In cloud computing terms, Dedicated Hosts offer maximum isolation and control, while Dedicated Instances provide dedicated virtual resources within a shared physical environment. The choice between them depends on the level of isolation and control your workloads require

Capacity Reservations – reserve capacity in a specific AZ for any duration

Reserve On-Demand instances capacity in a specific AZ for any duration •

You always have access to EC2 capacity when you need it •

No time commitment (create/cancel anytime), no billing discounts

Combine with Regional Reserved Instances and Savings Plans to benefit from billing discounts

You're charged at On-Demand rate whether you run instances or not

Suitable for short-term, uninterrupted workloads that needs to be in a specific AZ

SPOT FLEET

A Spot Fleet allows you to automatically request and manage multiple Spot instances that provide the lowest price per unit of capacity for your application. You specify the instance types your application can use. You define a target capacity based on your needs such as number of instances, CPUs, memory etc.

Spot Fleet launches instances across different Spot capacity pools to meet the target capacity. When instances are interrupted, it automatically replaces them to maintain capacity.

There are a few allocation strategies for distributing instances across pools:

1. Capacity optimized - Analyzes pool metrics to provision from most available pools. Best for workloads like big data and HPC.
2. Lowest price - Provisions from pools providing lowest price per unit at time of request.
3. Diversified - Provisions across multiple pools for availability and cost reduction over time.

By distributing instances across diverse pools, Spot Fleet reduces costs while maintaining capacity even as pool availability changes.

You can check the EC2 documentation for more details on Spot Fleet configuration and strategies.

Public vs Private IP

- Public IP: • Public IP means the machine can be identified on the internet (WWW) • Must be unique across the whole web (not two machines can have the same public IP). • Can be geo-located easily

Private IP means the machine can only be identified on a private network only • The IP must be unique across the private network • BUT two different private networks (two companies) can have the same IPs. • Machines connect to WWW using a NAT + internet gateway (a proxy) • Only a specified range of IPs can be used as private IP

Elastic IPs • When you stop and then start an EC2 instance, it can change its public IP. • If you need to have a fixed public IP for your instance, you need an Elastic IP • An Elastic IP is a public IPv4 IP you own as long as you don't delete it • You can attach it to one instance at a time

- With an Elastic IP address, you can mask the failure of an instance or software by rapidly remapping the address to another instance in your account. • You can only have 5 Elastic IP in your account (you can ask AWS to increase that). • Overall, try to avoid using Elastic IP: • They often reflect poor architectural decisions • Instead, use a random public IP and register a DNS name to it • Or, as we'll see later, use a Load Balancer and don't use a public IP

Private vs Public IP (IPv4) In AWS EC2 – Hands On • By default, your EC2 machine comes with: • A private IP for the internal AWS Network • A public IP, for the WWW. • When we are doing SSH into our EC2 machines: • We can't use a private IP, because we are not in the same network • We can only use the public IP. • If your machine is stopped and then started, the public IP can change

EC2 Placement Groups

EC2 Placement Groups: In Amazon EC2, a placement group is like a special area where you can put your virtual servers (EC2 instances) **to optimize their performance or meet specific requirements**. Think of it as a designated space in the cloud where your instances are placed.

Placement Strategies: There are different strategies for placing instances within a placement group. Each strategy has its own advantages and use cases. Let's look at a couple of them:

Cluster Placement Group:

- *What it is:* In a cluster placement group, instances are placed very close to each other within a single Availability Zone.
- *Why it matters:* This is useful when your applications need extremely low-latency and high-throughput communication between instances, for example, in applications that require fast data exchange.
- *Use case:* • Big Data job that needs to complete fast • Application that needs extremely low latency and high network throughput

Spread Placement Group:

- *What it is:* Instances in a spread placement group are spread across distinct hardware to reduce the risk of simultaneous failures.
- *Why it matters:* It's a good choice if you want to build highly available and fault-tolerant applications. Instances are less likely to be affected by hardware failures or maintenance events because they are on different underlying infrastructure.
- *Application that needs to maximize high availability* • Critical Applications where each instance must be isolated from failure from each other

Partition Placement Group:

- *What it is:* In a partition placement group, instances are spread across logical partitions, each with its own set of hardware.
- *Why it matters:* This is beneficial for distributed and big data applications that can benefit from having instances in separate partitions for parallel processing. Each partition operates independently, making it suitable for applications that can be divided into independent tasks.
- *Use cases:* HDFS, HBase, Cassandra, Kafka

In Layman's Terms: Continuing with our city analogy, think of a partition placement group like dividing your city into districts. Each district has its own set of houses and facilities, and they operate somewhat independently. If one district has a big event, it doesn't necessarily impact the others. This is useful for applications that work on big tasks that can be split into smaller, independent pieces.

ENI?

In basic terms, an Elastic Network Interface (ENI) in AWS is like a virtual network card for your virtual servers (EC2 instances). It helps your instances connect to the internet and other resources.

Example: Web Server with Two ENIs: Imagine you have a web server in the cloud, and you want it to have two "network cards." One card (ENI) connects to the internet to serve web pages, while the other card connects to a private network for internal communication with databases.

Use Cases:

Network Segmentation: You can use ENIs to separate different types of traffic. For instance, one ENI for public-facing traffic and another for internal communication.

Security Groups: Each ENI can have its own security group, allowing you to control access separately. This is useful for enforcing different security rules for different types of traffic.

Versatility: ENIs provide flexibility by allowing you to attach and detach them from instances. You can adapt your network setup based on your application's needs.

Isolation: With multiple ENIs, you can isolate different types of traffic, enhancing security by controlling communication between instances.

IP Address Limitations: Each ENI requires a private IP address, and there are limits to the number of private IPs a single instance can have. This may be a consideration in scenarios with a large number of ENIs.

Complexity: Managing multiple ENIs introduces complexity. It requires understanding how each network interface is configured and how traffic flows through them.

Think of an ENI like a computer with multiple network plugs. Each plug serves a different purpose—maybe one is for talking to the outside world, and another is for talking to your team inside the office. ENIs let you customize how your virtual servers connect, making your cloud infrastructure more flexible and secure. So, in summary, ENIs are like having multiple network cards for your virtual servers, allowing you to control and customize their connections in the cloud. They're handy for setting up different types of communication and enhancing the security of your applications.

EC2 Hibernate

EC2 hibernation is a feature that allows you to pause (or hibernate) your Amazon EC2 instances instead of terminating them. When an instance is hibernated, its current state, including the contents of its RAM, is saved to the Amazon Elastic Block Store (EBS) volumes attached to the instance.

Example: Blogging Website with EC2 Hibernation: Imagine you have a blogging website running on an EC2 instance. During periods of low activity, instead of shutting down the

instance (which would lose all in-memory data), you hibernate it. This means the current state of the website, including unsaved drafts or session data, is saved. When you resume the instance, it picks up right where it left off.

Use Case:

Cost Savings: Hibernating instances during periods of low demand can save costs compared to running instances continuously.

Preserving State: For applications that have in-memory data or ongoing processes, hibernation allows you to pause the instance and resume later without losing any state

Advantages:

Fast Startup: When you resume a hibernated instance, it starts up faster than a regular start because it doesn't need to go through a full boot process; it simply reloads the saved state from the EBS volumes.

Data Persistence: Hibernation preserves the current state of the instance, including in-memory data, which is crucial for applications that need to maintain state between sessions.

Disadvantages:

EBS Costs: While hibernated, the instance's state is stored on EBS volumes, and you may incur additional costs for the storage.

Not Suitable for All Workloads: Hibernation is most beneficial for workloads that can benefit from pausing and resuming, and not all workloads are suitable for this. Stateless or short-lived instances might not see significant advantages.

In Layman's Terms:

Think of EC2 hibernation like putting your computer to sleep instead of turning it off. When you hibernate an EC2 instance, it's like taking a snapshot of your computer's current state and saving it to a special place. Later, when you wake it up, it opens to exactly where you left off, without starting from scratch. This helps save money and preserves your applications' current state, making it a handy feature for certain types of workload

Supported Instance Families – C3, C4, C5, I3, M3, M4, R3, R4, T2, T3, ... • Instance RAM Size – must be less than 150 GB. • Instance Size – not supported for bare metal

instances. • AMI – Amazon Linux 2, Linux AMI, Ubuntu, RHEL, CentOS & Windows... • Root Volume – must be EBS, encrypted, not instance store, and large • Available for On-Demand, Reserved and Spot Instances

EBS Volume

- An EBS (Elastic Block Store) Volume is a network drive you can attach to your instances while they run • It allows your instances to persist data, even after their termination • They can only be mounted to one instance at a time (at the CCP level) • They are bound to a specific availability zone

- Analogy: Think of them as a “network USB stick” • Free tier: 30 GB of free EBS storage of type General Purpose (SSD) or Magnetic per month

- It's a network drive (i.e. not a physical drive) • It uses the network to communicate the instance, which means there might be a bit of latency • It can be detached from an EC2 instance and attached to another one quickly • It's locked to an Availability Zone (AZ) • An EBS Volume in us-east-1a cannot be attached to us-east-1b • To move a volume across, you first need to snapshot it • Have a provisioned capacity (size in GBs, and IOPS) • You get billed for all the provisioned capacity • You can increase the capacity of the drive over time

EBS Snapshots

Make a backup (snapshot) of your EBS volume at a point in time • Not necessary to detach volume to do snapshot, but recommended • Can copy snapshots across AZ or Region

EBS Snapshot Archive • Move a Snapshot to an “archive tier” that is 75% cheaper • Takes within 24 to 72 hours for restoring the archive

- Recycle Bin for EBS Snapshots • Setup rules to retain deleted snapshots so you can recover them after an accidental deletion • Specify retention (from 1 day to 1 year) • Fast

Snapshot Restore (FSR) • Force full initialization of snapshot to have no latency on the first use (\$\$\$)

AMI Overview

AMI Overview • AMI = Amazon Machine Image

AMI are a customization of an EC2 instance

- You add your own software, configuration, operating system, monitoring...
- Faster boot / configuration time because all your software is pre-packaged
- AMI are built for a specific region (and can be copied across regions)
- You can launch EC2 instances from:
 - A Public AMI: AWS provided
 - Your own AMI: you make and maintain them yourself
 - An AWS Marketplace AMI: an AMI someone else made (and potentially sells)

AMI Process

- 1) Start an EC2 instance and customize it
- 2) Stop the instance (for data integrity)
- 3) Build an AMI – this will also create EBS snapshots
- 4) Launch instances from other AMIs

EC2 Instance Store

EBS volumes are network drives with good but “limited” performance

- **If you need a high-performance hardware disk, use EC2 Instance Store**
- Better I/O performance
- EC2 Instance Store lose their storage if they're stopped (ephemeral)
- Good for buffer / cache / scratch data / temporary content • Risk of data loss if hardware fails • Backups and Replication are your responsibility

EBS Volume Types

EBS Volume Types • EBS Volumes come in 6 types

- gp2 / gp3 (SSD): General purpose SSD volume that balances price and performance for a wide variety of workloads
- io1 / io2 (SSD): Highest-performance SSD volume for mission-critical low-latency or high-throughput workloads
- st1 (HDD): Low cost HDD volume designed for frequently accessed, throughput-intensive workloads
- sc1 (HDD): Lowest cost HDD volume designed for less frequently accessed workloads

EBS Volumes are characterized in Size | Throughput | IOPS (I/O Ops Per Sec) • When in doubt always consult the AWS documentation – it's good!

- **Only gp2/gp3 and io1/io2 can be used as boot volumes**

EFS V/S EBS V/S Instance Store

USE CASE: Amazon EFS:

Scenario: You have a web application with multiple EC2 instances that need to share files and data, and you want a scalable and shared storage solution.

Use Case:

Multiple Instances: If your web application is hosted on multiple EC2 instances and they need to share the same files (like code, images, or configuration files), Amazon EFS is a good choice. EFS allows you to mount the same file system across multiple instances.

Scalability: EFS scales automatically as your storage needs grow, making it suitable for applications with dynamic or growing data requirements.

USE CASE: Amazon EBS:

Scenario: You have a web application that requires high-performance, block-level storage, and you want data persistence even if an instance is stopped.

Use Case:

Database Storage: If your web application relies on a database and requires consistent, high-performance storage, you might use Amazon EBS volumes. EBS is often a good choice for databases that need reliable and persistent block storage.

Data Persistence: If you need data to persist even when an instance is stopped or terminated, EBS volumes can be detached from one instance and attached to another, providing data continuity.

USE CASE:INSTANCE STORE

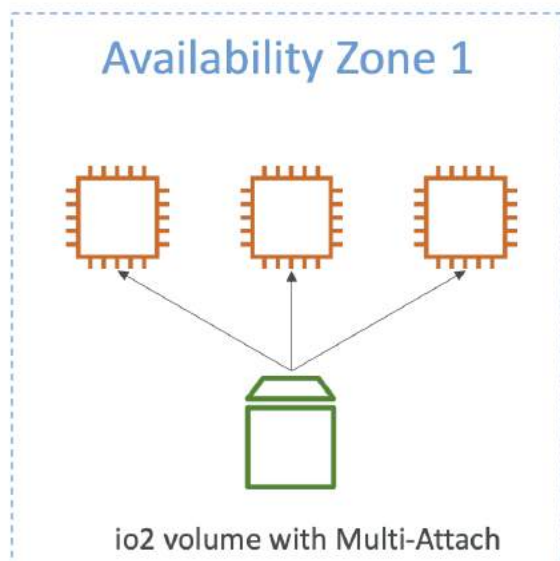
Scenario: You have a stateless application or temporary data processing tasks where data persistence is not crucial, and performance is a top priority.

Use Case:

Temporary Workloads: For temporary storage needs, such as processing large datasets or running batch jobs, instance store (ephemeral storage) could be suitable. Keep in mind that data on instance store is lost if the instance is stopped or terminated.

High-Performance Needs: If your application requires extremely high I/O performance, instance store, being physically attached to the instance, can provide better performance compared to network-attached storage like EBS or EFS.

EBS Multi-Attach – io1/io2 family



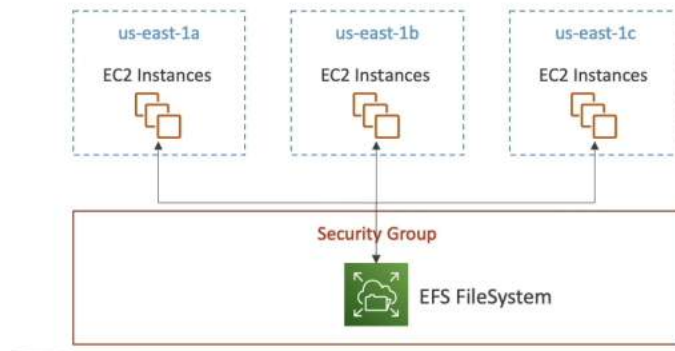
Attach the same EBS volume to multiple EC2 instances in the same AZ
Each instance has full read & write permissions to the high-performance volume
Use case:

- (1) Achieve higher application availability in clustered Linux applications (ex: Teradata)
- (2) Applications must manage concurrent write operations
- (3) Up to 16 EC2 Instances at a time
- (4) Must use a file system that's cluster-aware (not XFS, EXT4, etc...)

Exam Question -EFS

When to use EFS ? What options to set for EFS

available, scalable, expensive (3x gp2), pay per use



It is managed file system

- (1) Connecting EC2(s) in different AZs at the same time
 - (2) Highly available
 - (3) Very Scalable
 - (4) Pay per use
 - (5) Costly
 - (6) CMS, Data sharing , wordpress, Web serving
 - (7) For access setup security group
 - (8) NFS.V4.1 protocol
 - (9) Only for Linux based AMI'(s) and not windows.
 - (10) Auto scaling
 - (11) Can enable Encryption at rest
 - (12) Setting Performance Mode
 - General Purpose (Default) ,latency sensitivity use cases
 - Max I/O (*Higher Throughput and Higher Latency BIG Data and Media processing needs*)
 - Throughput modes*
-
- Burst Mode
 - Provisioned (you can have 1gb for 1tb storage)
 - Elastic (auto scale based on workload for unpredictable workload)
- (13) Storage Tiers - move to a diff tier after a few days
 - (1) Standard (for freq access)
 - (2) EFS(IA) - Lower price for in freq access and cost to retrieve and you must chose lifecycle policy
 - (14) Can be Single Zone (Dev) or Multi AZ (For Prod)

Quiz

Which of the following EBS volume types can be used as boot volumes when you create EC2 instances?

`gp2, gp3, st1, sc1`

`gp2, gp3, io1, io2`

`io1, io2, st1, sc1`

Scalability

Vertical Scalability

- Its when you want to use more computer power in your resources
- So you increase the size of your instances
- RDE ,ElastiCache are some examples
- Common for Non distributed Systems
- From t2nano to I2

Horizontal Scalability

- Its when you want to increase the number of instances
- This is more for Distributed Systems
- E.G(Auto scaling group, Load balancer)

High Availability

- Generally goes hand in hand with horizontal scaling
- Implies running your app in at least 2 data centers (AZs)
- Goal is to avoid a data center loss
- Auto scaling group multi AZ
- Load balancer multi AZ

Load Balancer

Why use Load Balancer

- Spread load across multiple instances
- Expose single point of DNS entry
- Seamlessly handle failure of downstream instances
- Perform regular health checks
- Provide SSL termination for your websites

Application Load Balancer

- X-Forwarded-For - gets the IP address of the client
- X-Forwarded-Proto - gets protocol and X-Forwarded-Port gets the port

Network Load Balancer

- Operates at Layer 4
- Handles millions of requests

- Less latency ~100ms
- Has one static IP per AZ and supports assigning Elastic IP
- information alert
- Network Load Balancer has one static IP address per AZ and you can attach an Elastic IP address to it. Application Load Balancers and Classic Load Balancers have a static DNS name.
- Extreme performance using TCP/UDP
- Target Groups (EC2, Private IP addresses ,ALB)
- Health checks TCP, HTTP, HTTPS

Gateway Load Balancer

- Deploy ,scale and manage fleet of 3rd party n/w virtual appliances
- Example have all traffic in your system go through a firewall or a IDS , Deep Packet Inspection,
- Operates at Layer 3 (IP Packets)
- Combines (1) Transparent n/w gateway (2) Distributes traffic to virtual appliances
- Uses the GENEVE protocol 6081
- TARGET (EC2, IP address and must be private)

Sticky Session (SESSION AFFINITY)

- Multiple request from the same client always go to the same EC2 instance from Load Balancer
- Works with Classic, Application and Network Load Balancer
- Cookie is sent from the client that is used for this
- Use Case : User is connected to the same backend instance
- Brings imbalance to the load
- Two types of cookies (1) Application based cookies (2) Duration based cookies
- Cookie name just be specified for each target group
- Cannot use AWSALB,AWSALBAPP,AWSALBTG

CROSS ZONE BALANCING

- Cross-Zone Load Balancing refers to the distribution of incoming traffic evenly across instances in multiple Availability Zones within the same region.
- In AWS Elastic Load Balancing (ELB), when Cross-Zone Load Balancing is enabled, the load balancer distributes traffic evenly across all registered instances in all enabled Availability Zones.
- This helps in achieving better fault tolerance and high availability by spreading the load across multiple zones, ensuring that if one zone fails, the application can continue serving traffic from instances in the healthy zone
-

SSL/TLS

- Traffic will be encrypted between client and server
- SSL = Secure Socket layer
- TLS = Transport Layer Security
- Public SSL certificates are issued by CA
- SSL certs have expiration date that must be set
- ALB uses x.509 certificate
- ACM used to manage certificate
- Clients used Server name Indicate SNI to specify hostname
- SNI solves the problem of loading multiple certs in one server
- Client will share hostname in initial handshake
- Server then returns the correct cert
- This works ALB and NLB
- Connection draining is the time to complete in flight requests while the instance is de-registering or unhealthy

Auto Scaling Group

Why use ASG

- You can scale out no of instances to match a high load
- You can scale in by removing instances to match a lower load
- We can have a min and max of EC2 instances running
- ASG is free and we pay for EC2
- We can scale ASG based on CloudWatch Alarms
- Metrics like Avg CPU or custom metrics are used

Dynamic Scaling Policies

1. Target Tracking Scaling : (You need to avg ASG CPU to be around 40%)
2. Simple Step Scaling :(When CW alarm is triggered (CPU > 70%) , then add 2 units
3. Scheduled Actions: (Anticipate a scaling based on known usage patterns.

