**Identity Access Management(IAM)**

* It allows us to manage users and their level of access to the ASW console.

**What does IAM gives you?**

* Centralized control of your AWS account
* Shared Access to your AWS account
* Granular Permission: we can give the access to a specific user, specific service and specific level of access.
* Identity Federation (including Active Directory, Facebook, linkedin etc) like it allow us to login like SSO
* Multifactor Authentication
* Provides temporary access for users/devices and services where necessary.
* Allow you to set up your own password rotation policy
* Integrate with many different AWS Services.
* Supports PCI DSS compliance

**IAM Lab:**

Customize the IAM user sign link?

Service->IAM->click on 'customize' -> enter the name what you want to display.

root account:

It is a email address that we use for sing in with the AWS.

Activate MFA on your root account:

Manage MFA->A virtual MFA device->Next Step->Next->google authenticator generate the code. provide the code->next

We are activating this because if someone get our email id and password, they still not able to login to without the physical device.

**Create Individual IAM users:**

Create individual IAM users ->Manage users->Add users->provide the user name->Access tpe->set the password->Netx Permission->create group->provide the name of the group anf select the group->create group->next review->create user

Note: the "Access key ID and 'Secret access key' can only be use when we are programmatically interacting AWS. We cannot use the 'Access key id' and 'secret key ID' to login to the console

**Creating a Group:**

Group->Create a New Group->Provide the group name->Next Step->select the access->next step->create Group

**Creating Role:**

Role->Create New Role->Provide the role name->Next->Select the Role type->click on the select and then select a service->Next->Create Role

Note: The purpose of role is to allow one ASW service to interact with another AWS service.

**Summary:**

* IAM consists of the following:
  + users
  + Groups (A way to group our users and apply polices to them collwctively)
  + Roles
  + Policy Documents
* IAM is universal. It does not apply to regions at this time.
* The "root account" is simply the account created when first setup your AWS account. It has complete Admin access.
* New Users have NO permissions when first created.
* New Users are assigned "Access Key ID and Secret Access Keys" when first created.
* These are not the same as a password and you cannot use the Access key ID and Secret Key to login in to the console. You can use this to access AWS via the APIs and Command line. However, you only get to view these once. If you lose them, you have to regenerate them. So save them in a secure location.
* Always setup Multifactor Authentication on your root account.
* You can create and customize your own password rotation policies.

========================================================================================================

**Create a billing Alarm**

go to your dash board-> you name->My Billing Dashboard->Monitor your estimate charges. Enable now to begin...->select Billing Alerts-> Save Preferences

============================================================================================================

**S3 (simple Storage Service)**

* It provides developer and IT teams with secure, durable, high-scalable object storage. Amazon S3 is easy to use, with a simple web service interface to store and retrieve any amount of data from anywhere on the web.
* S3 is a safe place to store your file.
* It is Object based storage (means file like videos, pdf, photo or flat file etc).
* The data is spread across multiple devices and facilities.

**S3-Basics**

* + It is Object based i.e. allows you to upload files
  + Files can be from 0 Bytes to 5TB.
  + There is unlimited storage.
  + Files are stored in Buckets (it’s just a folder).
  + S3 is a universal namespace, that is, name must be unique globally.
  + SO when we create a bucket then basically we create a this DNS address (https://s3-ap-south-1.amazonaws.com/acloudguru)
  + When you upload a file to S3 you will receive a HTTP 200 code if the upload was successful.
  + Build for 99.99% availability for S3 platform.
  + Amazon guarantees 99.9999999999% (11x9's) durability for S3 information.
  + Tired Storage Available
  + Lifecycle Management
  + Versioning
  + Encryption
  + Secure your data using Access Control Lists and Bucket Polices.

**Data Consistency Model for S3**

* + Read after Write consistency for PUTS of new Objects
  + Eventual Consistency for overwrite PUTS and DELETES (can take some time to propagate)

**S3 - S3 is a simple key, value store**

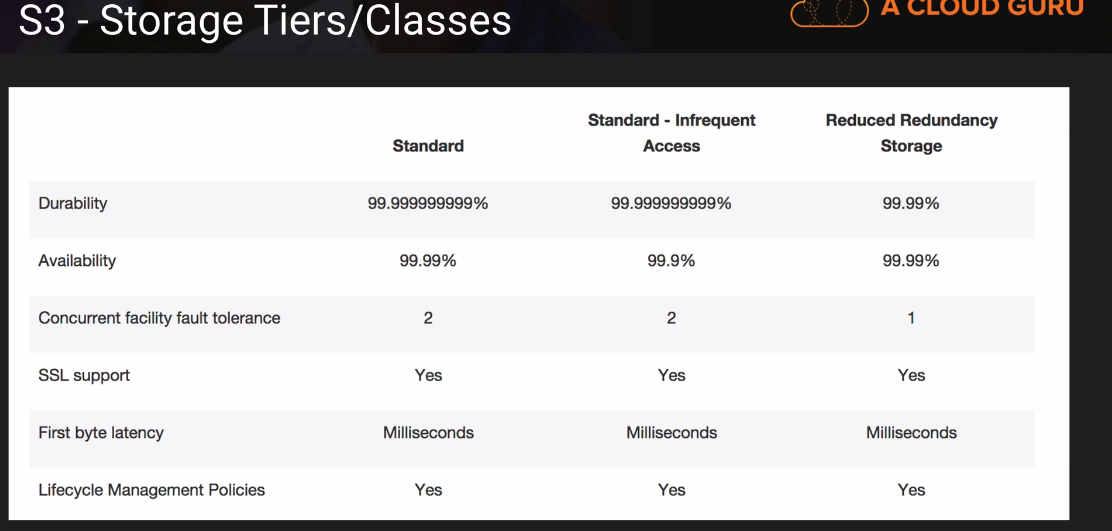
* S3 is Object based. Objects consist of the following:
  + - Key (This is simply the name of the Object)- This is simply the name of the object.
    - Value (This is simply the data and is made up of the Object) - This is simply the data and is made up of a sequence of bytes).
    - Version ID (Important for versioning)
    - Metadata (Data about the data you are storing)
    - Subresources - It consists of two things:

1. Access Control list

2. Torrent - S3 support bit torrent protocol

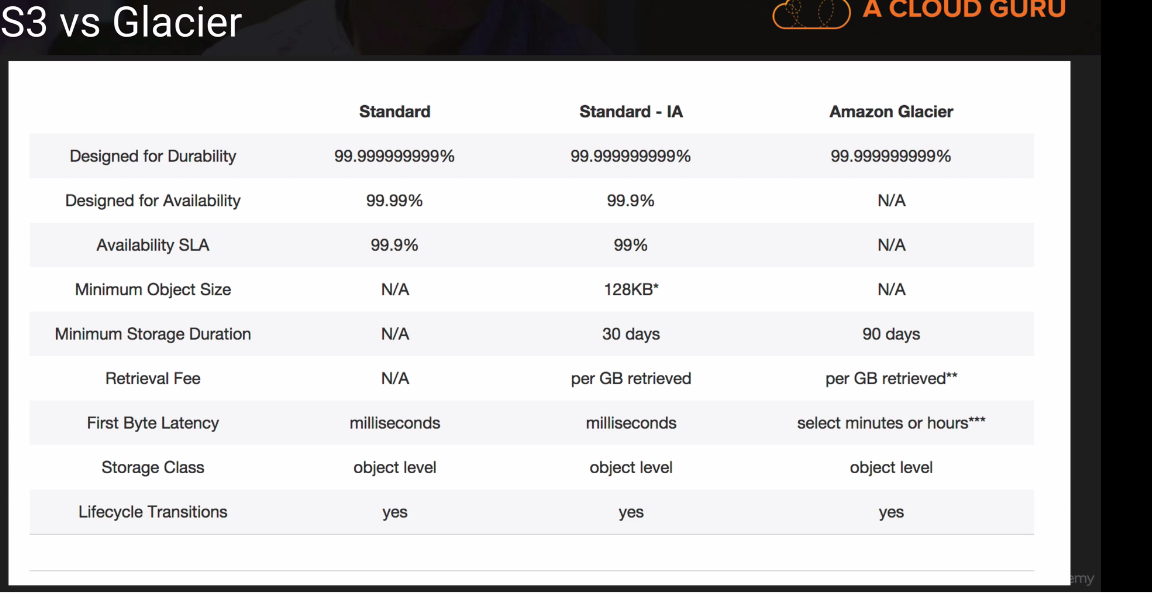
**S3 - Storage Tiers/Classes**

* + S3 - 99.99% availability, 99.(11x9's) durability, stored redundantly across multiple devices in multiple facilities and is designed to sustain the loss of 2 facilities concurrently.
  + S3- IA (Infrequently Accessed) for data that is accessed less frequently, but requires rapid access when needed. Lower fee then s3, but you are charged a retrieval fee.
  + Reduces Redundancy Storage - Designed to provide 99.99% durability and 99.99% availability of objects over given year.
  + Glacier - very Cheap, but used for archival only. It takes 3-5 hours to restored from glacier.



**What is Glacier?**

It is an extremely low-cost storage service for data archival. Amazon Glacier stores data for as little as $0.01 per gigabyte per month, and is optimized for data is infrequently accessed and for which retrieval times of 3 to 5 hours are suitable.



**S3- Charge**

Chare for:

* Storage
* Request – No. of request.
* Storage Management Pricing – Charged on per tag basis
* Data Transfer Pricing – Data coming into S3 id free but moving data around the within S3 is charged.
* Transfer Acceleration –

**What is Transfer Acceleration?**

* Amazon S3 Transfer Acceleration enables fast, easy, and ensures transfer of files over long distances between your end users and an S3 bucket.
* It takes advantage of Amazon CloudFront’s globally distributes edge locations. As the data arrives at edge locations, data is routed to Amazon S3 over an optimized network path.