Important Concepts [RDS] :

Introduction

1. Stands for relational database Service
2. It's a managed service.
3. E.g PostgreSQL, MySQL, MariaDB, Oracle, Microsoft SQL, Aurora
4. Scales Automatically
5. Useful for apps with unpredictable workloads

Why RDS

1. Automatic Provision of Database and OS patching
2. Continuous backups and point in time restore.
3. Dashboards to view performance
4. Read replicas
5. Multi AZ for Disaster Recovery
6. Maintenance window for upgrades
7. Scaling Capability

RDS Auto Scaling

1. When RDS detects your running out of space ,it scales automatically.
2. Useful for apps with unpredictable workloads
3. Supports PostgreSQL, MySQL, MariaDB, Oracle, Microsoft SQL

READ REPLICAS

1. They help to scale your reads.
2. You can have up to 15 Read Replicas
3. You can do this with AZ, Cross AZ and Cross Region
4. They are eventually consistent
5. ASYNC replication
6. RDS replicas within same region have no cost
7. RDS replicas across regions have cost
8. One example is where you want to run some reporting analytics on your database. Now instead of running the reporting analytics on the main DB, you create a read replica out of the main DB and the reporting analytics happens on this read replica , thereby not affecting the prod db.
- 9.

RDS MULTI AZ(DISASTER RECOVERY)

1. SYNC Replication
2. When app writes to master DB , its written to the other DB as well
3. There is one DNS name
4. There is a failover when there is a failure or loss of AZ.
5. Not for scaling
6. No manual intervention
7. You can setup READ REPLICAS AS MULTI AZ

RDS SINGLE AZ TO MULTI AZ

1. This is a Zero downtime operation.
2. Click modify on the DB.
3. And enable Multi AZ
4. Internally a snapshot is taken and kept in a standby db
5. New DB is restored from the standby DB

RDS CUSTOM

1. For Oracle and Microsoft SQL server
2. We have full admin access to OS and underlying database
3. We can access underlying ec2 instance using SSH
4. We can deactivate automation mode to perform customization

AMAZON AURORA

1. Postgres and MySQL are both supported by Aurora
2. Aurora storage automatically grows in increments of 10 GB til 128 TB
3. Can have up to 15 read replicas
4. Replication process is faster than MySQL

5. Failover is instantaneous
6. 20% more costly than RDS
7. Stores 6 copies of your data across 3 AZs
8. 4 out of 6 needed for writes
9. 3 out of 6 needed for reads
10. If there is bad data it does peer-peer relocations
11. One instance takes writes (Master)
12. 15 read replicas and any of these can be master
13. Supports cross region replication
14. Reader endpoints helps with connection load balance
15. Reader endpoints automatically connects to all read replicas
16. Any time a client connects to Reader endpoints, it will get connected to one of the Read Replicas.
17. Aurora Global Database allows 5 secondary read only regions and up to 16 read replicas.
18. Takes < 1 second for Aurora Global for cross region replication.
19. ML integration with AWS Sagemaker , AWS Comprehend.
20. RDS Backups
 - Every 5 mins , transaction logs.
 - 1-35 days of retention

Creating a snapshot is better and more cost effective if you plan on using infrequently as storage costs more
21. Aurora Backups
 - 1- 35 days and cannot be disabled
22. RDS & Aurora Restore options

You can create a new database from a snapshot

You can restore MySQL RDS from S3

- (1) backup your on prem
- (2) Store in S3
- (3) Restore the file in a new RDS instance running MySQL

You can restore MySQL Aurora from S3

- (1) backup using Percona Xtrabackup
- (2) Store in S3
- (3) Restore the file in a new Aurora cluster running MySQL

23. Aurora Cloning is creating a new cluster from an existing one.

Faster than snapshot and create

Uses copy on write protocol

Uses when creating staging db from prod db without affecting prod db.

24. RDS & Aurora Security

At rest encryption using AWS KMS

To encrypt an encrypted one ,create a snapshot and restore it as unencrypted.

For inflight encryption ,AWS TLS root certificate

IAM roles to connect

n/w access to DB can be controlled using Sec Groups

Except for RDS Custom , SSH is not available.

Audit logs can be enabled and sent to Cloud Watch(for long retention)

25. RDS Proxy

Fully manage DB proxy

Apps can pool and share DB connections

Helps reduce stress on DB and increase RAM

Supports MySQL, MSSQL, MariaDB, PostgreSQL, Aurora

Enforce IAM auth and store creds in AWS Secret manager

Only access via VPC

You can not create encrypted Read Replicas from an unencrypted RDS DB instance.

You need to store long-term backups for your Aurora database for disaster recovery and audit purposes. What do you recommend?

AMAZON ELASTICACHE

Overview

1. Way to manage cache such as redis or memecache
2. Helps reduce on database
3. REDIS is more for high durability, availability and failover scenarios
4. MEMECACHD is more for sharing and multithreaded scenarios

Security

(REDIS)

1. Supports IAM for REDIS
2. Supports SSL in flight encry

(memecached)

1. Supports SASL based auth

Patterns

Lazy loading : All read data is cached and data can become stale

Write Through : Writes or updates data to cache when we update DB

Session Store : Stores temporary data in cache using TTL feature

Redis Use Case

(1) Sorted Sets : ELEMENT ORDERING AND UNIQUENESS

(2) Each time an element is added , its rank is computed and added in correct order

(3)

Important Concepts [ROUTE 53] :

Introduction

Amazon Route 53 is a highly scalable and an authoritative(You can update DNS records) DNS. Route 53 is a domain registrar. 53 is a reference to traditional DNS port.

Records

Each Record contains Domain/ Subdomain

Record Type *example.com*

A value *123.12.23.56*

Routing policy *How Route 53 responds*

TTL *cache time*

Route 53 supports A/AAAA/CNAME/NS

It also supports CAA/DS/MX/NAPTR/PTR/SOA/TXT/SPF/SRV

A maps to IP4 address

AAAA maps to ipv6 address

CNAME

maps hostname to another hostname

Target is a hostname that is either A or AAAA

You can't create CNAME for top record like example.com but can for www.example.com

NS (name servers) control how traffic is routed for the domain

Hosted Zones

A container for records that tell how to route to domain/sub domain

Public hosted zone : Manage routing on the internet

Private hosted zone : Manage routing on the VPC

You pay 0.5 per month for the hosted zone.

TTL

High TTL : 24 hrs and possibly outdated records

Low TTL : 60 secs , more traffic ,

Except for Alias records, TTL is mandatory for each DNS record

CNAME vs ALIAS

CNAME maps a hostname to another hostname (works only for non root domain)

ALIAS maps a hostname to a AWS resource (works for both root and non root)

You can;t set TTL for ALIAS record

Targets for ALIAS are (EC2,CLOUD FRONT,Elastic Beanstalk, API Gateway,S3,VPC endpoint,Global Accelerator)

You cannot set a ALIAS record for EC2 DNS name

ROUTING policies

How Route 53 responds to queries?

Supports following policies

- (1) Simple
- (2) Weighted
- (3) Failover
- (4) Latency based
- (5) Geo Location
- (6) Multi Value Answer
- (7) Geo Proximity

Simple Routing Policies

Typically routes traffic to a single resource

You can specify multiple values in the same record

When multiple values are returned, a random one is chosen by the client.

When ALIAS is enabled , specify only 1 AWS resource.

Cannot be associated with Health checks.

Weighted Routing Policies

Control % of requests that goes to each resource

Each record as a relative weight in $\% = \text{weight of record} / \text{total weight of all records}$

Assign a weight of 0 to a records to stop sending traffic

If all have 0 , then all are returned equally,.

Latency Routing Policies

Redirect to a resource based on least latency
Very helpful when latency is priority for users
Latency is based on traffic between users and AWS region
Germany users may be redirected to US

ROUTE 53 - Health checks

HTTP Health checks for public users
They are used for automated DNS failover
They can monitor an endpoint
They can monitor other health checks
They can monitor CW alarms
They can be integrated with CW metrics

ROUTE 53 Monitor an endpoint

~15 global health checkers will check an endpoint
Healthy/ Unhealthy threshold is 3
30 sec interval for checking
When > 18% report the endpoint is healthy
The Health checks pass for status codes of 2xx and 3xx
Pass /fails based on the text in the first 5120 bytes of response
Configure router.firewall to allow incoming requests

ROUTE 53 -Calculated Health checks

Combine results of multiple health checks to a single health check
You can use OR AND NOT
Can monitor up to 256 health checks
Specify how many children need to pass to make parent pass
Usage : Maintenance without all HEALTH CHECKS failing

ROUTE 53 - HEALTH CHECKS PRIVATE HOSTED ZONE

Route 53 health checker are outside VPC

They can't access private endpoints

You can create CW metric and associate CW to it

ROUTE 53 - GEOLOCATION

Routing based on user location

Specified by continent , country, state

Use case : website localization,content distribution,load balancing

ROUTE 53 - GEOPROXIMITY

Routing based on geo location of users and resources

Ability to shift resources based on bias

ROUTE 53 - IP BASED ROUTING

Routing based on IP ADDRESS

You provide a list of CIDRS for clients with locations

Optimize performance, reduce network costs

ROUTE 53 - multi value

Route 53, Multi-Value Routing Policies enable you to associate multiple values with a single DNS name, and the system randomly responds with one of those values.

Let's break it down with a layman's example:

Imagine you have a website, and you want to make sure it's always available and responsive to your users. You have multiple servers located in different regions or with different capabilities to handle the incoming traffic. This is where Multi-Value Routing comes into play.

Example:

Server in the US: You have a web server hosted in the United States.

Server in Europe: You also have another server hosted in Europe.

Now, you set up a Multi-Value Routing Policy for your website's domain name using Route 53. You associate both the US server's IP address and the Europe server's IP address with your domain. When users try to access your website, Route 53 randomly provides one of these IP addresses in the DNS response.

Use Cases:

- **Load Balancing:** If your website experiences heavy traffic, you can distribute the load among different servers by associating their IP addresses using Multi-Value Routing. This helps prevent a single server from getting overwhelmed.
- **Geographic Redundancy:** If you want your website to be resilient and available even if one server or region goes down, Multi-Value Routing allows you to route traffic to another server or region.
- **A/B Testing:** You can use Multi-Value Routing to conduct A/B testing by associating different server IP addresses with different versions of your website. This way, users are randomly directed to different versions, helping you analyze performance and user preferences.

In essence, Route 53 Multi-Value Routing Policies provide a simple yet effective way to enhance the availability, performance, and reliability of your website by randomly distributing traffic among multiple resources.

Route traffic to multiple resources

Routing value returns multiple value/resources

Up to 8 healthy records for each Multi Value query

Important Concepts [SOLUTION ARCH DISCUSSION]: whatistime.com :

Vertical Scaling:

T2 micro changes M5 instance because of more users

Horizontal Scaling:

Multiple M5 instances but users need to know of the elastic IP address

ROUTE 53

Have a A record that has multiple ip address of the M5 instances

Load Balancer:

We have private instances in an AZ and have an ELB in front of the M5

ELB is public facing. Users will query whatistime.com and there is an ALIAS record for the ELB and users will hit the ELB

ASG:

Scaling and Scaling out based on traffic

MULTI AZ:

The ELB is now launched in multiple AZ and Auto scaling in multiple AZ

ELB Health checks:

Enabling Health checks to have requests sent to the servers that are healthy

Important Concepts [SOL ARCH MyClothes.com] :

Initial Phase:

An arch that has Route 53 for MyClothes.com DNS resolving , an ELB inside a Multi AZ
And auto scaling group to scale in and scale out based on capacity.

Requirement:(user adds item to cart but data lost in sec request)

- Sticky Session is enabled to mitigate this
- Cookies are sent as a part of the request informing ELB of the prev session
- Introduce an Elastic Cache and Session ID is used as cache key to save session information
- Another alternative is DynamoDB to save session data.

Requirement:(save user data)

- Add Amazon RDS to save user data

Requirement:(scale reads as users are generally reading)

- Add RDS read replicas most

Requirement:(survive disasters)

- Add multi Az feature for RDR and ElastiCache

Requirement:(security groups)

- (1) Restrict traffic to Ec2 , sec group from Load Balancer
- (2) Restrict traffic to Elastic Cache, Sec group from EC2
- (3) Restrict traffic to RDS . Sec group from EC2

Important Concepts [SOL ARCH MyWordpress.com] :

Requirement:(display picture uploads)

- Initial Arch has Route 53 , Multi AZ ELB , M5 EC2s in an Mutli AZ Auto scaling group and an RDS Mutli AZ.

Requirement:(Scaling)

- Using Aurora for scaling Read replicas

Requirement:(storing images)

- Using EBS for storing images in EBS volumes

Requirement:(Resolve error when images are stored in diff EBS volumes)

- Use EFS using ENI
- All EC2s will now have a shared data space

INSTANTIATION APPLICATION QUICKLY :

- EC2 can use Golden AMI
- Bootstrap using User Data
- Hybrid Mix of Golden AMI + User Data
- RDS can be restored from a snapshot will have schemas ready
- EBS can be restored from a snapshot

ELASTIC BEANSTALK:

- Most web apps have a common arch (ALB + ASG_
- Beanstalk is a dev center view of deploying an app on AWS
- Managed Service that auto handles capacity provision , lb, scaling
- Beanstalk is free but you pay underlying components
- Collection of Beanstalk components
- App version
- Environment (has triers and can create multiple environments)
- Create -> upload > launch-> manage
- Web server env for a web app

- Worker env for long running workloads or tasks on aschedule

AMAZON S3:

- (1) Buckets are defined at the region level
- (2) 3-63 characters long name.
- (3) Must start with lowercase or number
- (4) No uppercase
- (5) No underscore
- (6) Must not start with prefix xn-
- (7) MAX OBJECT SIZE = 5TB = 5000GB
- (8) If uploading more than 5GB use multi part upload
- (9) JSON based policy
 - (1) Resources : Buckets and objects
 - (2) Effect : Allow/ Deny
 - (3) Action : Set of API to Allow or Deny
 - (4) Principal: The account or user to apply the policy to
- (10) MNEMONIC FOR S3 BUCKET POLICY
 - VSEPAR
 - V - Version
 - S - STATEMENT
 - E- Effect
 - P - Principal
 - A - Action
 - R - Resources
- (11) S3 Bucket versioning is enabled at the bucket level
- (12) When you enable Replication , only new objects will be replicated
- (13) To allow existing objects to be replicated , you can enable S3 Batch Replication.
- (14) S3 Storage Classes
 - (1) GENERAL PURPOSE
 - (2) STANDARD INFREQUENT ACCESS
 - (3) ONE ZONE INFREQUENT ACCESS
 - (4) GLACIER INSTANT RETRIEVAL
 - (5) GLACIER FLEXIBLE RETRIEVAL
 - (6) GLACIER DEEP ARCHIVE
 - (7) S3 INTELLIGENT TIERING

You can have storage move between classes or use S3 Lifecycle configurations

(15) High Durability : 99.9999999999

(16) High Availability 99.99%

S3 General Purpose:

- Use for Freq accessed Data
- Need for low latency
- Sustain 2 conc failures
- Use cases : Big data analytics ,mob and gaming apps

S3 INFREQUENT ACCESS:

- For data that is less freq accessed
- Lower cost than S3 standard
- 99.9% availability
- Use case : DR backups

S3 ONE ZONE INFREQUENT ACCESS:

- High durability
- 99.5% availability
- Storing secondary backups

AMAZON S3 GLACIER STORAGE CLASSES

- Low cost for archiving and backup
- Price for storage and obj retrieval

AMAZON S3 GLACIER INSTANT RETRIEVAL

- Millisecond retrieval and great for data uses once a quarter
- Minimum storage duration of 90 days

AMAZON S3 GLACIER FLEXIBLE RETRIEVAL

- Retrieval Types: Expedited(1 To 5 mins) , Standard : 3 to 5 hrs , Bulk : 5 to 12 hours
- Minimum storage duration of 90 days

AMAZON S3 DEEP ARCHIVE

- Retrieval Types: Standard :12hrs , Bulk : 48 hours
- Minimum storage duration of 180 days

AMAZON INTELLIGENT TIERING

1. Small monthly monitoring fee
2. Moves objs automatically between tiers based on usage
3. No retrieval charges

FREQUENT ACCESS TIER : DEFAULT

INFREQUENT ACCESS TIER : Object not accessed for 30 days

ARCHIVE INSTANT ACCESS TIER: Object not accessed for 90 days

Archive access tier : Configurable from 90 days to 700 +days

Deep Archive Access : Configurable from 180o days to 700+ days

AMAZON MOVING CLASSES:

AMAZON AUTOMATED MOVING

1. Moving infrequent accessed objects to standard IA
2. Moving archive to Glacier or Glacier Deep Dive
3. Lifecycle rules can be used to move them automatically

LIFE CYCLE RULES

1. Transition :
This allows objects to transition to another storage class.
Move obj to standard IA **60 days after creation**
Move to Glacier after **6 months**
2. Expiration:
Configure objects to expire **60 days to expire**
Access log files can be set to delete after **365 days**
Can be used to delete older version of files
Can be used to delete incomplete multi part uploads
3. Rule can be created for certain prefix ,object tags

S3 Analytics:

1. Helps you to decide when to transition objects to the right storage class
2. Provides recommendations for standard and Standard IA
3. Not for One Zone IA or Glacier
4. 24-48 hrs to start seeing data analysis
- 5.

S3 REQUESTER:

1. with requester the requester pays for S3 requests and downloads
2. Helpful when you want to share large data sets with other accounts

S3 BASELINE PERFORMANCE:

1. Receive 3500 PUT/COPY/POST/DELETE
2. Receive 5500 GET/HEAD REQUESTS
3. No Limit on Prefixes
4. Spreading read across 4 prefixes , you achieve 22,000 req on GET and HEAD

S3 PERFORMANCE:

1. Multi part upload is recommend for file > 100 MB
2. Must have for files > 5GB
3. Can help parallelize uploads

S3 TRANSFER ACCELERATION:

1. You can increase speed here by transferring the file to an Edge location
2. The edge location then help transfer the file to the target region of the S3 bucket
3. Compatible with multi part

S3 BYTE RANGE FETCH:

1. Parellize get by specifying byte range
2. Can be used to retrieve only partial data

S3 SELECT AND GLACIER SELECT

1. Retrieve less data using SQL using server side filtering.
2. Filter by rows and Columns
3. Less CPU cost client side

S3 Batch Operation

1. You can perform bulk operation on existing S3 objects using a single request
2. This can be modifying, copy object props, encrypt objects,
3. A job contains a list of objects, the action to perform
4. S3 inventory to get object list and S3 select to filter

Important Concepts [S3 Security] :

Objects in S3 can be encrypted in 4 methods

1. SERVER SIDE ENCRYPTION with S3 Managed Keys (default)
2. SERVER SIDE ENCRYPTION with KMS keys stored in AWS KMS
3. SERVER SIDE ENCRYPTION with Customer Provider keys (SSE-C)
4. Client Side Encryption

SERVER SIDE ENCRYPTION SSE-S3

1. AES 256 encryption
2. You must set header `"x-amz-server-side-encryption":"AES:256"`
3. Enabled by default for new buckets and objects

KMS

1. Encryption managed by KMS by AWS
2. User control of keys and audit key usage by Cloud Trail
3. Object is encrypted server side
4. You must set header `"x-amz-server-side-encryption": "aws:kms"`

Limitation :

- Affected by KMS limits
- When you upload it calls KMS API (GenerateDataKey)
- When you download it calls Decrypt KMS API
- SERVICE QUOTA INCREASE (you can request quota increase)

SEC

1. Using Server side enc by keys managed by Customer outside of AWS
2. S3 won't store the enc key you provide
3. HTTPS must be used
4. Enc key must be provided by S3 in every request

Client Side Encryption

1. Client side libraries used for encryption
2. Client will encrypt data before sending to S3
3. Clients will decrypt data before downloading from S3
4. Customers fully manage keys and encryption cycle

ENCRYPTION IN TRANSIT (TTL/SSL)

1. Encryption in flight is called TTL/SSL
2. S3 exposes 2 endpoints (http/https)
3. Https mandatory for SSE-C

Forcing encryption in Transit

4. Bucket policy aws:
Condition:{
 Bool:{
 aws:secureTransport:"false"
 }
}

MFA

1. Required to perm delete obj version
2. Suspend versioning on bucket
3. To use MFA ,enable versioning in bucket

S3 PRE SIGNED URLs

1. Time of 1 mins to 720mins
2. `--expires-in` param

S3 GLACIER VAULT

3. A write once read many model
4. Create a vault lock policy
5. Lock policy for future edits
6. Helpful for compliance and data retention

S3 Object Vault

7. A write once read many model
8. This is for each object and not the whole s3 bucket
9. Block object version deletion for some time
10. Compliance :
 - (1) Object version can't be deleted by users and root users
 - (2) retention periods can't be shortened
11. Governance
 - (1) most users (except admin users) cannot delete objects or overwrite
 - (2) some user can change retention or delete object
12. Retention Period: Protect the obj for a fixed period
13. Legal Hold : Protect it indefinitely
Someone with s3:PutObjectLegalHold permission

S3 Access Points

14. Simplify sec management

S3 Object Lambda

15. Change obj before its retrieved
16. Use case : redacting , file conversion, resizing watermark

Amazon Cloudfront] :

Cloudfront:

1. Cache data at edge locations and improves read performance
2. 216 Edge location

Amazon Storage extra :

AWS Snow Family:

1. Secure and portable devices at the edge to process data in/out of AWS
2. Data Migration : Snowcone, Snowball edge, Snowmobile
3. Edge computing : Snow Cone, Snowball edge

If it takes more than a week to transfer then use Snowball devices

Snowball edge

1. Physical data transport solution
2. Block storage and AWS S3 compatible storage
3. Use cases include : Large data cloud migration , DC Decommission, disaster recovery

Snowcone and Snowcone SSD

1. Small portable computing
- 2.

[AWS Messaging] :

SQS:

1. Default Retention period of 4 days
2. Max retention of 14 days
3. Message limit of 256kb
4. Can have duplicate and out of ordering messages
5. SQS Consumer receives about 10 messages at a time

SNS - (1) SQS (2) Lambda (3) Kinesis firehose

[AWS Containers] :

ECS:

Launching ECS Tasks on ECS clusters

ECS:(Fargate Launch Type)

We do not provision infrastructure

ECS:(IAM roles)

EC2 Instance Profile

is used by the ECS Agent
Sends container logs to Cloudwatch Logs
Reference Sensitivity Data in secrets manager or SSM parameter store

ECS Task Role

Allows each task to have a specific role
Use different roles for different ECS services you run

ECS Load Balance Integration

ECS -Data Volumes

Mount EFS onto ECS tasks to share data
Works with EC2 and Fargate
Target running in any AZ will share the data

Use case : Persistent multi AZ storage for your containers
*S3 cannot be mounted as a file system.