**Summary:**

* + Remember that S3 is Object based i.e. allows you to upload files.
  + Files can be from 0 Bytes to 5 TB.
  + There is unlimited storage.
  + Files are stored in Buckets (folder)
  + S3 is a universal namespace, that is, name must be unique globally (<https://s3-ap-south-1.amazoneaws.com/bucket_name>)
  + Read after Write consistency for PUTS of new Objects
  + Eventual Consistency for overwrite PUTS and DELETES ( can take some time to propagate)
  + S3 – Storage Classes/Tiers
  + S3 (durable, immediately available, frequently accessed)
  + S3 – IA (durable, immediately available, frequently accessed)
  + S3 – Reduced Redundancy Storage ( data that is easily reproducible, such as thumb nails etc).
  + Glacier – Archived data, where you can wait 3 – 5 hours before accessing.
  + Remember the core fundamentals of an S3 objects:
    - * Key (name)
      * Value (data)
      * Version ID
      * Metadata
      * Subresources
        + ACL
        + Torrent
* Object based storage only (for files)
* Not suitable to install an operating system on.

Successful upload will generate a HTTP 200 status code.

**Creating an S3 Bucket –Lab**

login to console ->service->Stoge-S3->click on " Create Bucket" ->provide the name->create->

**S3- Versioning Lab**

* Stores all version of an object (including all writes and even if you delete an object)
* Great back up tool
* Once enables, versioning cannot be disables, only suspended.
* Integrate with life Lifecycle rules.
* Versioning’s MFA Delete capability, which uses multi-factor authentication, can be used to provide an additional layer of security.

**Lifecycle Management IA S3 & Glacier Lab**

* It can be used in conjunction with versioning.
* It can be applied to current version and previous version.
* Following actions can now be done:
* Transition to the Standard – Infrequent Access Storage Class (128Kb and 30 days after the creation date)
* Archive to the Glacier Storage Class ( 30 Days after IA, if relevant)
* Permanently Delete.

**CloudFront**

* + Amazon CloudFront can be used to deliver your entire website, including dynamic, static, streaming, and interactive content using a global network of edge locations. Requests for your content are automatically routed to the nearest edge locations. SO content is delivers with the best possible performance.
  + It is optimized to work with other Amazon Web Services, like Amazon Simple Storage Service (Amazon S3), Amazon Elastic Compute Cloud (Amazon EC2), Amazon Elastic Load Balancing, and Amazon Route 53. Amazon CloudFront also works seamlessly with any non-AWS origin server, which stores the original, definitive versions of your files.

**What is CDN?**

* A content delivery network (CDN) is a system of distributed servers (network) that deliver webpage and other web content to a user based on the geographic locations of the user, the origin of the webpage and a content delivery server.

**CloudFront – Key Terminology**

* **Edge Location** – This is the location where content will be cached. This is separate to an AES Region/AZ.

It is not just READ only, you can write to them too (i.e. put an object on to them).

Objects are cached for the life of the TTL (Time to live)

You can clear the cache objects, but you will be charged.

* **Origin** – This is the origin of all the files that the CDN will distribute. This can be an S3 Bucket, an EC2 Instance, an Elastic Load Balancer or Route53.
* **Distribution** – This is the name given the CDN which consists of a collection of Edge Locations.
* **Web Distribution** – Typically used for Websites.
* **RTMP** – Used for Media Streaming.

**S3 – Security and Encryption**

**Securing your bucket**

* By default, all newly created buckets are PRIVATE.
* You can setup access control to your buckets using:
* Bucket Policies
* Access Control List
* S3 buckets can be configured to create access logs which log all requests made to the S3 bucket. This can be done to another bucket.

**Encryption**

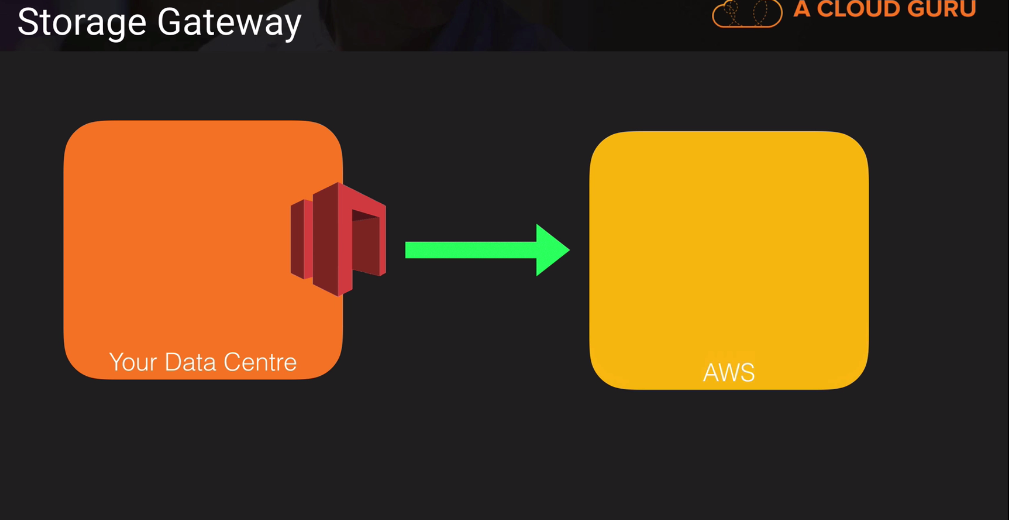
* In Transit – Sending information the PC to the bucket.