ECS Auto scaling

Scaling based on
(1) ECS Service Avg CPU Utilization
(2) ECS Service Avg Memory Utilization
(3) ALB Req count per Target
(4) Scale based on target value set in Cloud Watch Metric

Amazon ECR

Storing docker images in AWS

Fully integrated in ECS and stored in S3

Access controlled in IAM

Amazon EKS

Managing Kubernetes in AWS

Alternative to ECS

Why ? As Kubes is Op Source

Kube is cloud agnostic

Node types

(1) managed node groups supports on demand or spot

(2) self managed (created by you) or use prebuilt ones by AMI and supports on demand or spot

(3) AWS Fargate where no maintenance is required

Data Volume

Need to specify storage class on your EKS cluster

Needs Compliant Storage Interface Driver

Supports for EBS, EFS, FSx for Lustre, FSx for NetApp

Amazon App Runner

Fully managed service to deploy web applications

Automatically build and deploy the app

[AWS Serverless] :

Meaning: (Devs don't manage servers)

Lambda limits per region

Execution limits:

128MB-10GB (1MB Increments)

15mins is max execution time

Env variables : 4kb

Disc capacity : 512MB TO 10GB

Concurrency execution: 100

Deployment:

Deployment size: 50mb

Size of uncompressed deployment : 250mb

Env variable : 4kb

Lambda Snapstart

Improves Lambda performance by 10x for java 11 or above

Customization at the edge

Edge fns run close to users to reduce latency

Lambda with RDX proxy

Improve scalability by pooling and sharing connections

Improve sec by enforcing IAM

Lambda needs to be executed in your VPC , because RDS proxy is never public

DynamoDb

A NoSQL DB

Single digit millisecond performance

Standard and IA class

Max item size is 400kb

Supports scalar types as :String, number, boolean,null

Documents : List,map

Evolving Schema

Modes

(1) provisioned mode (you specify read/write) for predictable workloads

(2) on demand - pay for what you use (unpredictable workloads)

DynamoDb Accesslarator(DAX)

For caching reads

Microseconds latency

Sits in front of dynamodb

DynamoDb Stream Processing

Ordered stream of item level modifications in table

React to changes in DD in real time

Real time usage analytics

Cross Region replication

(1) DB streams 24 hrs limit

(2) Kineses Streams (1 year of retention ,

DynamoDb Global team

Low latency access in multiple regions

Enable Dynambo DB streams is a must

TTL

Expire items after a specific time

Backups

Optional for 35 days

Can also have on demand backups

Integration S3

Export to s3 and analysis with athena

ETL transformation

Format:JSON or ION format

API gateway

Create API

WORKS WITH LAMBDA

WORKS WITH HTTP

WORKS WITH AWS SERVICE

API endpoint types

(1) Edge optimized : requests routed through CloudFront to improve latency

(2) Regional: for clients within same region

(3) Private : within your VPC

API SECURITY

User Authentication through

- IAM Roles (useful for internal applications)
- Cognito (identity for external users – example mobile users)
- Custom Authorizer (your own logic)

Custom Domain Name HTTPS security through integration with AWS Certificate Manager (ACM)

if using Edge-Optimized endpoint, then the certificate must be in us-east-1

If using Regional endpoint, the certificate must be in the API Gateway region

Must setup CNAME or A-alias record in Route 53

API Cognito

- 1) Cognito User Pools provide sign in for apps
- 2) Cognito Identity pools provides temp AWS credentials

- S3: Object Storage
- S3Glacier:ObjectArchival
- EBS volumes: Network storage for one EC2 instance at a time
- Instance Storage: Physical storage for your EC2 instance (high IOPS)
- EFS: Network File System for Linux instances, POSIX filesystem
- FSx for Windows: Network File System for Windows servers
- FSx for Lustre: High Performance Computing Linux file system
- FSx for NetApp ONTAP: High OS Compatibility
- FSx for OpenZFS: Managed ZFS file system
- Storage Gateway: S3 & FSx File Gateway, Volume Gateway (cache & stored), Tape Gateway • Transfer Family: FTP, FTPS, SFTP interface on top of Amazon S3 or Amazon EFS
- DataSync: Scheduled data sync from on-premise to AWS, or AWS to AWS
- Snowcone / Snowball / Snowmobile: to move large amount of data to the cloud, physically • Database: for specific workloads, usually with indexing and querying

Important Concepts [CIDR] :

A Dedicated Direct Connect connection supports 1Gbps and 10Gbps.

Default VPC

1. All new AWS accounts have a default VPC
2. New EC2 instances are launched in the default VPC if no subnet is specified
3. Has internet Access
4. All EC2 instances have public IPV4 addresses
- 5.

CIDR:

Two components

- (1) Base IP (XX.XX.XX.XX)
- (2) Subnet mask that defines how many bits can change in the IP

Classless Inter-Domain Routing (CIDR) blocks are for specifying a range to IP addresses in format of IPv4 or IPv6. For the sake of simplicity I will explain rest of this in format of IPv4 however it is applicable to IPv6. General format for CIDR Blocks: **x.y.z.t/p** .x, y, z and t are numbers from 0 to 255. Basically, each represents an 8 bit binary number. That's why it is range is up to 255. Combination of this numbers becomes an IPv4 IP address that must be unique to be able to identify a specific instance. In case of AWS, p is a number from 16 to 28. It represents the number of bits that are inherited from given IP address. For example: 10.0.0.0/16 represents an IP address in following format: 10.0.x.y where x and y are any number from 0 to

255. So, actually it represents a range of IP addresses, starting from 10.0.0.0 to 10.0.255.255. However for each CIDR block, AWS prohibits 5 possible IP addresses. Those are the first 4 available addresses and the last available address. In this case:

Virtual Private Cloud : At a high level, you can think of a VPC in AWS as a logical container that separates resources you create from other customers within the Amazon Cloud. It is you defining a network of your own within Amazon. You can think of a VPC like an apartment where your furniture and items are analogous to databases and instances. The walls of your apartment isolate and protect your things from being accessible to other tenants of the apartment complex.

Subnet : Subnets would then be analogous to the different rooms in your apartment. They are containers within your VPC that segment off a slice of the CIDR block you define in your VPC. Subnets allow you to give different access rules and place resources in different containers where those rules should apply. You wouldn't have a big open window in your bathroom on the shower wall so people can see you naked, much like you wouldn't put a database with secretive information in a public subnet allowing any and all network traffic. You might put that database in a private subnet (i.e. a locked closet).

Internet Gateway

Allows resources to connect to the internet in a VPC. Highly available , scales horizontally. One VPC can have one Internet Gateway

Bastion Host

Allows us to SSH into Private ECE instances through a public subnet (BH)[Inbound]

PUBLIC SUBNET ⇒ BASTION HOST ⇒ PRIVATE EC2 INSTANCE

Bastion Host Security Group must allow inbound from the internet from port 22 from restricted CIDR

Security group of the the EC2 instance must allow the Security group of the Bastion host

NAT Instance(Network Address Translations)

A NAT instance, on the other hand, generally enables hosts(EC2instances) in a private subnet within your VPC, outbound access to the internet. So a bastion host allows inbound access to known IP addresses and authenticated users, a NAT instance allows instances within your VPC to go out to the internet.

NAT Gateway

AWS Managed , higher bandwidth,higher availability, no administration.Pay per hour for usage and bandwidth.

Created in a specific AZ uses an Elastic IP

CANNOT BE USED BY EC2 INSTANCE in the same subnet.

Requires an IGW and BW 5GBPS to 100 gbps

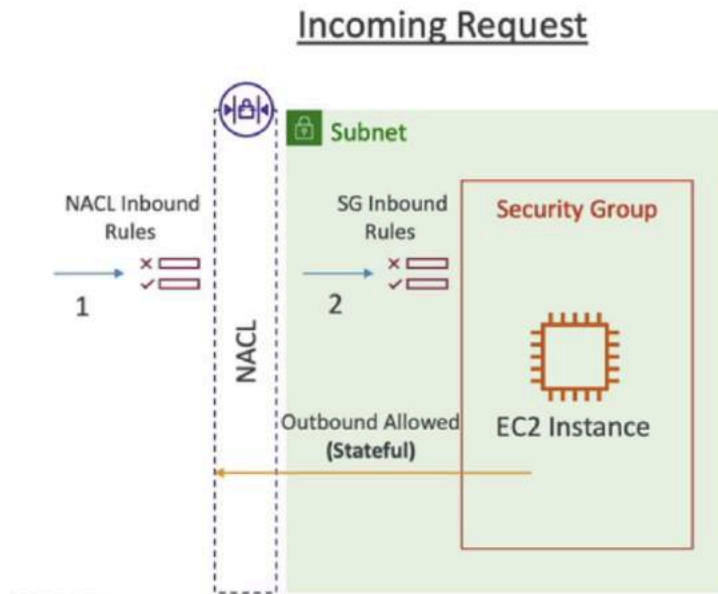
NACL

Network Access Control List (NACL)



- NACL are like a firewall which control traffic from and to subnets
- One NACL per subnet, new subnets are assigned the Default NACL
- You define NACL Rules:
 - Rules have a number (1-32766), higher precedence with a lower number
 - First rule match will drive the decision
 - Example: if you define #100 ALLOW 10.0.0.10/32 and #200 DENY 10.0.0.10/32, the address will be allowed because 100 has a higher precedence over 200
 - The last rule is an asterisk (*) and denies a request in case of no rule match
 - AWS recommends adding rules by increment of 100
- Newly created NACLs will deny everything
- NACL are a great way of blocking a specific IP address at the subnet level

Security Groups & NACLs



NACL are like firewall which control traffic from and to subnets.

Default NACLs allow everything in and everything out.

Clients connect to a server on a defined port and expects a response on an **Ephemeral Ports**

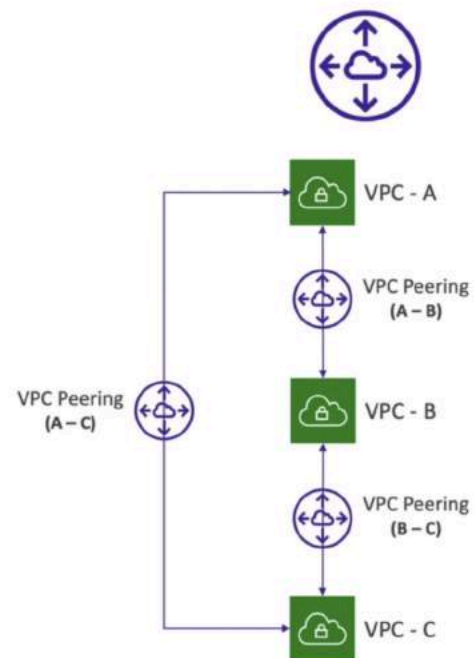
Security Group	NACL
Operates at the instance level	Operates at the subnet level
Supports allow rules only	Supports allow rules and deny rules
Stateful: return traffic is automatically allowed, regardless of any rules	Stateless: return traffic must be explicitly allowed by rules (think of ephemeral ports)
All rules are evaluated before deciding whether to allow traffic	Rules are evaluated in order (lowest to highest) when deciding whether to allow traffic, first match wins
Applies to an EC2 instance when specified by someone	Automatically applies to all EC2 instances in the subnet that it's associated with

NACL Examples: <https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html>

VPC Peering

VPC Peering

- Privately connect two VPCs using AWS' network
- Make them behave as if they were in the same network
- Must not have overlapping CIDRs
- VPC Peering connection is **NOT** transitive (must be established for each VPC that need to communicate with one another)
- You must update route tables in each VPC's subnets to ensure EC2 instances can communicate with each other



VPC Endpoints

Your instances don't have to go through the internet to access AWS resources. Use a VPC endpoint deployed within a VPC . Thereby you use a Private n/w to connect to AWS services

Two types

- 1) Interfaces Endpoints - Uses an ENI | cost per hour + GB processed
- 2) Gateway Endpoints - Uses a Gateway and uses a target in the Route Table . Two targets (S3 and DynamoDB)

When to use which ?

For SA-CO3 gateway is preferred . As you need to access the route table only and no additional cost. [Interface is preferred for On Premise or when you connect from another VPC](#)

VPC Flow Logs

- Capture information of IP traffic going into your interfaces.
- VPC, Subnet, ENI
- Monitor and Troubleshoot connectivity issues.
- SRCADDR and DSTADDR helps identify problematic ips
- SRCPORT & DSTPORT helps identify problematic ports
-

Site to Site VPN

- Use Case is to connect AWS to Corporate Data Center
- VPN Gateway at AWS side and Corporate Gateway at CDC side
- This creates a link between n/w of VPC with n/w of CDC
- CG could either have a public ip address or have a private IP address in which case it is behind a NAT . This case you use the public ip address of the NAT. In addition you need to Route Propagation in route table. If you need to ping EC2 instance , from on premise ,you need to enable ICMP on the inbound of the security group.
- AWS VPN CloudHub : Now you have multiple CGateways(s) and you require secure communication b/w them. Low cost hub and spoke model for pri & sec b/w diff location.

DX or Direct Connect

- Provides a p/w connection from remote n/w to your VPC
- Private virtual interface is needed for connected your vpc to AWS
- Direct connect gateway to one or more VPCs
- use case : increased bandwidth and throughput ,large data sets lower costs
- Consistent network experience
- If you want to set up a Direct Connect to one or more VPC in different regions you must use a Direct Connect Gateway.
- Dedicated Connections : Has physical ethernet ports dedicated to a customer. Request made to AWS first then completed by AWS Direct Connect Partners

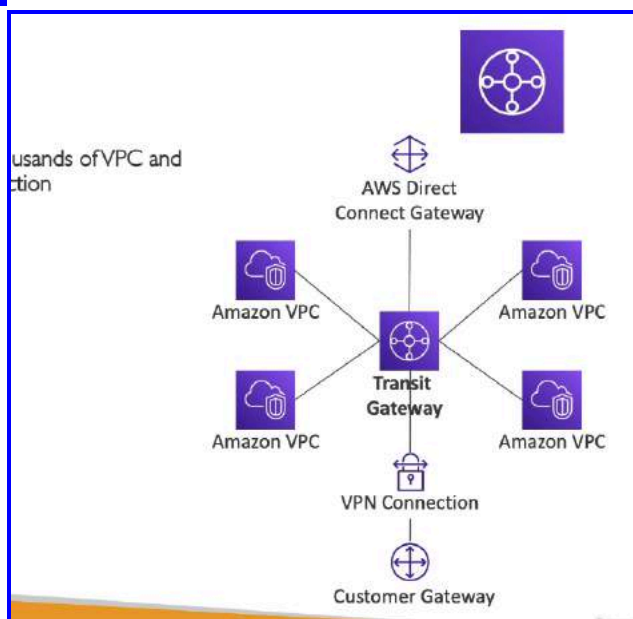
- Hosted Connections are made via AWS Direct Connect Partners and capacity can be added or removed on demand
- 1,2,5,10 GBPS are made available
- **Lead times are often longer than a month to establish a new connection**
- Data in transit is not encrypted but is private.
- High Resilient for Critical Workloads where we have one connection at multiple locations
- Max resilience is achieved by separate connections terminating on separate devices in more than one location.

Site to Site connection as a backup

Incase DC fails your can setup a backup DC or a site to site VPN connection

Transit Gateway

- For having transitive peering between VPC(s)
- Route tables decide which VPC talk to which for n/w security
- Only service that supports **IP Multicast**
- Another use case is increasing bandwidth(using ECMP) equal cost multi path routing
- Its a routing strategy to allow to forward a packer over multiple best paths
- Share Direct Connect between multiple accounts
-



VPC Traffic mirroring

- Allows you to capture and inspect network traffic
- Route traffic to sec devices you manage
- Filter packets
- Source and target can be in same or diff VPC
- Content inspection, threat monitoring , troubleshooting

IPV6

- 3.4×10^8 address
- 0000 - ffff
- IPV4 Cannot be disabled for subnets and vpc
- So operating in dual mode - priv ip4 and public ipv6
- If you cannot launch an ec2 instance you have most likely exhausted all ip address so you need to create a new IP4V4 CIDR

EGRESS ONLY INTERNET GATEWAY

- Used for IPV6 Only.
- EC2 Instance in the Private subnet can access the internet using the gateway over IPV6 , but the internet cannot access the instance . Allows only outbound connection from the private subnet.

Region :

AWS Global Infrastructure is divided into

- 1) Regions (us-east-1)
- 2) Availability Zones
- 3) Local Zones
- 4) Points of Presence
- 5) Network

A cluster of Data Centers

Services is scoped to a Region

How to choose an AWS region ? **AWS Possible Question**

Factors in choosing a region :

- 1) COMPLIANCE: Govts want it local to a region
- 2) LATENCY: More users in America then makes sense to use America
- 3) AVAILABLE SERVICES : Whether the services are available
- 4) PRICING : As it varies from region to region

AVAILABILITY ZONES :

Each Region has availability zones. Usually 3

Min : 2

Max : 6

Designed to avoid failures as they are separate from each other
AZ(s) connected to each other together are called regions

Points of Presence

400 + POP
400 + Edge Locations
10 + Regional Cache

IAM -INTRODUCTION

Groups only contain users and not others.
Users don't have to belong to a group.
Users can belong to multiple groups
Users or Groups can be assigned JSON documents called policies.

IAM -POLICIES INHERITANCE

GROUP level policy gives access to all users within a group
INLINE level policy is assigned at the user level and is unique

IAM POLICY STRUCTURE

- Version: policy language version , always include "2012-10-17"
 - Id : an identifier for the policy (optional)
 - Statement " one or more individual statements (required)
- Sid : an identifier for the statement
Effect : Allow or Deny
Principal: account/user/role to which this policy is applied to
Action: list of actions this policy allows(get put post)
Resource: List of resources the action applies to
Condition: Conditions for when the policy is in effect

```
{
  Version:"2012-10-17"
  Id:"S3-Account-Permissions",
  Statement:[ {
    "Sid":1
    "Effect":"Allow
    "Principal":{"AWS":["arn:aws:iam::12121212121:root"]},
    "Action":["s3:GetObject","s3:PutObject"],
    "Resource":["arn:aws:s3::mybucket/*"]
```

}

IAM Password policy

MFA -AWS

AWS Recommended

Options

- (1) VIRTUAL MFA DEVICE
- (2) Universal 2nd Factor (U2F) security key
- (3) Hardware Key FOB
- (4) AWS GovCloud

ACCESSING AWS

- (1) AWS Management Console
- (2) AWS CLI
- (3) AWS SDK

Access Keys are generate through AWS console

Using CLI

```
ananthkumbala@Ananth's-MBP ~ % aws iam list-users
{
  "Users": [
    {
      "Path": "/",
      "UserName": "ananth",
      "UserId": "AIDAZCL5I5KAJJGLRHOPW",
      "Arn": "arn:aws:iam::623569857152:user/ananth",
      "CreateDate": "2023-12-16T20:50:44+00:00",
      "PasswordLastUsed": "2023-12-16T20:54:03+00:00"
    }
  ]
}
```

IAM Roles for Services

- (1) Some AWS services will need to perform actions on your behalf.
- (2) To do so we will assign permission to AWS services with IAM roles

IAM

IAM Does not require REGION SELECTION.

Identity Access Global Management.

We use it for creating users .

Users can be grouped together.

For eg - A group for developers , A group for operations.

Not all users need to be in a Group but its a best practice.

A user can belong to multiple groups

User or Groups can be assigned using a **Policy Document**

Inline policy when user does not belong to a group

IAM UserGroups cannot be a part of another UserGroup. They can contain only IAM Users

A statement in an IAM Policy consists of Sid, Effect, Principal, Action, Resource, and Condition. Version is part of the IAM Policy itself, not the statement.

COMMANDS:

- 1) `aws configure // to configure`
- 2) `aws iam list-users // to list users`

Some AWS services would need to perform actions on our behalf
In order for the AWS services to perform actions on our behalf , they need to be
Given permissions

This is done using IAM roles

E.G

EC2 Instance Roles
Lambda Function Roles

CREATE ROLE

SELECT EC2 in the USE CASE

Then specify what EC2 should do

Give IAMReadOnly

DemoRoleForEC2

IAM Security Tools

IAM Credential Reports: User list and Credentials Status

IAM Access Advisor : Service Permissions granted and when they were last accessed

DO NOT USE ROOT ACCOUNT FOR ANYTHING EXCEPT ACCOUNT SETUP

EC2

Bootstrapping

“Bootstrapping means launching commands when the machine starts”

- Install updates , run software etc can be done during bootstrapping

Instance	vCPU	Mem (GiB)	Storage	Network Performance	EBS Bandwidth (Mbps)
t2.micro	1	1	EBS-Only	Low to Moderate	
t2.xlarge	4	16	EBS-Only	Moderate	
c5d.4xlarge	16	32	1 x 400 NVMe SSD	Up to 10 Gbps	4,750
r5.16xlarge	64	512	EBS Only	20 Gbps	13,600
m5.8xlarge	32	128	EBS Only	10 Gbps	6,800

PUBLIC IP changes every time you stop and start
PRIVATE IP for an EC2 does not change

EC2 Instance Types

- 1) General purpose
- 2) Compute Optimized
- 3) Memory Optimized
- 4) Accelerated Computing
- 5) Storage Optimized
- 6) Instance Features
- 7) Measuring Instance Performance

m5.2xLarge

M : INSTANCE CLASS

5 : Generation (AWS Improves them over time)

2xLarge : Size within Instance Class

General Purpose

For **Web Servers** and **Repositories**

- Good Balance between memory, compute and networking

T series

Compute Optimized

For compute intensive tasks that requires high performance processors

Batch processing ,Machine Learning

Dedicated Gaming

C Series

Memory Optimized

Fast performance when processing large data sets in memory

High Performance for Relational/ Relational Database

Distributed Cache Store

Processing Real time Unstructured Data

R series

Storage Optimized Instance

Great for Data Intensive Tasks for Local Storage

High Frequency OLTP systems

Relations/ Non Relational

Redis

Data Warehouse

DFS

M series

Security Groups control how traffic is allowed INTO and OUT of the AWS EC2

- Fundamentals of EC2 security
- They only contain allow rules
- They regular access to ports
- Authorized IP ranges

What is a Security Group?

- Virtual Firewall for your EC2 instances.
- Blocks all traffic except the ports, protocols, and sources you specify.



Be A.
Better

SSH Port 22

FTP : 21

SFTP 22

80 UNSECURED WEB SITES

443 SECURED WEB SITES

3389 RDP

Ssh ec2-user@ip-address-comes-here

Navigate to the folder where the key is stored

Ssh -i <keyfile> ec2-user@ip-address-comes-here

Chmod 0400 <key>

Whomami

EC2 Instances

On Demand : pay for what use

Billed for per-second after the first minute

-High cost

-No upfront payment

- No Long term commitment
- RECOMMENDED FOR SHORT TERM
- Whenever you can't predict how application can behave*

RESERVED INSTANCE

1 Year or 3 year.

More the time bigger the discount
NO UPFRONT or ALL UPFRONT
Discount again for THIS
Reserve a specific instance
You also converted RESERVED INSTANCES
You can change this instance here

SCHEDULED INSTANCE

(for a specific time window) but you want to commit for 1-3 years

SPOT INSTANCES

HIGHEST DISCOUNT IN AWS

YOU CAN LOSE THIS INSTANCE AT ANY TIME , IF THE PRICE YOU ARE WILLING TO PAY FOR THEM IS LESS THAN THE CURRENT SPOT PRICE

Spot price changes over time
Most Cost Efficient

Useful for workloads that are RESILIENT to FAILURE as you can lose your workload

BATCH JOBS
DATA ANALYSIS
IMAGE TRANSFORMATION
DISTRIBUTED WORK LOAD (So even if one fails)
FLEXIBLE START AND END TIME
Never use it for DATABASE OR CRITICAL TASKS

AMAZON DEDICATED HOSTS

HERE YOU RENTING AN ENTIRE SERVER

COMPLIANCE REQUIREMENTS AND REDUCE COSTS BY YOUR EXISTING SERVER
BOUND SOFTWARE LICENSE
3 YEAR RESERVATION
EXPENSIVE
USEFUL for S/W COMPLEX LICENSE MODEL
BYOL
STRONG COMPLIANCE NEEDS
DEDICATED INSTANCES
EC2 INSTANCES THAT ARE RUNNING ON H/Q DEDICATED TO YOU
You may share h/w with other instances
INSTANCE WON'T GIVE YOU ACCESS TO UNDERLYING H.W

SCENARIO

EC2 Instance Spot Instance Request

You define a max price

YOUR MAX PRICE > INSTANCE PRICE

THE INSTANCE PRICE WILL CONSTANTLY VARY

IF IT GOES UP

YOU HAVE 2 MINS GRACE PERIOD TO STOP OR TERMINATE IT.

of Spot Instances and optionally On-Demand Instances that is launched based on criteria that you specify. The Spot Fleet selects the Spot capacity pools that meet your needs and launches Spot Instances to meet the target capacity for the fleet. By default, Spot Fleets are set to *maintain* target capacity by launching replacement instances after Spot Instances in the fleet are terminated. You can submit a Spot Fleet as a one-time *request*, which does not persist after the instances have been terminated. You can include On-Demand Instance requests in a Spot Fleet request.

A Spot Fleet is set of Spot Instances and optionally On-Demand Instances that is launched based on criteria that you specify. The Spot Fleet selects the Spot capacity pools that meet your needs and launches Spot Instances to meet the target capacity for the fleet. By default, Spot Fleets are set to *maintain* target capacity by launching

replacement instances after Spot Instances in the fleet are terminated. You can submit a Spot Fleet as a one-time *request*, which does not persist after the instances have been terminated. You can include On-Demand Instance requests in a Spot Fleet request.

SPOT BLOCK:

IF YOU DONT WANT AWS TO RECLAIM

YOU CAN BLOCK THE INSTANCE FOR A TIME WINDOW

1-6 HOURS

THEY ARE NO LONGER AVAILABLE SINCE JULY 1 ,2021

IP4: 4 numbers with 3 dots

3.7 billion different address in public space

IPV6 : 6 Numbers and AWS supports

PUBLIC IP :

ACCESSIBLE OVER THE INTERNET

UNIQUE OVER THE WEB

MACHINE CAN BE IDENTIFIED OVER THE INTERNET (www)

PRIVATE IP :

ONLY ACCESSIBLE OVER THE NETWORK

ONLY ACCESSIBLE OVER THE N/W

UNIQUE WITHIN THE N/W

ONLY SPECIFIC RANGE CAN BE USED

ELASTIC IP:

IP CHANGES WHEN YOU START AND STOP IT

FOR FIXED IP , YOU NEED AN ELASTIC IP

YOU CAN MASK FAILURE OF AN INSTANCE USING AN ELASTIC IP , BY

RAPIDLY REMAPPING THE ADDRESS TO ANOTHER INSTANCE

YOU CAN HAVE 5 BY DEFAULT (YOU CAN ASK AWS TO INCREASE IT)

CAN BE ASSOCIATED TO AN INSTANCE OR A NETWORK INSTANCE

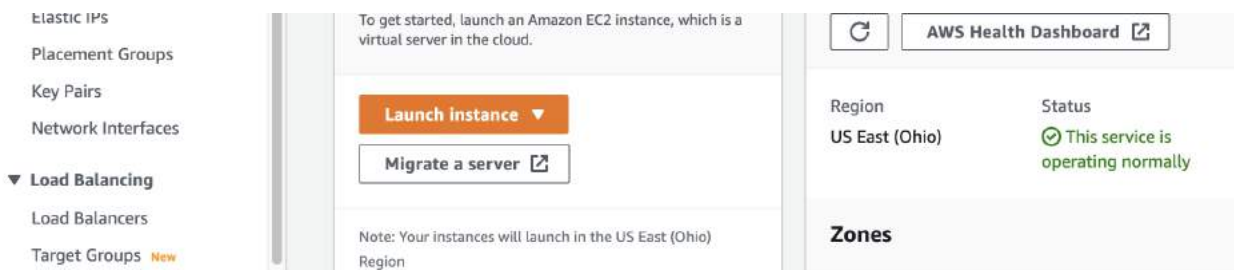
Overall try to avoid

- Poor arch decision
- Use random IP and register DNS name

- USE Load balancer

PLACEMENT GROUPS

- NEED FOR CONTROL ON HOW INSTANCES ARE PLACED IN AWS INFRA
- PLACEMENT GROUPS HELP HERE
- CLUSTER (**High performance , High Risk , Low latency , Same Rack and same AZ**)
 - RISK AS ALL INSTANCES FAIL IF THE RACK Fails
 - Big data job that needs to finish fast
 - App that needs LOW latency
- SPREAD (Critical Application)
 - SPANNED ACROSS DIFFERENT AZ
 - REDUCED RISK
 - LIMIT OF 7 INSTANCES PER AZ
 - USE CASE IS AN APP THAT NEEDS HIGH AVAILABILITY
- PARTITION (SPREAD ACROSS PARTITION BUT NOT ISOLATED)
 - SPANNED ACROSS DIFFERENT AZ
 - EACH PARTITION REPRESENT A RACK
 - SAFE FROM RACK FAILURE
 - UPTO 7 PER AZ
 - PARTITION FAILURE DOES NOT AFFECT OTHER PARTITIONS
 - USE CASE OF DISTRIBUTED DATA , BIG DATA APPLICATIONS, PARTITIONS AWARE , Cassandra , HDFS



ELASTIC NETWORK INTERFACE

LOGIC COMPONENT IN A VPC

REPRESENTS A VIRTUAL NETWORK CARD

EACH ENI

- 1 PRIMARY IP V4 AND 1 OR MORE SECONDARY IPV4
- ONE OR MORE SECURITY GROUPS
- A MAC ADDRESS
- CAN BE MOVED INDEPENDENTLY

EC2 HIBERNATE

DATA ON EBS IS INTACT(STOP)

DATA ON EBS IS REMOVED(TERMINATED)

WHEN EC2 IS STARTED . IT PLAYS THE SCRIPT

OS BOOTS

CACHE IS WARMED UP

IN EC2 HIBERNATE , ALL DATA IN RAM IS PRESERVED

BOOT IS MUCH FASTER

OS IS NOT STOPPED

RAM IS DUMPED INTO EBS VOLUME

ROOT EBS VOLUME IS ENCRYPTED

USE CASE

LONG PROCESS RUNNING

SAVE RAMR STATE

SERVICE TAKES A LONG TIME TO INITIALIZE

SUPPORTS C M and R

INSTANCE RAM : 150 GB or lesser

Not supported for bare metal

AMI INSTANCE

HAS TO EBS VOLUME

AHS TO BE ENCRYPTED AND LARGE

ON DEMAND AND RESERVED

CANNOT BE HIBERNATED FOR MORE THAN 60 DAYS

EC2 NITRO UNDERLYING PLATFORM NEXT GEN EC2
BETTER NETWORKING OPTIONS
HIGH SPEED EBS
BETTER SECURITY

Vcpu

Multiple threads on 1 cpu
Each thread is a virtual cpu
Ec2 comes with ram and vcpu

EBS VOLUMES

EBS Volume is a n/w drive that we can attach to your instance
Can only be attached to one instance
They are bound to a specific AZ
They are like a USB stick
30gb SSD
You need to do snapshot to move
You have to provision capacity in advance
Two EBS volumes to one instance

You can create unattached EBS volumes

"Delete on termination" when you create EBS
DELETED IS CLICKED by default for ROOT volume
Not enabled for the other volumes
Summary of EBS Volumes
N/W DRIVE
LIKE A USB STICK
THEY CAN BE DETACHED FROM AN EC2 AND ATTACHED TO ANOTHER
THEY ARE LOCKED TO AN AZ
YOU HAVE TO PROVISION CAPACITY IN ADVANCE
SAME EBS VOLUME CANNOT BE ATTACHED TO TWO INSTANCES AT THE SAME TIME

EBS Snapshot is like a backup
You can copy SNAPSHOTS across AZ(s)

You are recommended to detach a volume but its not mandatory

You can transfer data to another AZ

Use case :

- stop the EC2 instance ahead of tiem
- create a snapshot of the EBS
- Now use the snapshot in another A-Z
- EBS Snapshot archive - 75% cheaper
- Recycle bin for snapshots for restoring data based on rules

AMI(S) ARE CUSTOMIZATION OF AN EC2 instance

AMAZON MACHINE IMAGE

We can customize based on our requirements

FASTER Boot time

They can be built for a specific region and copied across regions

They are built for a specific region but can be copied

You have vendors in AWS who create AMI and sell it in AWS marketplace

Process:

You start an instance

Customize it

Stop it

Then Build an AMI from it

Launch instance from the AMI

EC2 Instance Store

FOR HIGH PERFORMANCE USE ECE INSTANCE STORE

Like a hardware disk attached

HIGH I/O PERFORMANCE

CAVEAT :

BUT IF YOU STOP => STORAGE WLL B LOST

Good Use Case:

BUFFER /CACHE/ TEMPORARY DATA

BUT NOT FOR LONG TERM

EBS IS GOOD USE CASE

IF THE UNDERLYING SERVER WHERE EC2 INSTANCE IS DEPLOYED FAIL

THEN THIS STORE WILL BE LOST

HIGH PERFORMANCE HARDWARE TO EC2(exam usecase) -----EC2 INSTANCE STORE

EBS Volume Types

6 TYPES

GP2/GP3 -

BALANCE PRICE AND PERFORMANCE COST EFFECTIVE

LOW LATENCY

IO1/IO2 -

HIGH PERFORMANCE --- MISSION CRITICAL ,LOW LATENCY , HIGH THROUGHPUT

ST1 –

LOW COST HD VOLUME , FREQUENTLY ACCESSED THROUGHPUT

LOG PROCESSING BIG DATA

SC1 -

LOW COST , LESS FREQUENTLY

ARCHIVED DATA

ONLY GP2/GP3 AND IP1/IP2 CAN BOOT

GP2/GP3

Y

1GB - 16TB

GP 2 : SMALL (3000 IOPS) , THEY ARE LINKED

GP3 : NEW , 3000 IOPS , CAN INCREASE INDEPENDENTLY

PROVISIONED IOPS

MISSION CRITICAL

USE CASE:

DATABASE WITH HIGH WORKLOADS

EBS MULTI ATTACH -

ATTACH EBS TO Multiple EC2s in the same AZ
(only if its a part of IO1/IO2 family)

EBS Encryption

EFS

Managed network file system that can be mounted on Many EC2(s)

EFS works with EC2s on Multi AZ

Highly available scalable and expensive

Content Management , Wordpres

Only for LINUX BASED AMI

Scalability and Availability

VERTICAL:

- Increase size of instance to take more volume
- T2 small to T2 large

HORIZONTAL

- Increase number of instances LIKE MORE SERVER
- Use case for a Distributed System

HIGH AVAILABILITY goes hand in hand with HORIZONTAL SCALING

ELASTIC LOAD BALANCER

PROVIDE SSL TERMINATION

STICKY COOKIES

PRIVATE TRAFFIC IN YOUR CLOUD

ELB IS **MANAGED** LOAD BALANCER

CLB - HTTP, HTTPS, TCP, SSL

(Supports layer 4 (tcp) and HTTP (layer 7))

ALB - HTTP, HTTPS, WEB SOCKET

(Layer 7 only)

Machines grouped in a Target Group

Has support for HTTP2 and Web Sockets

Supports Route routing

Fixed Hostname

X-Forwarded-Port

X-Forwarded-proto

NLB

Forward TCP/UDP traffic

Handle millions of requests

Has one static ip for AZ

They are used for **EXTREME PERFORMANCE**

The Target Group for an NLB can be

-EC2

- Application Load Balancer

- Private IP address

NLB - TCP , TLS

GWLB - (at the n/w layer) IP Protocol

All traffic to go through a firewall

Maybe you want to inspect the traffic before it goes to your application

Uses the Geneve Protocol - 6081

Target Groups

EC2 Instances

Private IP Addresses

Stick Session

2 requests to ELB will go through the same instances

Use case that user must not lose session data

Application based cookie AWSALBAPP

Duration based cookie (AWSALB for ALB , AWSCLB FOR CLB)

Cross Zone Load Balancing

Cross Zone

Load balancer across AZ

Equally distributed across AZ

WITHOUT CROSS

SSL /TLS Basics

Traffic will be encrypted

SNI - Determining which certificate to load when there are multiple certificates

Works only for ALB AND NLB

APPLICATION LOAD BALANCER

APPLICATION LB IS LAYER 7

LOAD BALANCING TO MULTIPLE APPLICATIONS ACROSS MACHINES

LOAD BALANCING TO MULTIPLE APPLICATIONS ON THE SAME MACHINE

SUPPORT FOR HTTP/2 AND WEB SOCKET

SUPPORT REDIRECTS

AMAZON RDS

Multi-AZ keeps the same connection string regardless of which database is up.

You can not create encrypted Read Replicas from an unencrypted RDS DB instance.

Read Replicas add new endpoints with their own DNS name. We need to change our application to reference them individually to balance the read load

Read Replicas cannot help with disaster recovery.

RDS ENCRYPTION

- Encryption at Rest AES 256 Encryption
- *If the master is not encrypted ,then the read replica also cannot be encrypted*
- Transparent Data Encryption is for Oracle and SQL server

IN FLIGHT Encryption

SS Certificates to encrypt in flight

RDS Operations

Snapshot of unenc is unec

Snapshot of enc is enc

How to enc an unenc RDS

Create a snap of the unenc

Copy the snap

Enable enc for the snap

Restore database from the snap

Network security

Deployed within a private subnet

Uses sec groups to determine which IP address /sec to be allowed to communicate

IAM policies

Allows who can manage

Username /Password

For Mysql and Postgres you using IAM authentication (auth token)

Amazon Aurora

5x Performances Improvement over MySQL

3x Performance Improvement over PosGres

Storage grows in increments of 10gb upto 128tb

Upto 15 replicates compared MySQL 5

Replication process is faster

Instantaneous Failover

Good for savings

Saves 6 copies of data over 4AZ

How does Aurora Interface with multiple instances

We have a shared storage volume auto -expanding from 10gb to 64gb

WRITER ENDPOINT ALWAYS POINT TO MASTER

CLIENTS WILL ALWAYS TALK TO THE WRITER ENDPOINT THAT WOULD REDIRECT TO THE REQUIRED INSTANCE

Now you can many Read Replicates

READER ENDPOINT , helps with conn balancing

It will help connect to the right replicate

Only after deleting reader and writer instance , you can delete the aurora db

Advanced Concepts

Auto Scaling

Scenario: 3 Aurora Instances (1 WRITING , 2 READING).