SSL/TSL

* At Rest
* Server Side Encryption
* S3 Managed Keys – SSE-S3
* AWS Key Management Service, Managed Keys – SSE-KMS
* Server Side Encryption with Customer Provided Keys – SSE-C
* Client Side Encryption

**Storage Gateway**

* AWS storage Gateway is a service that connects an on-premises software application with cloud-based storage to provide seamless and secure integration between an organization’s on-premises IT environment and ASW storage infrastructure. The Service enables you to securely store data to the AWS cloud for scalable and cost-effective storage.



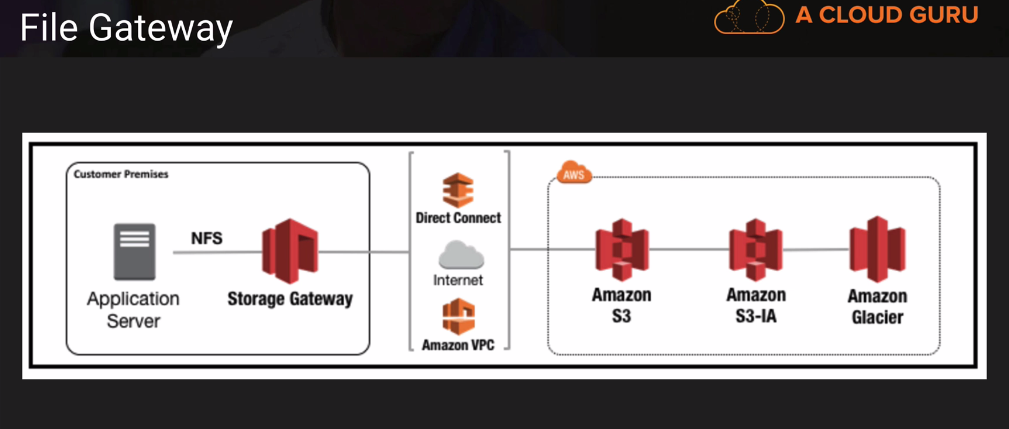
* AWS Storage Gateway’s software appliance is available for download as a virtual machine (VM) image that you install on a host in your datacenter. Storage Gateway supports either VMware ESXi or Microsoft Hyper-V. Once you’ve installed your gateway and associated it with your AWS account through the activation process, you can use the AWS Management Console to create the storage gateway option that is right for you.

**Four Types of Storage Gateway**

* File Gateway (NFS) - it store the flat file like pdf, picture, video etc. and it stores directly in the S3.
* Volume Gateway (ISCSI) – it store the block base storage like datavase, OS etc. it is divided into two types:
* Stored Volumes: it will store the entire copy of the data set on site.
* Cached Volumes: This is where we are only storing the most recent access data on premises.
* Tape Gateway (VTL) – it is a backup and archiving solutions that allow you create us to create virtual tapes and send them to S3.

**File Gateway**

* Files are stored as objects in your S3 buckets, accessed through a Network File System (NFS) mount point. Ownership, permissions, and timestamps are durably stored in S3 in the user-metadata of the object associated with the file. Once object are transferred to S3, they can be managed as native S3 objects, and bucket polices such as versioning, lifecycle management, and cross-region replication apply directly to object stored in your bucket.

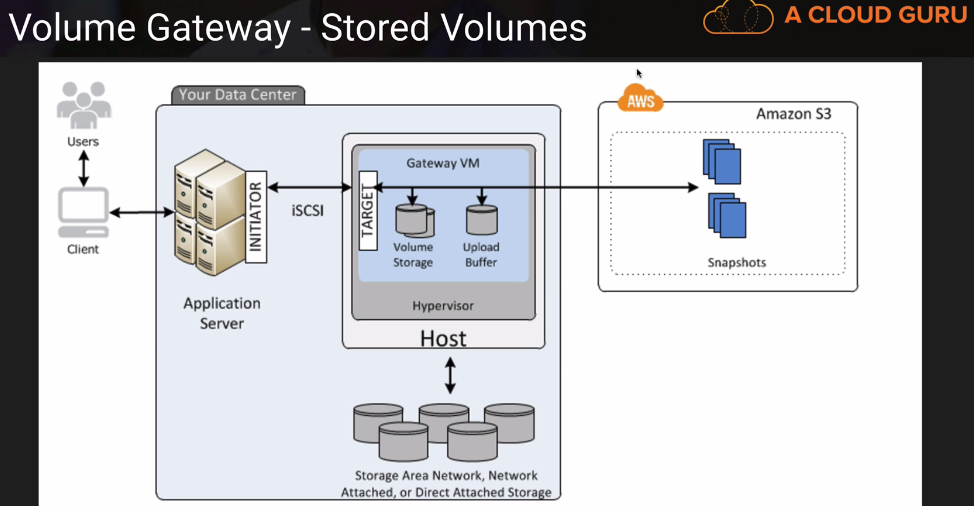


**Volume Gateway**

* The Volume interface presents your applications with disk volumes using the iSCSI block protocol.
* Data written to these volumes can be asynchronously backed up as point-in-time snapshots of your volumes, and stored in the cloud as Amazon EBS snapshots.
* Snapshots are the incremental backups that capture only changes blocks.
* All snapshots storage is also called compressed to minimize your storage charges.