Assume there are many reads , the reader endpoints will be extended (scaled up)

Custom Endpoint

Scenario: DBR5 Large and DBR52XLarge . Some instances larger than the others

Assume you assign custom endpoint on the larger instances

These instances are powerful. They will run Analytical queries

Reader endpoint is not used and this custom endpoint is used for these specific workloads

A custom endpoint for handling complex heavy workloads. So you query only subset of your workloads for certain scenarios

Serverless

Automated db initialization and auto scaling based on workloads
Infrequent , unpredictable workloads are the common scenario for this
No Capacity Planning
Pay as you go
Client talks to Proxy FLEET and instances are created on the fly

Aurora Multi Master

Immediate failover for Write instances

Global Aurora
Very good for DR
Simple to put in place
16 RR for each region
RECOVERY TIME of 1 min in case of failure
Helps for decreasing Latency

Aurora machine learning

ML BASED Prediction
Simple Integration (SAGE MAKER AND AMAZON COMPREHEND)
Use cases:
Fraud detection ,sentiment analysis,ad targeting , product recommendations

ElastiCache

Cache must invalidation strategy for more recent data
Session data can be stateless where data can be fetched from other instances
Storing Session Data in ElastiCache is a common pattern to ensuring different EC2 instances can retrieve your user's state if needed.

Cluster mode in Redis ensure high scalability and availability

Redis v/s Memecache

Redis	Memcached
Multi AZ Failover	No Multi AZ Failover

Read replicas to scale reads and thereby high availability	No High availability
Data Durability through AOF Persistence	No High Persistence
Backup and Restore	No Backup and Store
No multi threaded arch	Multithreaded Arch

All Cache in elasti cache DO NOT SUPPORT IAM AUTHENTICATION
IAM is used for AWS API Security
Support SSL security for in flight security

Memecache SASL security

3 types of loading data

- Lazy Loading
- Write through
- TTL using Session Store

Use Case- Scenario - Gaming leaderboard

Feature: Sorted sets (Uniqueness and element ordering)

Each time - Ranked in real time and then added

What is DNS ?

Domain Name System

Translation of Human friendly address to IP address that points to the server

Domain Register : AMAZON Route 53 , GoDaddy

-DNS Records define how you want to route traffic to your domain

Record Types

Record Type	Description
A	MAPS HOSTNAME TO IPV4 (example.com -> 10.11.12.13
AAAA	MAPS HOSTNAME TO IPV6
CNAME	MAPS HOSTNAME TO HOSTNAME(Target

	<p>maybe an A or AAAA)</p> <p>You can't create a CNAME for example.com but you can create a CNAME for www.example.com.</p> <p>Basically you can;t create CNAME FOR TLD</p>
NS	<p>Name Servers (how traffic is routed to domain)</p>

Hosted Zones

Public : Routing in the Internet
Private : Routing in the VPC.

CNAME AND ALIAS

CNAME Maps hostname to hostname but works only for non-root domain
ALIAS
Maps hostname to hostname and works for root and non-root domain
You can map hostname to AWS resource

Targets: ELB, CDN,API Gateway , EBeanstalk, VPC Interface , Global Accelerator
NOT FOR EC2 DNS NAME

ROUTING POLICIES

Route 53 Supports the following policies
SIMPLE, WEIGHTED ,FAILOVER,LATENCY
BASED,GEO-LOCATION,MULTI-VALUED,GEO-PROXIMITY

Routing policy	
Simple	Routes to a single resource. If multiple values returned , then client redirects to a random resource
Weighted	Percentage of requests that can go to a specific instances .Use cases : Load balancing and testing new app versions. Assign weight 0 to stop sending requests
Latency	Redirect to resource closest to the user

AWS S3

SSE-S3

“x-amz-server-side-encryption:AES256” => Object + AWS managed key

SSE-KMS:

“X-amz-server-side-encryption:aws:kms” (*User Control + Audit Trail*)

SSE-C

HTTPS must be used

Encryption key in HTTP header

Client side data key used

Client Side encryption

User is charge of encrypting the data

User is in charge of decryption the data

Encryption in Transit

Solution Arch Discussion

[6/13, 19:57] Ananth: 13/6

Stateless Architecture Solution Notes

Scaling

- Vertical scaling by increasing server strength M5 large instead of T2 micro
- Horizontal scaling by adding more servers

(Using an Elastic IP)

- Setup route 53 for client facing url using A record instead of elastic IP
 - For Disaster scenarios add a ALB with health checks in front of the EC2 instances
- Security group to restrict traffic between EC2 and ALB

Using alias record in Route 53 to point to ALB

To avoid manually adding and removing instances use an Auto scaling Group.

Suppose AZ-1 is down.

We need to modify and use a Multi Az arch

Auto scaling Group

And inside that

az-1

az-2

az-3

To cut costs, you can reserve instances because we'll always need at least 2 instances per AZ

-

- Different ways to achieve this

- 1) user cookies
- 2) elastic cache
- 3) RDS

We can't use EBS as EBS is restricted to an AZ. So if request goes to another AZ our state is lost.

Golden AMI for fast instantiating
Bootstrapping with scripts for dynamic configuration
Restoring data from a snapshot where data and schemas are ready (RDS)
Restoring from EBS snapshot where disk will be formatted and have data
[6/13, 20:07] Ananth: Stateful Architecture notes

(Shopping cart website)

Initial arch

Client
Multi AZ
Route 53

What's the problem?
Every request will go to a different instances due to the front facing ALB

How to fix?

Option 1
Sticky session an ELB feature

Cons
But if Ec2 instance is terminated we lose data

Option 2

Use user cookies

Client sends cookies that has shopping data.

Each server will know shopping cart content

Cons:
Heavy requests
Security Risks

Disclaimer:
Validate cookies
Cookies max size :5KB

Option 3:

Server session
Client sends session ID

In the back end we have elastic cache
Elastic cache uses session ID to get shopping data
Another option is using Dynamo DB

Pros
More secure

Scaling reads

Use RDS master
Use RDS replicas

Another option
Use Write through cache mechanism
Less traffic on RDS and improve performance

For Disaster Recovery

Use a Multi AZ ALB
Use Multi AZ RDS
Use Multi AZ for Elastic Cache
Use Security groups in ELB and restrict traffic between source and destination
[6/13, 20:10] Ananth: Stateful Web App : Another use case

Use case

Upload pictures on a word press website on AWS like a blog

Initial Arch

Route 53
RDS in backend to store data
Multi AZ ALB for distribution of load
EC2 instances in an Auto scaling Group
[6/13, 20:11] Ananth: How to store images?

Use an EBS Volume
[6/13, 20:11] Ananth: And we have one instance for example
[6/13, 20:11] Ananth: Problem?
[6/13, 20:11] Ananth: If we have 2 instances
[6/13, 20:11] Ananth: Two EBs volumes
[6/13, 20:12] Ananth: But if the next request goes to second instance
[6/13, 20:12] Ananth: Then first image is not accessible
[6/13, 20:12] Ananth: How to solve ?
[6/13, 20:12] Ananth: Use EFS
[6/13, 20:12] Ananth: Along with ENIs
[6/13, 20:12] Ananth: ENIs will act as a shared storage mechanism

AMAZON S3

- Max Object size is 5TB
- Buckets are defined at **regional** level
- If uploading more than 5GB at a time use MULTI-PART UPLOAD
- Versioning is enabled at the BUCKET LEVEL
- Any file that is not versioned prior to enabling versioning will have version :null

AMAZON S3 ENCRYPTION

- SSE:S3 (Keys handled and managed by AWS)

Objects are encrypted server side. AES-256 uses and set header “x-amz server-side encryption”.”AWS256”

- SSE:KMS (Leverage AWS Key management Service to manage encryption keys)

Advantages include control over the keys and audit trail. We must set the header “x-amz-server-side-encryption”.”aws:kms”

- SSE-C (When you want to manage your own encryption keys)

Using data keys fully managed by customers outside AWS . Amazon does not store the encryption key that you provide.HTTPS is a must and Encryption in transit. Key for every request.

- Client Side Encryption

You encrypt the object before uploading to S3 using some client libraries . Client must also decrypting the data. Customers manages the entire encryption Cycle

Http: not encrypted and HTTPS is encrypted

Amazon S3 Security

User based IAM access control that allows certain APIs only

Resource based access control

Buckets can be protected by blocking access through ACLS

S3 can be accessed by VPC endpoints (without internet)

S3 can give you help in access log and audit trails

API logs can be logged

MFA delete can be enabled

Pre-Signed URLs for protecting resources

S3 Website

<bucketname>.s3-website-<AWS Region>.amazonaws.com

S3 MFA

Use Case : Must be enabled/ disabled by BUCKET OWNER . Use ROOT Account

S3 DEFAULT ENCRYPTION

Use Case: When all items in your bucket are encrypted by default when you upload/

S3 ACCESS LOGS

Do not put your logging bucket in the monitoring bucket

Go to Properties > Server Access Logging > Enable

Bucket logging will be enabled in target bucket

S3 Replication CRR and SRR

For enabling CROSS REGION REPLICATION , you must enable VERSIONING

In source and destination

Bucket can be in diff aws accounts and the copying is asynchronous

Use case :crr (compliance , lower latency access , replication across accounts) , srr (DR, log aggregation)

ONLY NEW OBJECTS ARE REPLICATED BY DEFAULT

TO REPLICATE OLD OBJECTS BATCH REPLICATION (or for failed objects)

DELETE OPERATION (FROM SOURCE TO TARGET UPDATE THE DELETE MARKERS)

There is no chaining of replication

GO TO MANAGEMENT > REPLICATION RULES

S3 PRE SIGNED URLS

DEFAULT OF 3600secs

Expires-in attribute

S3 STORAGE CLASSES

GENERAL PURPOSE

Freq access data

2 failures

Mobile gaming big data analytics and content distribution

Low latency and high throughput

INFREQUENT ACCESS

Less freq acces but rapid access when needed

Low cost than S3 standard

There's a cost on retrieval

Use case : DR recovery and backups

ONE ZONE FREQ ACCESS

High durability

Data lost if AZ is destroyed

Use case : secondary copy of backup

GLACIER INSTANT RETRIEVAL

Low cost obj storage

Pay for retrieval cost and storage

GREAT FOR DATA ACCESSED ONCE A QUARTER

MINIMUM STORAGE DURATION OF 90 DAYS

GLACIER FLEXIBLE RETRIEVAL

1-5 mins FLEXIBLE (EXPEDITED)

3-5 HOURS(STANDARD)

5-12 (BULK)

Minimum storage of 90 days

GLACIER DEEP ARCHIVE

Long: 12 hours

Bulk : 48 hours

Minimum : 180 days

INTELLIGENT TIRING

Move objects based on tiers based on access usage
No fee for retrieval

FREQUENT ACCESS TIER : DEFAULT

INFREQUENT ACCESS TIER : OBJECTS NOT ACCESSED FOR 30 DAYS

ARCHIVE INSTANCE ACCESS TIER : OBJECTS NOT ACCESSED FOR 90 DAYS

ARCHIVE ACCESS TIER : OBJECTS CONFIGURABLE FROM 90 - 700 + DAYS

DEEP ARCHIVAL : 180 - 700+ DAYS

Objects can be assigned classes and objects can be moved through classes through lifecycle configurations

Durability: how many times it can be lost

CDN

AWS SQS MESSAGE

UNLIMITED MESSAGES

Consists of Producer and Consumer

SQS Queue decouples a Producer and Consumer

Application Decoupling ⇒ SQS

DEFAULT : 4 DAYS MAX 14 DAYS

256KB LIMITATION

Can have Duplicate Messages
Can also have Out of Order Messages

SQS Consumer can receive upto 10 messages at a time

SQS Producer

Messages are sent using SDK
SendMessageAPI
Message is persisted till its read and deleted

SQS Consumer
10 messages at a time

Queue Length for ASG IS THE CLOUDWATCH Metric
ASG Capacity is thus increased
More the messages , more the EC2 instances

ECS

DOCKER IMAGES are stored in ECR

Container Management

ECS

EKS

FARGATE

Serverless and we do not manage infra
AWS run ECS tasks based on demand/ RAM

IAM ROLES FOR ECS

EC2 INSTANCE PROFILE

Used by eCS agent
Makes API calls to ECS service
Sends logs to CloudWatch
Pull docker from ECR
Interact with SSM

EC2 Task role

Allows each task to have a specific role
Use diff role for diff ECS service
Task role defined in Task definition

EFS

Mount file systems on ECS tasks
Tasks running in any AZ can share the same data
Works for ECS and Fargate

ECR(storing images)

ec2

AWS Lambda

LAMBDA

128MB - 10GB MEMORY ALLOCATION
900 SECS MAX EXECUTION TIME
ENV = 4KB
DISK CAPACITY

DEPLOYMENT : 50MB
MAX 250MB

LAMBDA @EDGE

Web Security
SEO

Dynamic Content
Bot mitigation
Authentication and Authorization
Intelligently Route across Origin
A/B Testing
User Prioritization
User Tracking and Analysis
Real time Image Transformation

Amazon Dynamo DB

EACH TABLE HAS PRIMARY KEY
INFINITE NO OF ROWS
MAX SIZE OF ITEM : 400KB

Provisioned Mode : You specify no of Read and Writes
On Demand Mode : No Capacity Planning needed

Dynamo DB Accelerator

Its a cache
Solves Read congestion by caching
5 mins TTL
DAX: FOR DYNAMO DB
ELASTIC CACHE : FOR APPS FOR COMPUTATION STORE DATA

Dynamo db streams

Stream data can be sent to KINESIS DATA STREAMS , AWS LAMBDA , KINESIS CLIENT
LIBRARY APPLICATION

Dynamo DB Global Tables

Access AZ
LOW LATENCY

INDEXES ALLOW YOU TO QUERY OTHER COLUMNS
PKEY IS MY DEFAULT
GSI AND LSI
GSI:
LSI

DYNAMODB TRANSACTION

WRITE TO TWO TABLES OR NONE AS A PART OF A WRITE

API GATEWAY
ENDPOINT TYPES

REGIONAL
PRIVATE
EDGE

DynamoDB Streams enable DynamoDB to get a changelog and use that changelog to replicate data across replica tables in other AWS Region**S**.

AWS Cloud Watch

Type of Metrics : CPU Utilization
Metrics belong to Namespaces
Dimension is an attribute of metric

10 dim per metric

EC2 Instance have metrics every 5 mins[DEFAULT]

Detailed Monitoring you get data every 1 min(when you wanted to scale fast for ASG)

Free tier allows 10 detailed monitoring metrics

EC2 memory is not pushed by default

CloudWatch Custom Metric

Custom Metrics - YOU CAN DEFINE YOUR OWN

USE API CALL - PutMetricData TO CREATE A CUSTOM METRIC

You can use dimensions to segment Metrics

INSTANCE ID and ENVIRONMENT NAME

Metric Resolution - standard 60 sec and High Res (1/5/10/30) using the Storage Resolution

Cloud Watch Metric Streams :

Continuously stream metrics to a destination of your choice. - AMAZON KINESIS FIREHOUSE

Cloudwatch dashboard : You can access key metrics , alarms and view graphs.

View metrics across AWS regions

They are global

Multiple account / Multiple widgets

Cloudwatch logs

Saving logs ⇒ CQW LOGs

Group logs in Log Groups

Each group has log streams

Log expiry policy (never / 30 days)

Export logs to S3, KINESIS DATA STREAM, S FIREHOSE , LAMBDA , ELASTIC SEARCH

YOU CAN USE FILTER EXPRESSIONS

METRIC CAN BE LINKED IN CW ALARMS

CW INSIGHTS WHERE YOU CAN QUERY LOGS AND ADD TO DASHBOARD

S3 EXPORT

CAN TAKE ABOUT 12 HOURS

API CALL : CREATE EXPORT TASK

FOR STREAMING - Use Log Subscription

CLOUD WATCH ALARM STATES

OK

INSUFFICIENT DATA

ALARM

CLOUD WATCH ALARM TARGETS

EC2

AUTO SCALING

SNS

STUFF YOU CAN DO WITH THE CLOUD WATCH ALARM

CHECK THE EC2 INSTANCE AND THE UNDERLYING HARDWARE
ALARM ON TOP OF METRIC LOF FILTERS

AMAZON EVENT BRIDGE (CLOUD WATCH EVENTS)

- SCHEDULE SCRIPTS IF YOU LIKE FOR EVERY HOUR
- REACT TO AN EVENT

Event bus allows cross access using resource policies

Say you want to capture all authO events

- 1) Create a rule
- 2) Specify the bus
- 3) Now either specify a schedule / or for an event
- 4) Specify your source event
- 5) Specify the type of the event
- 6) Specify the target
- 7)

CLOUD WATCH CONTAINER INSIGHTS

EXTRACT METRIC AND LOGS INTO DASHBOARD

CLOUD WATCH LAMBDA INSIGHTS

MONITORING AND TROUBLESHOOTING SOLUTIONS FOR SERVERLESS IMPLEMENTATION

COLLECT AND SUMMARIZE METRICS SUCH AS CPU USAGE TIME

CONTRIBUTOR INSIGHTS

FIND THE TOP CONTRIBUTOR - FIND WHO'S AFFECTING PERFORMANCE
FIND BAD HOSTS

APPLICATION INSIGHTS

FIND ISSUES WITH MONITORED APPLICATIONS
POWERED SAGE MAKER
REDUCE TIME TO TROUBLESHOOT
SENT TO EVENT BRIDGE AND SSM up Center

AWS CLOUD TRAIL

GOVERNANCE, COMPLIANCE AND AUDIT FOR AWS ACCOUNT
ENABLED BY DEFAULT
GET HISTORY OF EVENTS/API CALLS
PUT LOGS FROM CLOUD TRAIL INTO CLOUD WATCH
LOGS FROM SINGLE REGIONS OR ALL REGIONS(DEFAULT)

THREE EVENTS

(1) MANAGEMENT EVENTS (Whenever someone configures security , create a subnet) ?

CAN SEPARATE READ AND WRITE EVENTS

(2) DATA EVENTS (Not logged by default)

GetObject,PutObject
CAN SEPARATE READ AND WRITE EVENTS

(3) INSIGHTS EVENTS

Detect unusual activity in your account
Inaccurate Resource provisioning

Will analyze normal activity and detect unusual patterns

CLOUD TRAIL EVENT RETENTION
EVENTS STORED FOR 90 DAYS
FOR > 90 DAYS LOG IN S3 AND USE ATHENA

AMAZON INTERCEPT API CALLS

DELETE TABLE —LOGGED AS AN EVENT IN AMZ EVENT BRIDGE

You can then look up the Delete API Call. You can create a rule and then define an alert for this api call.

AWS CONFIG

AUDIT AND RECORD COMPLIANCE OF AWS RESOURCES
IS THERE ANY UNRESTRICTED SSH ACCESS ?
HAS ANY ALB CONFIG CHANGED OVER TIME

TYPES OF RULES
- 75 RULES OF AMAZON MANAGED CONFIG RULES
- CUSTOM RULES
0.003 PER CONFIG
0.001 PER CONFIG EVALUATION
CONFIG REMEDIATION

Let's say your monitoring and IAM keys have expired. So to mark them as non-compliant
You can trigger a REMEDIATION ACTION such that those keys can be DEACTIVATED

USE EB TO TRIGGER NOTIFICATION

AWS Security

Encryption in Flight

Data enc before send and dec by the server

SSL helps with encryption with HTTPS
Protected from the MAN IN THE MIDDLE

SERVER SIDE ENC AT REST

Data enc after being recvd by the SERVER

Server uses KMS keys for enc
Data isdec before sent

Client Side Encryption

Data is enc by client

Data is not dev by server
Data is dec by receiving client

With Client-Side Encryption, the server doesn't need to know any information about the encryption scheme being used, as the server will not perform any encryption or decryption operations.

KEY MANAGEMENT SERVICE

Able to audit KMS keys using Cloud Trail
KMS keys - (1) Symmetric (2) Asymmetric

Symmetric
(1) Single key used to encrypt and decrypt

Asymmetric

- (1) One Public key for Encrypt and One Private key for Decrypt
- (2) Use case : Enc outside AWS where one can't access KMS API

Three types of kms KEYS

- 1) AWS Managed(aws/rds , aws/ebs)
- 2) Customer Managed 1\$/ month
- 3) Customer Managed but imported
Pay 0.03\$ for API calls

AWS Managed and Customer Managed need to be rotated once a year
Customer Managed imported needs an alias to be rotated

KMS KEY POLICY

- (1) DEFAULT : Everyone access
 - (2) CUSTOM: Define the users,roles and who can administer the key.
- Use Case : Cross Account access ,Copying Snapshots across accounts

Automatic Rotation : EVERY 1 YEAR

KMS Multi-Region Keys

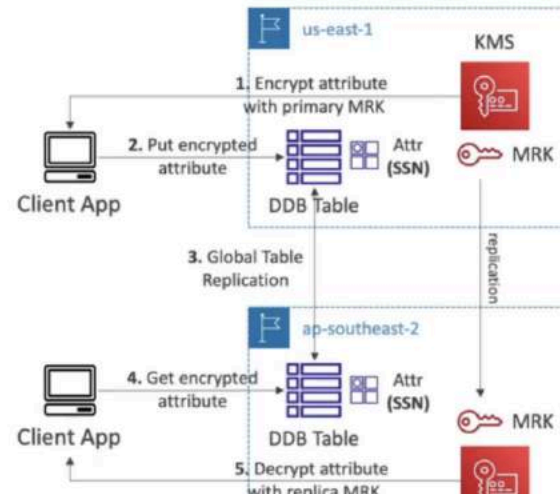


- Identical KMS keys in different AWS Regions that can be used interchangeably
- Multi-Region keys have the same key ID, key material, automatic rotation...
- Encrypt in one Region and decrypt in other Regions
- No need to re-encrypt or making cross-Region API calls
- KMS Multi-Region are NOT global (Primary + Replicas)
- Each Multi-Region key is managed independently
- Use cases: global client-side encryption, encryption on Global DynamoDB, Global Aurora



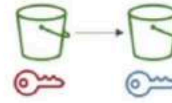
DynamoDB Global Tables and KMS Multi-Region Keys Client-Side encryption

- We can encrypt specific attributes client-side in our DynamoDB table using the **Amazon DynamoDB Encryption Client**
- Combined with Global Tables, the client-side encrypted data is replicated to other regions
- If we use a multi-region key, replicated in the same region as the DynamoDB Global table, then clients in these regions can use low-latency API calls to KMS in their region to decrypt the data client-side



S3 Replication

S3 Replication Encryption Considerations

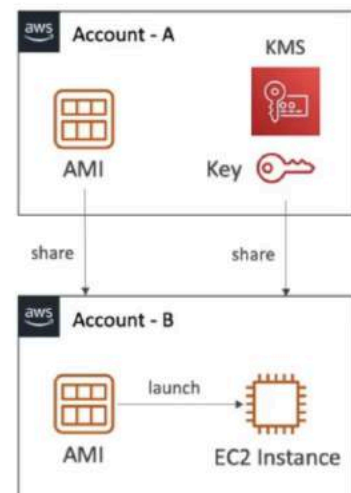


- Unencrypted objects and objects encrypted with SSE-S3 are replicated by default
- Objects encrypted with SSE-C (customer provided key) are never replicated
- For objects encrypted with SSE-KMS, you need to enable the option
 - Specify which KMS Key to encrypt the objects within the target bucket
 - Adapt the KMS Key Policy for the target key
 - An IAM Role with kms:Decrypt for the source KMS Key and kms:Encrypt for the target KMS Key
 - You might get KMS throttling errors, in which case you can ask for a Service Quotas increase
- You can use multi-region AWS KMS Keys, but they are currently treated as independent keys by Amazon S3 (the object will still be decrypted and then encrypted)

AMI Sharing Process

AMI Sharing Process Encrypted via KMS

1. AMI in Source Account is encrypted with KMS Key from Source Account
2. Must modify the image attribute to add a **Launch Permission** which corresponds to the specified target AWS account
3. Must share the KMS Keys used to encrypted the snapshot the AMI references with the target account / IAM Role
4. The IAM Role/User in the target account must have the permissions to DescribeKey, ReEncrypt, CreateGrant, Decrypt
5. When launching an EC2 instance from the AMI, optionally the target account can specify a new KMS key in its own account to re-encrypt the volumes



SSM Parameter Store

Standard and advanced parameter tiers

	Standard	Advanced
Total number of parameters allowed (per AWS account and Region)	10,000	100,000
Maximum size of a parameter value	4 KB	8 KB
Parameter policies available	No	Yes
Cost	No additional charge	Charges apply
Storage Pricing	Free	\$0.05 per advanced parameter per month
API Interaction Pricing (higher throughput = up to 1000 Transactions per second)	Standard Throughput: free Higher Throughput: \$0.05 per 10,000 API interactions	Standard Throughput: \$0.05 per 10,000 API interactions Higher Throughput: \$0.05 per 10,000 API interactions

SECRETS MANAGERS DIFFERENCE FROM SSM FORCES ROTATION EVERY X DAYS

AWS CERTIFICATE MANAGER

Easily provision , manage and deploy AWS Certificates

Provide inflight enc with HTTPS

Provides public and priv certificate (TLS)

Cannot use with EC2

How to request

- List the domain names in the the certificate
- Select type of validation (Email or DNS)
- For DNS , need to create a CNAME entry
- For Email , an email is sent to the email using WHOIS DATABASE.

There is no automatic renewal but there is an event sent to remind you daily on expiring certificates . This is received by Event Bridge

ACM certificate expiration check

Edge Optimized Endpoints - Request Routed through Cloud Front Edge Location
Regional Type Endpoints- Clients within same region
Private : Clients can only access through VPC. Resource policy is used to define this

AWS WAF

Protects against Layer 7 . http exploits
Deployed on ALB ,api gateway, cloudfront, cognito user pools , Appsync graphql API
You can then define rules in an ACL
Define an IPSET(upto 10,000 address)
Filter based on headers ,body
Use URLStrings for avoiding SQL Injection
GeoMatch to block countries
Rate based rules for ddos protection sending

WAF Does not support NLB
We can use GLOBAL Accelerator for fixed IP and WAF for ALB

AWS Shield is used for DDOS attacks

Free service for protecting against DDOS , for SYN/UDP Floods
For Layer 3 or Layer 4 attacks

Advanced
Optional advanced attack protection service
Cost 3000\$\$ per month
24/7 ACCESS TO DDOS RESPONSE TEAM

AWS Firewall Manager

Manage rules across many accounts

Comparing WAF / FIREWALL MANAGER/ SHIELD

Define web acl rules in waf
For granular protection use waf
If you want to use waf in multiple accounts , then use firewall manager
For sophisticated attacks and ddos use shield

AWS Guard Duty

- Using ML to detect Anomalies.
- Input data (VPC Flow logs , Cloud trail logs , DNS Logs ,Kube logs)

AWS Inspector

Independent Security Assessments.
You can automatically detect issues
Findings send to Sec Hub and Evt bridge
It evaluates only for ECE instances and container infrastructure
Look at package vulns .
You get a risk score

Macie:

ML and pattern matching for protect sensitive data (PII)
PII will be in S3 AND Macie will analyze and classify
Notification by Cloud watch events and then integrate to Lambda

AWS ORGANIZATION

Global service to manage multiple accounts
Main account is management account
Other accs are member accs that can be a part of only one org
Consolidate billing across all accs
We can share Reserve Instances and Saving Plan discounts across accounts

awsSourceIp : whitelist ip addresses // restrict where the client IP makes the API call **from**
awsRequestRegion : whitelist region // restrict the region where API calls are made to
awsResourceTag: Tag based security
aws:boolMultiFactorAuthPresent

IAM RESOURCE BASED POLICY IS APPLICABLE : SNS ,S3,SQS

VPC

CIDR: CLASSLESS INTERDOMAIN ROUTING

- A way to allocate IP addresses
- They help define IP ranges

Example

0.0.0.0/0 - Defining all IPS

WW.YY.XX.ZZ/25 - One specific IP

192.162.0.0/26 =====> 192.168.0.0 - 192.168.0.63 (64 IP ADDRESSES)

Understanding CIDR – IPv4

- A CIDR consists of two components
- Base IP
 - Represents an IP contained in the range (XX.XX.XX.XX)
 - Example: 10.0.0.0, 192.168.0.0, ...
- Subnet Mask
 - Defines how many bits can change in the IP
 - Example: /0, /24, /32
 - Can take two forms:


- A CIDR consists of two components
- Base IP
 - Represents an IP contained in the range (XX.XX.XX.XX)
 - Example: 10.0.0.0, 192.168.0.0, ...
- Subnet Mask
 - Defines how many bits can change in the IP
 - Example: /0, /24, /32
 - Can take two forms:
 - /8 \Leftrightarrow 255.0.0.0
 - /16 \Leftrightarrow 255.255.0.0
 - /24 \Leftrightarrow 255.255.255.0
 - /32 \Leftrightarrow 255.255.255.255

Mask basically allows part of the underlying IP to get its values from the base IP

32 => allows for **1** IP (2^0) \longrightarrow 192.168.0.0
31 => allows for **2** IP (2^1) \longrightarrow 192.168.0.0 -> 192.168.0.1
30 => allows for **4** IP (2^2) \longrightarrow 192.168.0.0 -> 192.168.0.3
29 => allows for **8** IP (2^3) \longrightarrow 192.168.0.0 -> 192.168.0.7
28 => allows for **16** IP (2^4) \longrightarrow 192.168.0.0 -> 192.168.0.15
27 => allows for **32** IP (2^5) \longrightarrow 192.168.0.0 -> 192.168.0.31
26 => allows for **64** IP (2^6) \longrightarrow 192.168.0.0 -> 192.168.0.63
25 => allows for **128** IP (2^7) \longrightarrow 192.168.0.0 -> 192.168.0.127
24 => allows for **256** IP (2^8) \longrightarrow 192.168.0.0 -> 192.168.0.255

16 => allows for **65,536** IP (2^{16}) \longrightarrow 192.168.0.0 -> 192.168.255.255

0 => allows for **All IPs** \longrightarrow 0.0.0.0 -> 255.255.255.255


Quick Memo

Octets

1st 2nd 3rd 4th

- **/32** – no octet can change
- **/24** – last octet can change
- **/16** – last 2 octets can change
- **/8** – last 3 octets can change
- **/0** – all octets can change

Understanding CIDR – Little Exercise

- 192.168.0.0/24 = ... ?
 - 192.168.0.0 – 192.168.0.255 (256 IPs)
- 192.168.0.0/16 = ... ?
 - 192.168.0.0 – 192.168.255.255 (65,536 IPs)
- 134.56.78.123/32 = ... ?
 - Just 134.56.78.123
- 0.0.0.0/0
 - All IPs!
- When in doubt, use this website <https://www.ipaddressguide.com>

<https://www.ipaddressguide.com/cidr>

Public vs. Private IP (IPv4)

- The Internet Assigned Numbers Authority (IANA) established certain blocks of IPv4 addresses for the use of private (LAN) and public (Internet) addresses
- Private IP can only allow certain values:
 - 10.0.0.0 – 10.255.255.255 (10.0.0.0/8) ← in big networks
 - 172.16.0.0 – 172.31.255.255 (172.16.0.0/12) ← AWS default VPC in that range
 - 192.168.0.0 – 192.168.255.255 (192.168.0.0/16) ← e.g., home networks

VPC in AWS – IPv4



- VPC = Virtual Private Cloud
- You can have multiple VPCs in an AWS region (max. 5 per region – soft limit)
- Max. CIDR per VPC is 5, for each CIDR:
 - Min. size is /28 (16 IP addresses)
 - Max. size is /16 (65536 IP addresses)
- Because VPC is private, only the Private IPv4 ranges are allowed:
 - 10.0.0.0 – 10.255.255.255 (10.0.0.0/8)
 - 172.16.0.0 – 172.31.255.255 (172.16.0.0/12)
 - 192.168.0.0 – 192.168.255.255 (192.168.0.0/16)
- Your VPC CIDR should NOT overlap with your other networks (e.g. corp.

VPC – Subnet (IPv4)



- AWS reserves 5 IP addresses (first 4 & last 1) in each subnet
- These 5 IP addresses are not available for use and can't be assigned to an EC2 instance
- Example: if CIDR block 10.0.0.0/24, then reserved IP addresses are:
 - 10.0.0.0 – Network Address
 - 10.0.0.1 – reserved by AWS for the VPC router
 - 10.0.0.2 – reserved by AWS for mapping to Amazon-provided DNS
 - 10.0.0.3 – reserved by AWS for future use
 - 10.0.0.255 – Network Broadcast Address. AWS does not support broadcast in a VPC, therefore the address is reserved
- Exam Tip, if you need 29 IP addresses for EC2 instances:
 - You can't choose a subnet of size /27 (32 IP addresses, $32 - 5 = 27 < 29$)
 - You need to choose a subnet of size /26 (64 IP addresses, $64 - 5 = 59 > 29$)

INTERNET GATEWAY

Internet Gateway (IGW)



- Allows resources (e.g., EC2 instances) in a VPC connect to the Internet
 - It scales horizontally and is highly available and redundant
 - Must be created separately from a VPC
 - One VPC can only be attached to one IGW and vice versa
-
- Internet Gateways on their own do not allow Internet access...
 - Route tables must also be edited!

BASTION HOST

Bastion Hosts

- We can use a Bastion Host to SSH into our private EC2 instances
- The bastion is in the public subnet which is then connected to all other private subnets
- Bastion Host security group must allow inbound from the internet on port 22 from restricted CIDR, for example the public CIDR of your corporation
- Security Group of the EC2 Instances must allow the Security Group of the Bastion Host, or the private IP of the Bastion host



NAT INSTANCES

Network address Translation

NAT Instance (outdated, but still at the exam)

- NAT = Network Address Translation
- Allows EC2 instances in private subnets to connect to the Internet
- Must be launched in a public subnet
- Must disable EC2 setting: Source / destination Check
- Must have Elastic IP attached to it



NAT Instance – Comments

- Pre-configured Amazon Linux AMI is available
 - Reached the end of standard support on December 31, 2020
 - Not highly available / resilient setup out of the box
 - You need to create an ASG in multi-AZ + resilient user-data script
 - Internet traffic bandwidth depends on EC2 instance type
 - You must manage Security Groups & rules:
 - Inbound:
 - Allow HTTP / HTTPS traffic coming from Private Subnets
 - Allow SSH from your home network (access is provided through Internet Gateway)
 - Outbound:
 - Allow HTTP / HTTPS traffic to the Internet
-

NAT GATEWAY

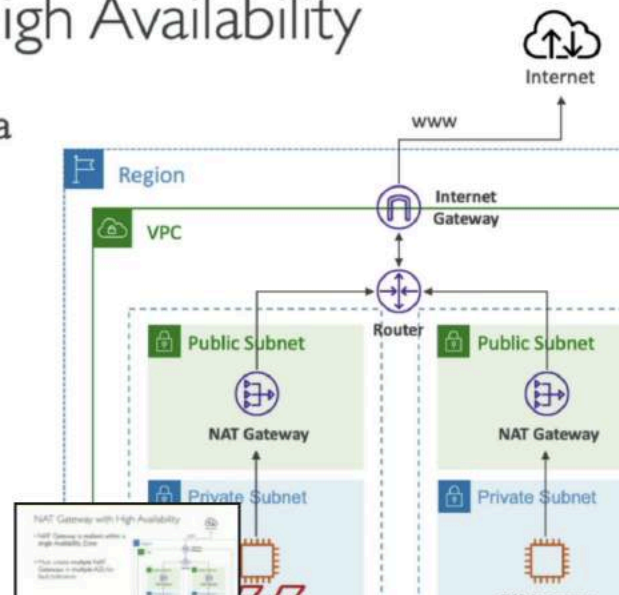
NAT Gateway



- AWS-managed NAT, higher bandwidth, high availability, no administrative overhead
- Pay per hour for usage and bandwidth
- NATGW is created in a specific Availability Zone, uses an Elastic IP
- Can't be used by EC2 instance in the same subnet (only from other subnets)
- Requires an IGW (Private Subnet => NATGW => IGW)
- 5 Gbps of bandwidth with automatic scaling up to 45 Gbps
- No Security Groups to manage / required

NAT Gateway with High Availability

- NAT Gateway is resilient within a single Availability Zone
- Must create multiple NAT Gateways in multiple AZs for fault-tolerance



NAT Gateway vs. NAT Instance

	NAT Gateway	NAT Instance
Availability	Highly available within AZ (create in another AZ)	Use a script to manage failover between instances
Bandwidth	Up to 45 Gbps	Depends on EC2 instance type
Maintenance	Managed by AWS	Managed by you (e.g., software, OS patches, ...)
Cost	Per hour & amount of data transferred	Per hour, EC2 instance type and size, + network
Public IPv4	✓	✓
Private IPv4	✓	✓
Security Groups	✗	✓
Use as Bastion Host?	✗	✓

More at: <https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-comparison.html>

NACL

Network Access Control List (NACL)



- NACL are like a firewall which control traffic from and to subnets
- One NACL per subnet, new subnets are assigned the Default NACL
- You define NACL Rules:
 - Rules have a number (1-32766), higher precedence with a lower number
 - First rule match will drive the decision
 - Example: if you define #100 ALLOW 10.0.0.10/32 and #200 DENY 10.0.0.10/32, the address will be allowed because 100 has a higher precedence over 200
 - The last rule is an asterisk (*) and denies a request in case of no rule match
 - AWS recommends adding rules by increment of 100
- Newly created NACLs will deny everything
- NACL are a great way of blocking a specific IP address at the subnet level

DEFAULT NACL

Default NACL

- Accepts everything inbound/outbound with the subnets it's associated with



Default NACL for a VPC that supports IPv4

Inbound Rules

Rule #	Type	Protocol	Port Range	Source	Allow/Deny
100	All IPv4 Traffic	All	All	0.0.0.0/0	ALLOW
*	All IPv4 Traffic	All	All	0.0.0.0/0	DENY

Outbound Rules

Rule #	Type	Protocol	Port Range	Destination	Allow/Deny
100	All IPv4 Traffic	All	All	0.0.0.0/0	ALLOW
*	All IPv4 Traffic	All	All	0.0.0.0/0	DENY

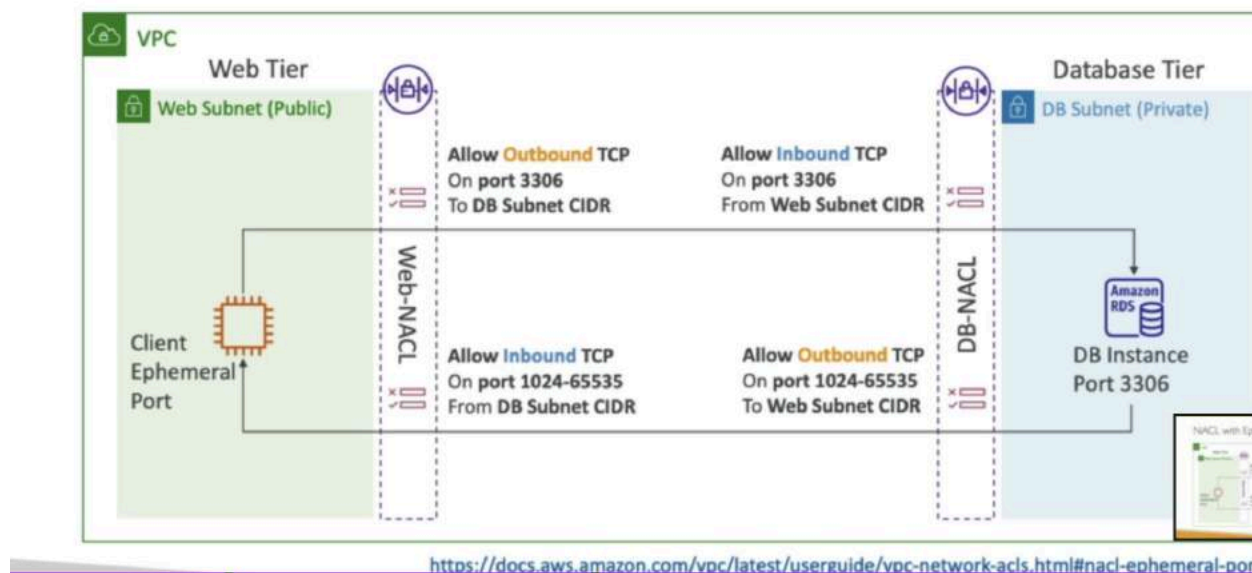
EPHERMAL PORT

Ephemeral Ports

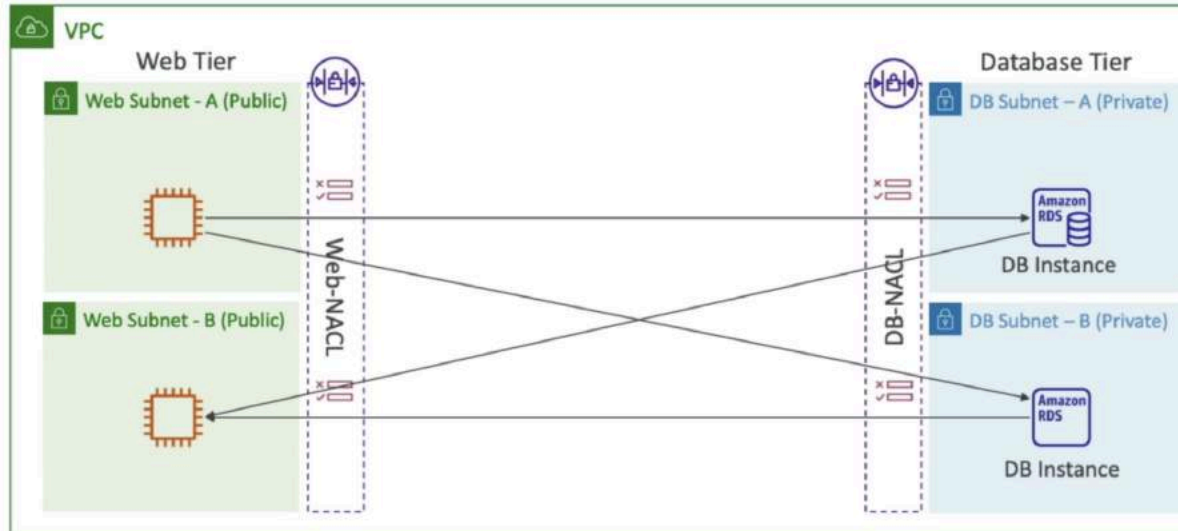
- For any two endpoints to establish a connection, they must use ports
- Clients connect to a **defined port**, and expect a response on an **ephemeral port**
- Different Operating Systems use different port ranges, examples:
 - IANA & MS Windows 10 → 49152 – 65535
 - Many Linux Kernels → 32768 – 60999