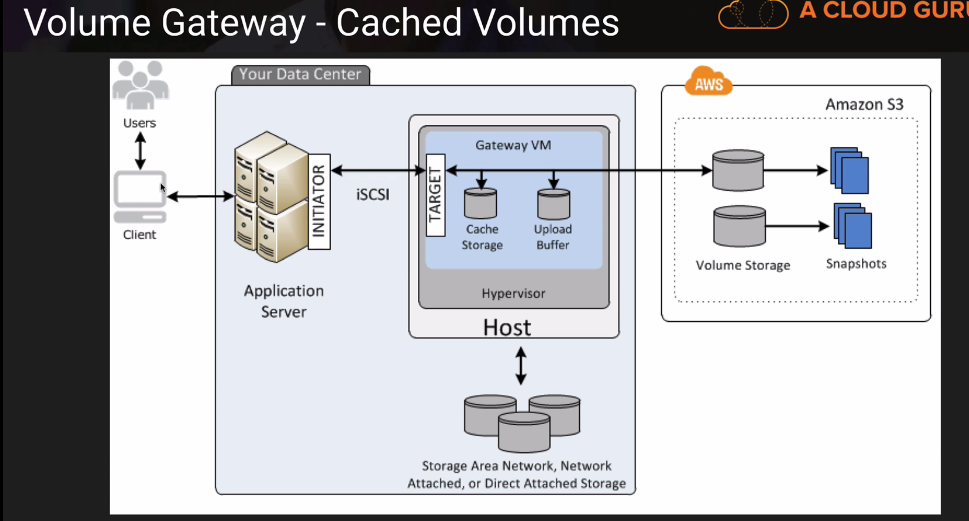
**Volume Gateway – Stored Volumes**

Stored volumes let you store your primary data locally, while asynchronously backing up data to AWS. Store volumes provide your on-premised application with low-latency access to their entire datasets, while providing durable, off-site backups. You can create storage volumes and mount them as iSCSI devices from your on-premises applications servers. Data written to your stored volumes is stored on your on-premises storage hardware. This data is asynchronously backed up to Amazon Simple Storage Service (Amazon S3) in the form of Amazon Elastic Block Store (Amazon EBS) snapshots. 1GB – 16TB in size for stored volumes.



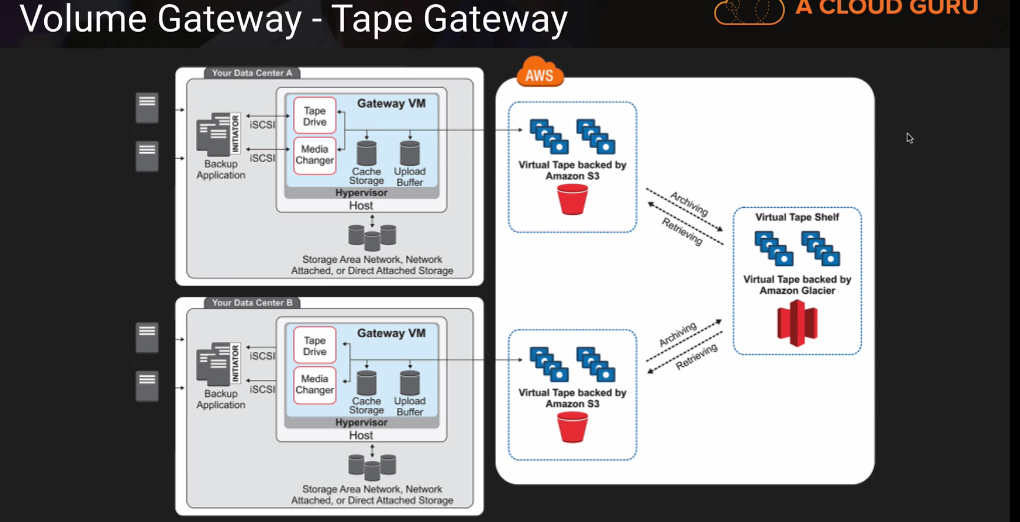
**Volume Gateway – Cached Volumes**

Cached volumes let you use Amazon Simple Storage (Amazon S3) s your primary data storage while retaining frequently accessed data locally in your storage gateway. Cached volumes minimize the need to scale your on-premises storage infrastructure, while still providing your applications with low-latency access to their frequently accessed data. You can create storage volumes up to 32 TiB in size and attached to them as iSCSI devices from your on-premises applications servers. Your gateway stored data that you write to these volumes in Amazon S3 and retain recently read data in your on-premises storage gateways’ cache and upload buffer storage. 1GB -32 TM inn size for cached Volumes.



**Volume Gateway – Tape Gateway**

Tape gateway offers a durable, cost-effective solutions to archive your date in the AWS cloud. The VTL interface it provides lets you leverage your existing tape-based backup application infrastructure to store data on virtual tape cartridges that you create on your tape gateway. Each tape gateway is preconfigured with a media changer and tape drives, which are available to your existing client backup applicants as iSCSI devices. You add tape cartridges as you need to archive your data. Supported by Netbackup, Backup Exec Veam etc.



**Summary:**

* File Gateway – For flat files, stored directly on S3

Volume Gateway

* Stored Volumes – Entire Dataset is stored on site and is asynchronously backed up to S3
* Cached Volumes – Entire Dataset is stored on S3 and the most frequently accessed data is cached on site
* Gateway Virtual Tape Library (VTL)

                Used for backup and uses popular backup applications like NetBackup, Backup Exec, Veam etc.

**Snowball**

* Import/Export Disk
* AWS Import/Export Disk accelerates moving large amount of data into and out of the AWS cloud using portable storage devices for transport. AWS Import/Export Disk transfer your data directly onto and off of storage devices using Amazon’s high-speed internal network and bypassing the internet.

**Types of Snowball**

* Snowball
* Snowball Edge
* Snowmobile

**Snowball**

* It is a petabytes-scale data transport solution that uses secure appliances to transfer large amount of data into and out if AWS. Using Snowball addresses common challenges with large-scale data transfers including high network costs, long transfer times, and security concerns. Transferring data with Snowball is simple, fast, secure, and can be as little as one-fifth the cost of high-speed Internet.
* 80TB snowball in all regions. Snowball uses multiple layers of security designed to protect your data including tamper-resistant enclosures, 256-bit encryption, and an industry-standard Trusted Platform Module (TPM) designed to ensure both security and full chain-of-custody of your data. Once the data transfer job has been processed and verified, AWS performs a software erasure of the Snowball appliance.

**Snowball Edge**

* AWS Snowball Edge is a 100TB data transfer device with on-board storage and compute capabilities. You can use Snowball Edge to move large amounts of data into and out of AWS, as a temporary storage tier for large local datasets, or to support local workloads in remote or offline locations.
* Snowball Edge connects to your existing applications and infrastructure using standard storage interfaces, streamlining the data transfer process and minimizing setup and integration. Snowball Edge can cluster together to form a local storage tier and process your data on-premises, helping ensure your applications continue to run even when they are not able to access the cloud.
* Snowmobile
* AWS Snowmobile is an Exabyte-scale data transfer service used to move extremely large amount of data to AWS. You can transfer up to 100PB per Snowmobile, a 45-foot long ruggedized shipping container, pulled by a semi-trailer truck. Snowmobile makes it easy to move massive volumes of data to the cloud, including video libraries, image repository, or even a complete data center migration. Transferring data with Snowmobile is secure, fast and cost effective.

**S3 Transfer Acceleration**

* S3 Transfer Acceleration utilize the CloudFront Edge Network to accelerate your uploads to S3.
* Instead of uploading directly to your S3 buckets, you can use a distinct URL to upload directly to an edge location which will then transfer that file to S3. You will get a distinct URL to upload to:
* Acloudguru.s3-accelerate.amazonaws.com

**Elastic Compute Cloud (EC2)**

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

**EC2 Options**

* On Demand – Allow you to pay a fixed rate by the hour with no commitment.
* Reserved – provided you with a capacity reservation, and offer a significant discount on the hourly charge for an instance. 1 Year or 3 Year Terms.
* Spot – enable you to bid whatever price you want for instance capacity, providing for even greater savings if your applications have flexible start and end times.
* Dedicated Hosts – Physical EC2 server dedicated for your use. Dedicated Hosts can help you reduce costs by allowing you to use your existing server-bond software licenses.

**On Demand**

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

**Reversed**

* Applications with steady state or predictable usages.
* Applications that requires reserved capacity
* Users able to make upfront payments to reduce their total computing costs even further.