NACL with Ephemeral Ports



Create NACL rules for each target subnets CIDR



Security Group vs. NACLs

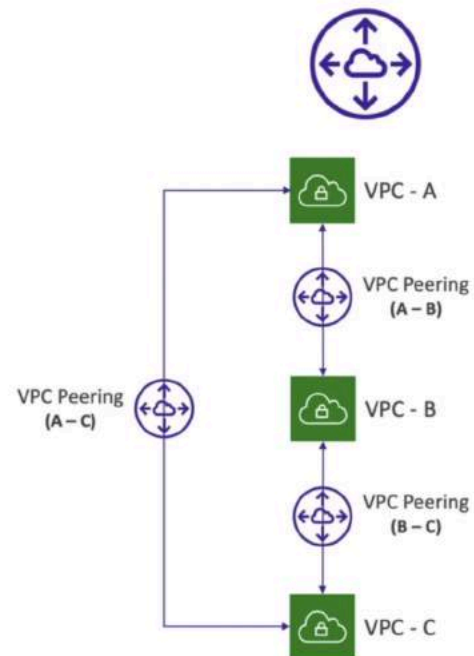
Security Group	NACL
Operates at the instance level	Operates at the subnet level
Supports allow rules only	Supports allow rules and deny rules
Stateful: return traffic is automatically allowed, regardless of any rules	Stateless: return traffic must be explicitly allowed rules (think of ephemeral ports)
All rules are evaluated before deciding whether to allow traffic	Rules are evaluated in order (lowest to highest) w deciding whether to allow traffic, first match wins
Applies to an EC2 instance when specified by someone	Automatically applies to all EC2 instances in the subnet that it's associated with

NACL Examples: <https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html>

VPC PEERING

VPC Peering

- Privately connect two VPCs using AWS' network
- Make them behave as if they were in the same network
- Must not have overlapping CIDRs
- VPC Peering connection is **NOT** transitive (must be established for each VPC that need to communicate with one another)
- You must update route tables in each VPC's subnets to ensure EC2 instances can communicate with each other

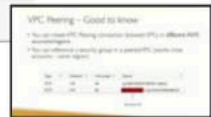


VPC Peering – Good to know

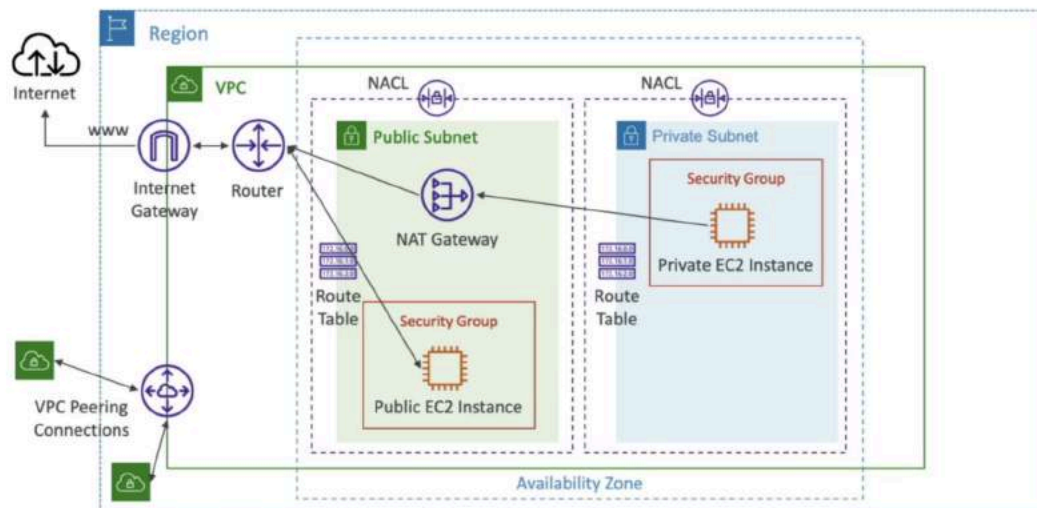
- You can create VPC Peering connection between VPCs in different AWS accounts/regions
- You can reference a security group in a peered VPC (works cross accounts – same region)

Type	Protocol	Port range	Source
HTTP	TCP	80	sg-04991f9af3473b939 / default
HTTP	TCP	80	[redacted] / sg-027ad1f7865d4be76

Account ID




VPC Peering



VPC ENDPOINT

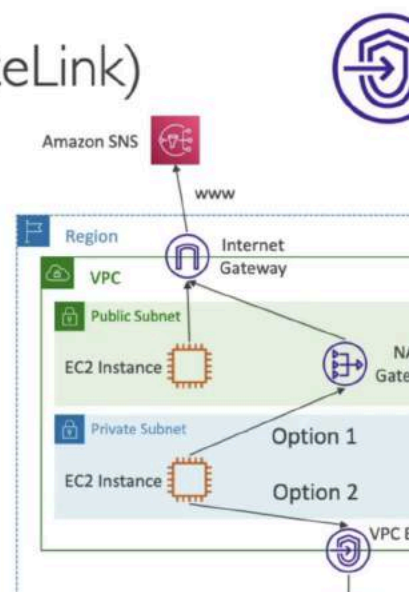
Your instances can go through your own network to access other services

For eg Amazon dYNAMOdb accessing amazon SNS

 Ultimate AWS Certified Solutions Architect Associate SAA-C03

VPC Endpoints (AWS PrivateLink)

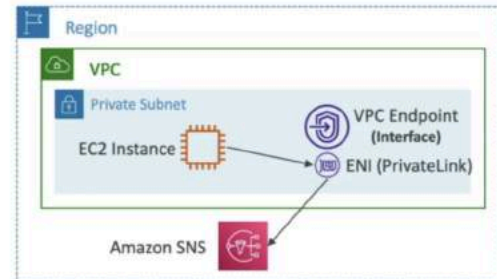
- Every AWS service is publicly exposed (public URL)
- VPC Endpoints (powered by AWS PrivateLink) allows you to connect to AWS services using a **private network** instead of using the public Internet
- They're redundant and scale horizontally
- They remove the need of IGW, NATGW, ... to access AWS Services
- In case of issues:
 - Check DNS Setting Resolution in your VPC
 - Check Route Tables



The diagram illustrates the VPC Endpoints (AWS PrivateLink) architecture. It shows a VPC with a Public Subnet and a Private Subnet. An EC2 Instance is in the Public Subnet, and another is in the Private Subnet. The Public Subnet is connected to the Internet Gateway (IGW) and the NAT Gateway. The Private Subnet is connected to the NAT Gateway. The NAT Gateway is connected to the Internet Gateway. The Internet Gateway is connected to Amazon SNS (public URL). Option 1 shows the EC2 Instance in the Private Subnet accessing Amazon SNS via the NAT Gateway and the Internet Gateway. Option 2 shows the EC2 Instance in the Private Subnet accessing Amazon SNS via a VPC Endpoint (VPC Ei).

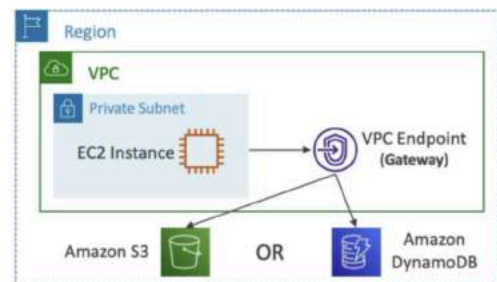
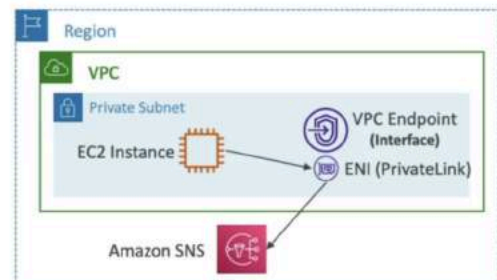
Types of Endpoints

- Interface Endpoints (powered by PrivateLink)
 - Provisions an ENI (private IP address) as an entry point (must attach a Security Group)
 - Supports most AWS services
 - \$ per hour + \$ per GB of data processed
- Gateway Endpoints



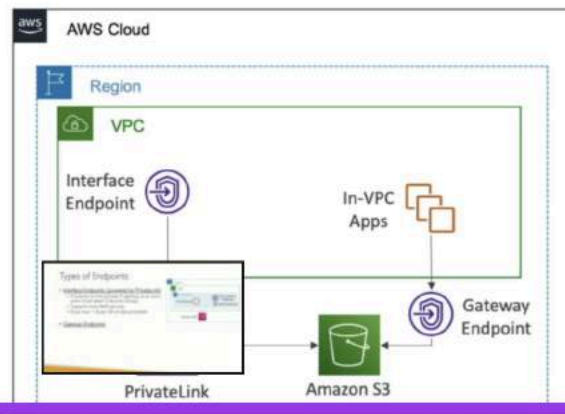
Types of Endpoints

- Interface Endpoints (powered by PrivateLink)
 - Provisions an ENI (private IP address) as an entry point (must attach a Security Group)
 - Supports most AWS services
 - \$ per hour + \$ per GB of data processed
- Gateway Endpoints
 - Provisions a gateway and must be used as a target in a route table (does not use security groups)
 - Supports both S3 and DynamoDB
 - Free



Gateway or Interface Endpoint for S3?

- Gateway is most likely going to be preferred all the time at the exam
- Cost: free for Gateway, \$ for interface endpoint
- Interface Endpoint is preferred access is required from on-premises (Site to Site VPN or Direct Connect), a different VPC or a different region



VPC Flow Logs

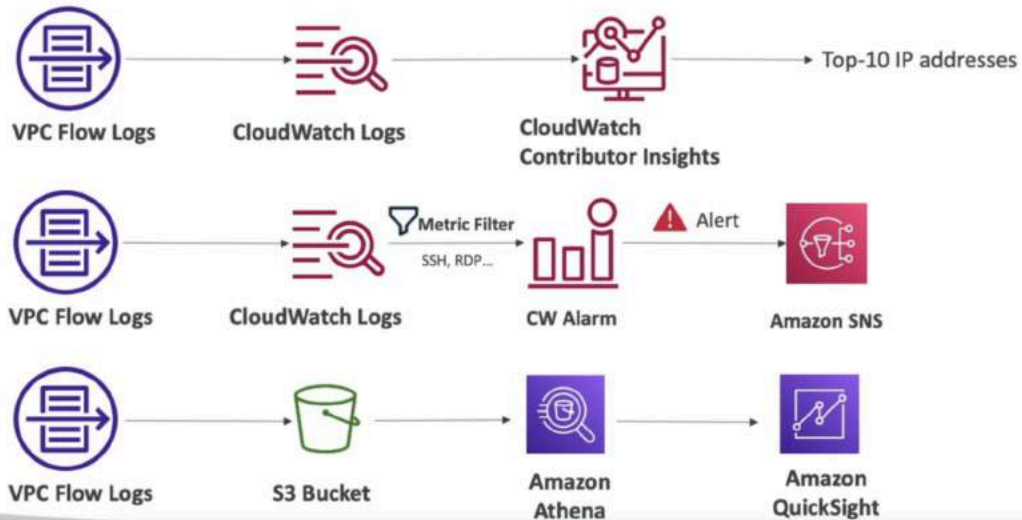


- Capture information about IP traffic going into your interfaces:
 - VPC Flow Logs
 - Subnet Flow Logs
 - Elastic Network Interface (ENI) Flow Logs
- Helps to monitor & troubleshoot connectivity issues
- Flow logs data can go to S3 / CloudWatch Logs
- Captures network information from AWS managed interfaces too: ELB, RDS, ElastiCache, Redshift, WorkSpaces, NATGW, Transit Gateway...

VPC Flow Logs Syntax

version	interface-id	dstaddr	dstport	packets	start	action
2	123456789010	eni-1235b8ca123456789	172.31.16.139	172.31.16.21	20641 22 6 20 4249	1418530010 1418530070 ACCEPT OK
2	123456789010	eni-1235b8ca123456789	172.31.9.69	172.31.9.12	49761 3389 6 20 4249	1418530010 1418530070 REJECT OK
account-id	srcaddr	srcport	protocol	bytes	end	log-status

VPC Flow Logs – Architectures



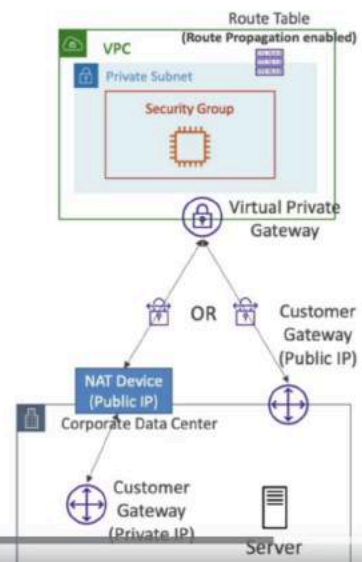
AWS Site-to-Site VPN



- Virtual Private Gateway (VGW)
 - VPN concentrator on the AWS side of the VPN connection
 - VGW is created and attached to the VPC from which you want to create the Site-to-Site VPN connection
 - Possibility to customize the ASN (Autonomous System Number)
- Customer Gateway (CGW)
 - Software application or physical device on customer side of the VPN connection

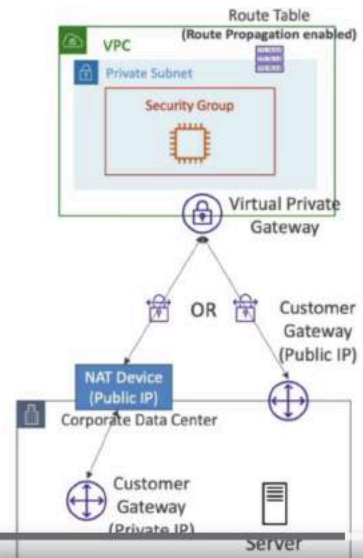
Site-to-Site VPN Connections

- Customer Gateway Device (On-premises)
 - What IP address to use?
 - Public Internet-routable IP address for your Customer Gateway device
 - If it's behind a NAT device that's enabled for NAT traversal (NAT-T), use the public IP address of the NAT device
- Important step: enable Route Propagation for the Virtual Private Gateway in the route table that is associated with your subnets



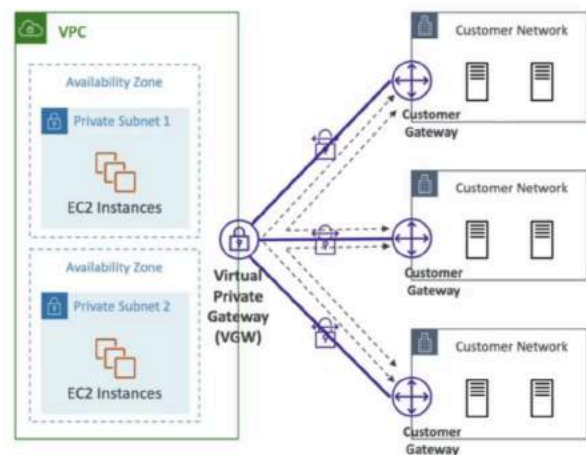
Site-to-Site VPN Connections

- Customer Gateway Device (On-premises)
 - What IP address to use?
 - Public Internet-routable IP address for your Customer Gateway device
 - If it's behind a NAT device that's enabled for NAT traversal (NAT-T), use the public IP address of the NAT device
- Important step: enable Route Propagation for the Virtual Private Gateway in the route table that is associated with your subnets
- If you need to ping your EC2 instances from on-premises, make sure you add the ICMP protocol on the inbound of your security groups



AWS VPN CloudHub

- Provide secure communication between multiple sites, if you have multiple VPN connections
- Low-cost hub-and-spoke model for primary or secondary network connectivity between different locations (VPN only)
- It's a VPN connection so it goes over the public Internet
- To set it up, connect multiple VPN connections on the same VGW, setup dynamic routing and configure route tables



Direct Connect (DX)



- Provides a dedicated private connection from a remote network to your VPC
- Dedicated connection must be setup between your DC and AWS Direct Connect locations
- You need to setup a Virtual Private Gateway on your VPC
- Access public resources (S3) and private (EC2) on same connection
- Use Cases:
 - Increase bandwidth throughput - working with large data sets – lower cost
 - More consistent network experience - applications using real-time data feeds

© Stéphane Maarek

Ip4 and ipv6

Hybrid (on prem and cloud)

Direct Connect Gateway

- If you want to setup a Direct Connect to one or more VPC in many different regions (same account), you must use a Direct Connect Gateway



1x



2:55 / 6:36

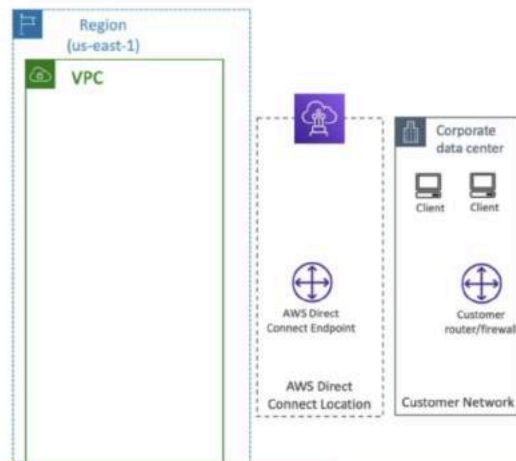


Direct Connect – Connection Types

- **Dedicated Connections:** 1 Gbps, 10 Gbps and 100 Gbps capacity
 - Physical ethernet port dedicated to a customer
 - Request made to AWS first, then completed by AWS Direct Connect Partners
- **Hosted Connections:** 50Mbps, 500 Mbps, to 10 Gbps
 - Connection requests are made via AWS Direct Connect Partners
 - Capacity can be added or removed on demand
 - 1, 2, 5, 10 Gbps available at select AWS Direct Connect Partners
- Lead times are often longer than 1 month to establish a new connection

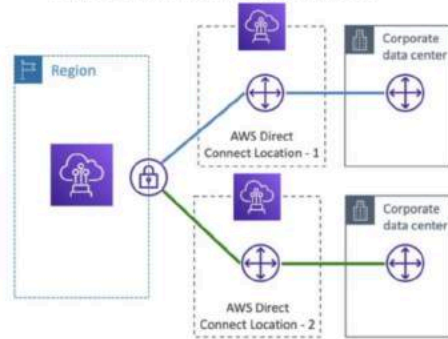
Direct Connect – Encryption

- Data in transit is not encrypted but is private
- AWS Direct Connect + VPN provides an IPsec-encrypted private connection
- Good for an extra level of security, but slightly more complex to put in place



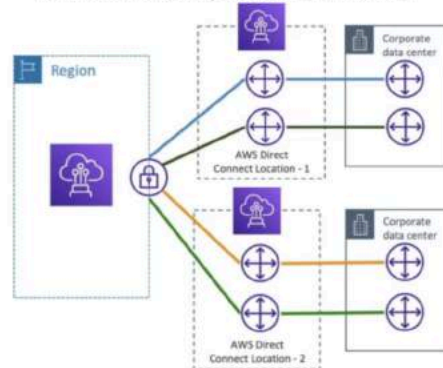
Direct Connect - Resiliency

High Resiliency for Critical Workloads



One connection at multiple locations

Maximum Resiliency for Critical Workloads

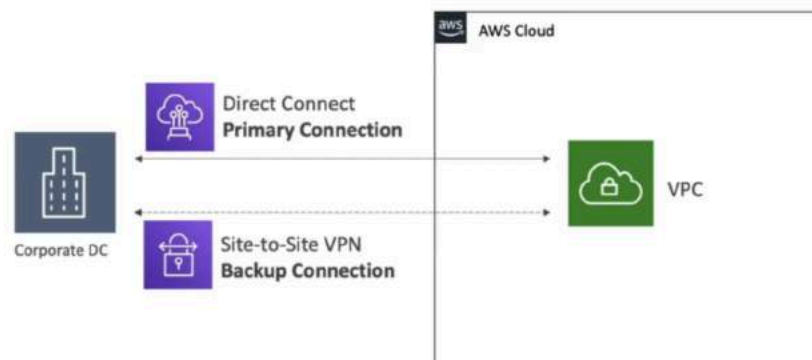


Maximum resilience is achieved by separate connections terminating on separate devices in more than one location.