**Spot**

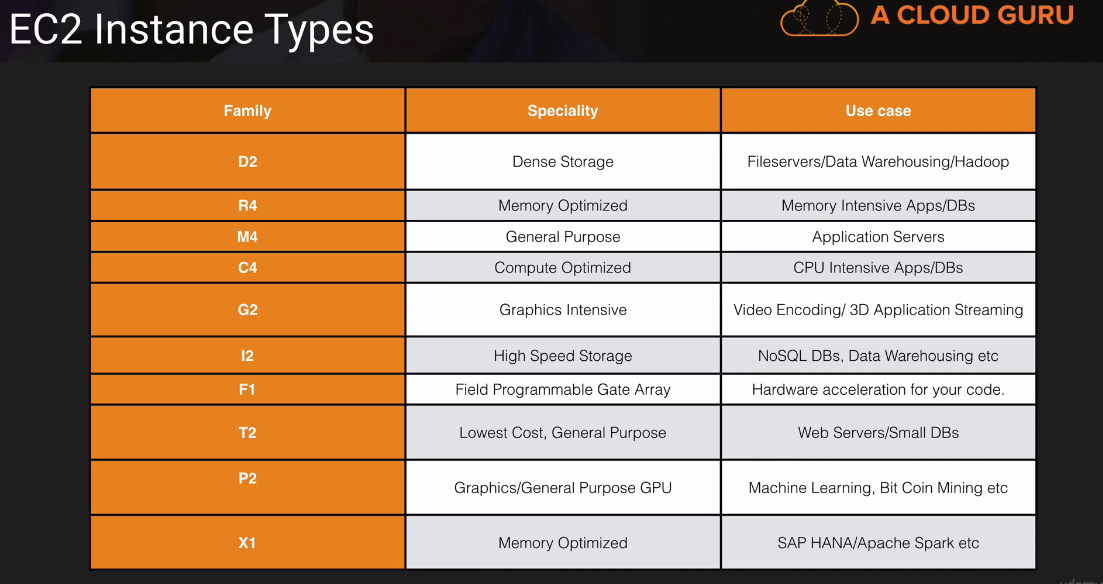
* Applications that have flexible start and end times.
* Applications those are only feasible at very low compute prices.
* Users with urgent computing needs for large amounts of additional capacity.

**Dedicated Hosts**

* Useful for regulatory requirements that may not support multitenant virtualization.
* Great for licensing which does not support multi-tenancy or cloud deployments.
* Can be punched On-Demand (hourly).
* Can be punched as Reservations for up to 70% off the On-Demand price.
* Note: if the Spot instance is terminated by Amazon EC2, you will not be charges for a partial hour of usage. However, if you terminate the instance yourself, you will be charged for any hour in which the instance ran.

**EC2 Instances Type**

* How to remember EC2 instance type:
  + DR Mc GIFT PX
    - D - Density
    - R - RAM
    - M – Main chose for general purpose apps
    - C - Compute
    - G - Graphics
    - I - IOPS
    - F - FPGA
    - T - Cheap general purpose (think T2 Micro)
    - P – Graphics (think Pics)
    - X - Extreme Memory



**What is EBS?**

Amazon EBS allows you to create storage volumes and attach them to Amazon EC2 instances. Once attached, you can create a file system on top of these volumes, run a database, or use them in any other way you would use a block device. Amazon EBS volumes are placed in a specific Availability Zone, where they are automatically replicated to protect you from the failure of a single component.

**EBS Volume Types**

* + General Purpose SSD (GP2)
    - It balances both price and performance.
    - Ratio of 3 IOPS per GB with up to 10.000 IOPS and the ability to burst up to 3000 IOPS for extended periods of time for volumes under 1Gib.
  + Provisioned IOPS SSD (IO1)
    - Designed for I/O intensive applications such as large relational or NoSQL database.
    - Use if you need more than 10,000IOPS.
    - Can provision up to 20,000 IOPS per volume.
  + Throughput Optimized HDD (ST1)
    - Big data
    - Data warehouses
    - Log processing
    - Cannot be boot volume
  + Cold HDD (SC1)
    - Lowest Cost Storage for infrequently accessed workloads
    - File Server
    - Cannot be a boot volume.
  + Magnetic (Standard)
    - Lowest cost per gigabytes of all EBS volume types that is bootable. Magnetic volumes are ideal for workloads where data is accessed infrequently and applications where the lowest storage cost is important.
* Note: You cannot mount 1 EBS volume to multiple EC2 instances, instead use EFS

**EC2 Instances**

* Summary:
  + EBS Root Volumes of your DEFAULT AMI’s cannot be encrypted. You can also use a third party tool ( such as bit locker etc ) to encrypt the root volume, or this can be done when creating AMI’s (lab to follow) in the AWS console or using the API.
  + Additional volumes can be encrypted.
  + Termination Protection is turned off by default, you must turn it on.
  + On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated.

**Volumes and Snapshots**

* Volumes exist on EBS
  + Virtual Hard disk
* Snapshots exist on S3
* Snapshots are point in time copies of volumes.
* Snapshots are incremental; this means that only the blocks that have changed since your last snapshot are moved to S3.
* If this is your first snapshot, it may take some time to create.

**RAID, Volumes & Snapshots**

* RAID – Redundant Array of Independent Disks
  + RAID 0 – Striped, No Redundancy, Good Performance.
  + RAID 1 – Mirrored Redundancy
  + RAID 5 – Good for reads, bad for writes, AWS does not recommend ever putting DAID 5’s on EBS
  + RAID 10 – Striped & Mirrored, Good Redundancy, Good Performance.

**Elastic Load Balancer**

* In Service or out Service
* Health Check
* Have their own DNS name. You are never given an IP address.