© Stéphane Maarek

Site-to-Site VPN connection as a backup

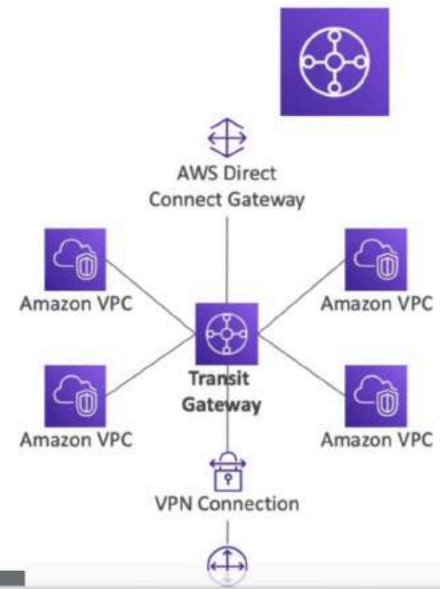
- In case Direct Connect fails, you can set up a backup Direct Connect connection (expensive), or a Site-to-Site VPN connection



© Stéphane Maarek

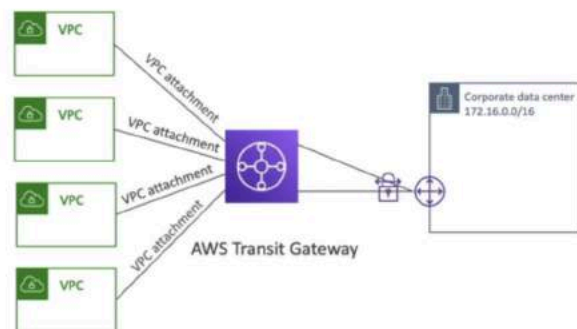
Transit Gateway

- For having transitive peering between thousands of VPC and on-premises, hub-and-spoke (star) connection
- Regional resource, can work cross-region
- Share cross-account using Resource Access Manager (RAM)
- You can peer Transit Gateways across regions
- Route Tables: limit which VPC can talk with other VPC
- Works with Direct Connect Gateway, VPN connections
- Supports IP Multicast (not supported by any other AWS service)



Transit Gateway: Site-to-Site VPN ECMP

- ECMP = Equal-cost multi-path routing
- Routing strategy to allow to forward a packet over multiple best path
- Use case: create multiple Site-to-Site VPN connections to increase the bandwidth of your connection to AWS



Transit Gateway: throughput with ECMP

VPN to virtual private gateway

$$1x \text{ VPN icon} = 1x \text{ VPC icon}$$

$$1x \text{ VPN icon} = 1.25 \text{ Gbps}$$

VPN connection (2 tunnels)

VPN to transit gateway

$$1x \text{ VPN icon} = 1x \text{ Transit Gateway icon}$$

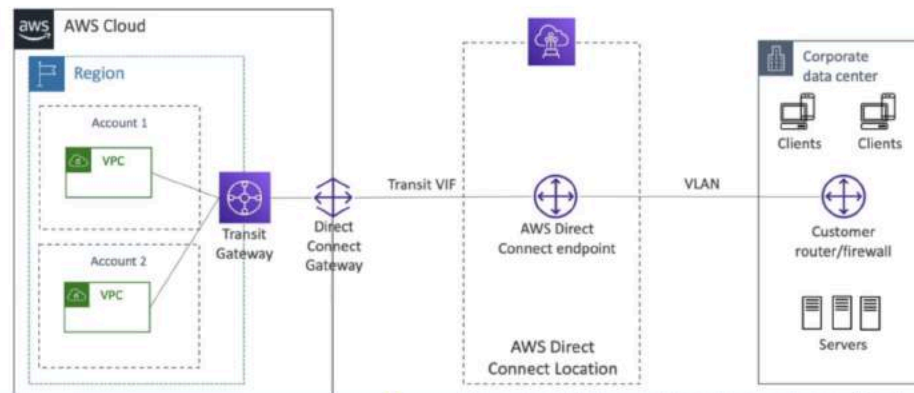
$$1x \text{ VPN icon} = 2.5 \text{ Gbps (ECMP) - 2 tunnels used}$$

$$2x \text{ VPN icon} = 5.0 \text{ Gbps (ECMP)}$$

$$3x \text{ VPN icon} = 7.5 \text{ Gbps (ECMP)}$$

1x 4:11 / 5:10

Transit Gateway – Share Direct Connect between multiple accounts



You can use AWS Resource Access Manager to share Transit Gateway with other accounts.

1x 5:04 / 5:10

CIDR Problems

i CIDR not should overlap, and the max CIDR size in AWS is /16.

Question 2:

You have a corporate network of size `10.0.0.0/8` and a satellite office of size `192.168.0.0/16`. Which CIDR is acceptable for your AWS VPC if you plan on connecting your networks later on?

☐ `172.16.0.0/12`

☒ `172.16.0.0/16`

☐ `10.0.16.0/16`

☐ `192.168.4.0/18`

Question 2 of 23

[Back to results](#)



Overview

Q&A

Notes

Announcements

Reviews

Learning tools

Now here , first thing to notes is CIDR should not overlap . So this rules out (C) and (D)

Max CIDR is /16 that is 2^{16} addresses , this rules out / 12 that is 2^{20}

So answer is (B)

ATHENA

1. Serverless query service to analyze data in S3
2. Supports CSV, JSON ,Parquet, ORC ,Avro
3. Business intelligence ,analyze logs

ATHENA PERFORMANCE IMPROVEMENT

4. Using columnar data(Apache Parquet or ORC)
5. Glue to perform data conversion
6. Compress data for quicker retrieval
7. Partition data sets by creating virtual columns
8. Use big files > 128MB to minimize overhead
9. Using federated queries to run query in CW logs, ElastiCache ,SQL server, S3

AMAZON REDSHIFT

10. Using it for analytics for OLAP
11. Column data storage
12. Based on postgresql
13. Redshift has indexes (better than athena)
14. Has Leader nodes (query planning) and compute nodes that does processing

AMAZON REDSHIFT DR and snapshots

15. Snapshots are Point in Time recovery.
16. Automated every 8 hours ,or every 5gb or a schedule.
17. You can use redshift to copy snapshots of a cluster to another AWS region

AMAZON REDSHIFT spectrum

- 18. Query data in s3 without loading it
- 19. Must have redshift cluster
- 20. Query then submitted to 1000 redshift nodes

AMAZON OPEN SEARCH SERVICE

- 21. Search for any fields even partial matches
- 22. Managed clusters or Serverless cluster (2 modes)
- 23. Native query lang
- 24. KFOSE, CW logs, IOT
- 25. Sec with cognito
- 26. Analytics using Open Search DB

AMAZON ELASTIC MAP REDUCE

- 27. Comes with all tools for big data Specialists
- 28. Auto scale clusters
- 29. Use case: Data processing , ML and using Big Data tools

AMAZON QUICK SIGHT

- 30. BI Dashboard
- 31. Per session pricing
- 32. Business analytics
- 33. Business insights on Athena , Redshift
- 34. Spice engine if data is imported in to Quick sight
- 35. Can setup Column Level security
- 36. Can integrate with RDS DB , AURORA, ATHENA, S3, Open Search, Timestream , SAAS

AMAZON GLUE

- 37. ETL service from amazon
- 38. Serverless
- 39. Convert data to parquet format for services like Athena
- 40. Glue Data catalog can run glue crawlers that will crawl dbs and get metadata
- 41. Glue bookmarks to prevent processing old data
- 42. Elastic views to replicate data across multiple data stores
- 43. gLUE BREW TO CLEAN

AMAZON DATA LAKE

- 44. Lake formation takes few days to create data lake
- 45. Uses ML transforms
- 46. AWS LAKE formations for Centralized Permissions

Kinesis data analytics

- 47. Reads from firehouse and data streams and runs sql
- 48. Time series analytics , dashboard , real time metrics
- 49. Use IFLINQing to analyze and manage stream data
- 50. FLINQ DOES NOT READ FROM FIREHOSE
- 51.

MANAGED STREAMING FOR APACHE KAFKA

Amazon Recognition

Facial search . Labeling in videos, content moderation

Content moderation to moderate certain images , safe user experience.

The image will be analyzed by Amazon and we set a confident level as low.

We now flag images

We then do manual review

Amazon TRANSCRIBE

Uses deep learning to convert speech to text

Removing personal identification number using redaction

Supports language detection

Amazon Polly overview

Turn text to speech opposite of transcribe

Customize pronunciation of words - pronunciation lexicons

Speech synthesis markup language for the type of language

Amazon Translate

Translates languages

Amazon Lex and Connect

Amazon Lex powers Alexa

Builds chatbots

Amazon connect receives calls , create contact flows

Can integrate with CRM or AWS

Amazon Comprehend

Used for NLP

Can do sentiment analysis on the text

Can find key phrases in the text

To analyze customer interaction by analyzing emails

We can see this also in Amazon .com where in reviews it consolidates and says whether customers have spoken positively /negatively about the product

Amazon Comprehend Medical

Detects and returns useful info in unstructured clinical notes such as

Physician notes, discharge summary , test results, case notes

DetectPHI to detect protected health info

Amazon Sage Maker

To build ML models

AmazonForecast

Use case : predict future sales of raincoats

50% more accurate than looking at data

Amazon Kendra

Fully managed document search service

Extract answers from doc

Learn from answers and feedback to improve results

Amazon Personalize

Think personalized recommendations

Amazon Textract

Extract text from documents or handwriting

Amazon ML Summary

Recognition - Facial recognition

Transcribe: Audio to text

Polly : Text to Audio

Sagemaker - ML Models

Textract : Extract text from documents

Kendra - ML powered search engine

Translate : for translating language

Comprehend - NLP for natural language for chatbox

Lexicon: ; for chatbots

Connect : chatbots and call centers

Forecast : Build accurate forecasts

Personalize : ML based recommendations

Amazon Cloud watch metric

1. Cloudwatch uses metrics that is a variable to monitor
2. These metrics belong to different namespaces
3. Dimension is an attribute of a metric (e.g instance id ,environment)
4. 30 is the limit of dimensions per metric
5. Metrics have timestamps
6. We can also create custom cloudwatch metric

Amazon Cloud watch metric stream

7. Cloudwatch metric streams can be used to stream data near real time to a destination of your choice for further analysis
8. You can CWM to Kinesis Firehose that can then save the data to S3 or AWS Redshift or AWS OpenSearch

Amazon Cloud watch logs

1. Define Log Groups (one of your applications)
2. Define Log instances within applications
3. Define expiration policy from 1-10 years or no expiry
4. You can send log streams to KFirehose,KStream,S3, Lambda,OpenSearch
5. Logs are encrypted by default or KMS

Amazon Cloud watch logs - SOURCES

1. AWS SDK
2. CW Agent
3. CF Unified Agent
4. EBearnstalks from App
5. ECS Collection from container
6. AWS LAMBDA
7. VPC

8. API GATEWAY
9. Cloudtrain
10. Route53
- 11.

Amazon Cloud watch Insights

1. For querying logs
2. Searching and analyzing logs
3. Prefdefines simple query
4. 25 most recent events, exceptions ,specific ip
5. Purposed based query language
6. Sort events ,limit events
7. Saved query and add to dashboards
8. Query different flog groups in diff accounts

Amazon Cloud watch export

1. Takes about 12 hours to be ready for export
2. CreateExportTask is used to initiate it
3. This is not real time
- 4.

Amazon Cloud watch Subscription

5. Get real time log events for analysis and subscription
6. Send to KStreams,KFirehose,Lambda, ECE
7. Sub filter for which logs you want o filter
8. Using CW log aggregation you can agg data from diff accounts
9. Using CROSS ACC SUBSCRIPTION you can send log events to resources in diff AWS accounts

Amazon Cloud watch Agents

- 10. By default logs from ec2 are not pushed to CW
- 11. You need to create an agent
- 12. You need to have an IAM role in EC2
- 13. 2 agents (1) CW LOG agent to CW (2) CW Unified agent

Amazon Cloud watch Alarms

- 14. Alarms are used to send notifications for a metric

Amazon Cloud watch Composite Alarms

- 15. Use AND and OR to combine multiple alarms

Amazon EVENT BRIDGE

Schedule Cron Jobs

React to event pattern

Partner Event bus(third party) ZEN DESK , D-DOG they send their events to partner event bus and you can react to events happening outside AWS

You can archive events and access across accounts.

Event Bridge can analyze events in bus and infer in schema

Schema can be versioned

Using Resource based policy you can allow/deny events another AWS acc or AWS region

Amazon Config

For compliance ,si there an unrsrestricted access to sec groups
Are buckets pub accessible

Per region service

Types of ules :(over 75)

Or create your own rules

You can dp remediation of non compliant AWS resources

Using EB to trigge notifications

CLOUD WATCH VS CLOUD TRAIL VS CONFIG

Cloud WATCH FOR MONITORING PERFORMANCE METRICS

Cloud Trail record whats happening to your resources in AWS like API calls

Config use to record config changes

AWS Organization

You can manage multiple accounts

Other accs are member accs

Get pricing benefits from agg usage

Shared reserved instances and saving plan discounts

API is available to automate

Advantages

- 1.Multi account va 1 account
2. Use tagging standards for billing purposes
3. Enable Cloud Train on all acs

SCP : IAM policies applied to OU or Accs to restrict Users and Roles

AWS Resource policy

aws:SourceIP: restricts client ip from where API calls are made
aws:RequestedRegion: some action (deny/allow) for specific regions
ec2:ResourceTag

AWS iam roles v/s RESOURCE policy

MICROSOFT AD

Found on Windows Server with AD Domain Services
Database of objects includes users printers etc
Centralized sec management , assign permissions
Objects organized as trees

Three flavors

(1) **AWS Managed Microsoft AD :**

Create your own AD in AWS , manage users, supports MF
Establish Trust connections with your own premise AD

(2) **AD Connector**

Directory gateway (proxy) to redirect to on premise AD , supports MFA

(3) **Simple AD**

If you don't have on premise AD , you can setup an AD compatible managed directory on AWS

Iam identity center - setup

If you need to **Connect to an AWS Managed Microsoft AD** , integration is out of the box

If you need **to connect to a Self Managed Directory** , you can

- (1) create a Two-way Trust Relationship using AWS Managed Microsoft AD
- (2) create an AD connector

AWS CONTROL TOWER

Easy way to setup and govern a secure and compliant multi account AWS environment based on best practices

AWS Control Tower uses AWS Organizations to create accounts

- (1) You can automate setup of your environment in a few clicks
- (2) You can automate ongoing policy management using guard rails
- (3) You can detect policy violations and remediate them
- (4) You can monitor compliance through a interactive dashboard

ENCRYPTION IN FLIGHT

Referred as TLS/SSL

Data being enc before sending and dec after receiving it

Reason : data sent via diff servers and we want to avoid man in the middle of attack

Server Side Encryption:

Here data is encrypted after receiving it by the server.

Data is decrypted before send it it

Client Side Encryption:

This time the data is en and dec at the client side
as

KMS Multi Region Keys

1. Identical KMS keys in different AWS regions that can be used interchangeably
2. Multi Region keys have same key id , key material, automatic rotation
3. You encrypt in one region and decrypt in another region
4. KMS Multi Region are not global
5. Each MR key is managed independently

AMI SHARING PROCESS VIA KMS

1. AMI in source account is encrypted with KMS key from Source Account
2. Launch permission to be modified where you provide the target aws account
3. You share the kms keys used to enc snapshot with target account/iam role
4. The IAM role in the target account must be created and it must have permission to describe key,reenc,create grant and decrypt
5. When launching an ec2 instance from the AMI, target acc can use a new KMS key in its own acc to re-encrypt the volumes

SSM PARAMETER STORE

1. Secure storage for configuration and secrets
2. Seamless encryption with KMS
3. Serverless ,scalable ,durable easy SDK
4. Version tracking of configuration/secrets
5. Security through IAM
6. Notifications with AMAZON Event bridge
7. Integration with Cloud Formation
8. Information can be stored in a hierarchy
9. Using parameter policies you can assign a TTL to force update /delete sensitive data
10. Multiple policies can be assigned at a time

AWS SECRETS MANAGER

Newer Service for storing secrets
Capability to force rotation of secrets every x days
Automate generation of secrets in rotation
Integrate with Amazon RDS

AWS CERTIFICATE MANAGER

Easily provision manage and deploy TLS certificates
Provide in flight encryption for websites
Supports for both public and private TLS certificates

Important Concepts [CIDR]

CIDR: Class Interdomain routing (CIDR) is a way to assign IP addresses

Important Concepts [VPC]

Max 5 per region
Max CIDR per VPC = 5
Min size : /28 = 16 ip addresses
Max size : /16 = 65,536 addresses

When choosing CIDR , VPC CIDR addresses should not overlap

Tenancy : Decides h/w of VPC as dedicated or shared

Important Concepts [DR]

1. Backup and Restore
2. Pilot Light
3. Warm Standby
4. Hot Site/ Multi Site

Backup and Restore: Data is periodically backed up and can be restored in the event of a disaster, but there is typically a longer recovery time.

Pilot Light: Essential systems are kept running at a minimal level, ready to be quickly scaled up when needed during a disaster.

Warm Standby: Some components of the system are running in a semi-active state, requiring minimal time and effort to fully activate during a disaster.

Hot Site/Multi-Site: Duplicate systems are fully operational in parallel with the primary system, enabling instant failover in the event of a disaster.

One simple way to remember these four DR modes in AWS is to associate them with the concept of temperature, which correlates with their level of readiness and the speed of recovery:

Backup and Restore: Think of this as the coldest state, where your data is stored away like in a freezer, waiting to be restored when needed.

Pilot Light: Imagine a pilot light on a stove. It's always on, ready to ignite quickly when needed. This mode involves maintaining a minimal setup that can be quickly scaled up in case of a disaster.

Warm Standby: Picture a warm cup of tea or coffee. It's not piping hot but warmer than room temperature. In this mode, certain components are operational but not fully active. They can be quickly activated to bring the system online.

Hot Site/Multi-Site: Visualize something actively burning or heated, like a fire. This mode involves having fully operational duplicate systems running in parallel, ready to take over instantly if the primary system fails.

Associating each mode with a temperature-related analogy can help you remember the characteristics and readiness level of each DR mode in AW

ChatGPT

AWS DataSync supports syncing data between on-premises storage and Amazon S3, Amazon EFS, or Amazon FSx for Windows File Server

Exam Concepts [DR]

Read each service's FAQ

- FAQ = Frequently asked questions
- Example: <https://aws.amazon.com/vpc/faqs/>
- FAQ cover a lot of the questions asked at the exam
- They help confirm your understanding of a service



Course	
<input checked="" type="checkbox"/>	386. Arch 3
<input type="checkbox"/>	387. 11
<input type="checkbox"/>	388. 4
<input type="checkbox"/>	389. 2
<input type="checkbox"/>	390. - No 11
<input type="checkbox"/>	391. 2
<input type="checkbox"/>	Prac Solu

Links to Whitepapers

Links to Whitepapers

1. Architecting for the cloud:

https://d1.awsstatic.com/whitepapers/AWS_Cloud_Best_Practices.pdf (Archived)

2. Whitepapers related to well-architected framework are mentioned here:

<https://aws.amazon.com/blogs/aws/aws-well-architected-framework-updated-white-papers-tools-and-best-practices/>

3. Disaster recovery whitepaper:

<https://d1.awsstatic.com/whitepapers/aws-disaster-recovery.pdf> (Archived)

.