**Cloud Watch EC2**

* + Standard Monitoring = 5 Minutes
  + Detailed Monitoring = 1 minutes

**What can I do with Cloudwatch?**

* + Dashboard – Creates awesome dashboards to see what is happening with your AWS environment.
  + Alarms – Allows you to set Alarms that notify you when particular threshold are hit.
  + Events – CloudWatch Events helps you to respond to state changes in your AWS resource.
  + Logs – CloundWatch Logs helps you to aggregate, monitor, and store logs.

**Roles Lab**

* + Roles are more secure than storing your access key and secret access key on individual EC2 instances.
  + Roles are easier to manage.
  + Roles can only be assigned when that EC2 instance is being provisioned.
  + Roles are universal, you can use them in any region.
* Commands:

#aws s3 ls //list all the S3 service available

# aws s3 help //display the information like man pages. see the "Available

EC2 instance meta data

Login to instances

Run the below command to get the full list of aws variable

#curl <http://169.254.169.254/latest/meta-data/>

EC2 Placement group

A placement group is logical grouping of instances with in a single Availability Zone. Using placement groups enables applications to participate in low-latency, 10 Gbsps network. Placement groups are recommended for applications that benefits from low network latency, high network throughput, or both.

A placement group can’t span multiple Availability Zones.

The name you specify for a placement group must be unique within your AWS account.

Only certain types of instances can be launched in a placement group (Compute Optimize, GPU, Memory Optimized, Storage Optimized)

AWS recommend homogenous instances within placement group. You can create an AMI from your existing instances, and then launched a new instances from the AMI into a placement group.

DNS

SOA Records

The name of the server that supplied the data for the zone.

The administrative of the zone.

The current version of the data file.

The number of seconds a secondary name server should wait before checking for updates.

The number of seconds a secondary name server should wait before retrying a failed zone transfer.

The maximum number of seconds that a secondary that a secondary name server can use data before it must either be refreshed or expire.

The default number of seconds for the time-to-live file on resource records.

NS Records

NS stands for the Name Server and are used by Top Level Domain server to direct traffic to the Content DNS server which contains the authoritative DNS records.

A Records

An “A” records is the fundamental type of DNS record and “A” in A record stands for “Address”. The A record is used by a computer to translate the name of the domain to the IP address. For example “http://www.acloud.guru might point to the “http://123.10.10.80

TTL

The length that a DNS records is cached on either the Resolving Server or the users own local PC is equal to the value of the “Time To Live” (TTL) in seconds. The lower the time to live, the faster changes to DNS records take to propagate throughout the internet.

CNAMES

A Canonical Name (CName) can be used to resolve one domain name to another. For example, you may have a mobile website with the domain name <http://m.acloud.guru> that is used for when users browse to your domain name on their mobile devices. You may also want to the name <http://mobile.acloud.gurn> to resolve to this same address.

Alias Records

Alias records are used to map resource record sets in your hosted zone to Elastic Load Balancer, CloudFront distributions, or S3 buckets that are configured as websites.

Alias records work like a CNAME records in that you can map one DNS name ([www.example.com](http://www.example.com)) to another ‘target’ DNS name (elb1234.elb.amazonaws.com).

Key Differences – A CNAME cannot be used for naked domain names (zone apex). You cant have a CNAME for <http://alound.gurn>, it must be wither an A record or an Alias.

Alias resource record set can save you time because Amazon Route 53 automatically recognizes changes in the records sets that the alias resource record set refer to.

For example, suppose an alias resource record set for example.com points to an ELB load balancer at lb1-1234.us-east-1.elb.amazonaws.com. if the IP address of the load balancer changes, Amazon Rote 53 will automatically reflect those changes in DNS answer for example.com without any changes to the hosted zone that contains resource record set for example.com

Summary:

ELB’s do not have pre-defined IPv4 address, you resolve to them using s DNS name.

Understand the difference between an Alias record and a CNAME.

Give the choice, always choose an Alias Record over a CNAME.

**51 - Databases 101**

**Relational Database**

* + SQL Server
  + Oracle
  + MySQL Server
  + PostgreSQL
  + Aurora
  + MariaDB

**Non Relational Databases (DynamoDB)**

* + Database
    - Collection = Table
    - Document =Row
    - Key Value Pairs = Fields

**What is Data Warehousing?**

* + It is used for business intelligence. Tools like Cognos, Jaspersoft, SQL Server Reporting Services, Oracle Hyperion, SAP Net Weaver.
  + It is used to pull in very large and complex data sets. Usually used by management to do queries on data (such as current performance vs. targets etc)

**What is Elasticache?**

* It is a web service that makes it easy to deply, operate, and scale an in-memory cache in the cloud. The service improves the performance of web applications by allowing you to retrieve information from fast, managed in-memory caches, instead of relying entirely on slower disk-based databases.
* ElastiCache supports two open-source in-memory caching engines:
* Memcached
* Redis

**AWS Database Types Summary**

* RDS-OLTP(Online Transaction Processing) – it consist of below server.
  + SQL
  + MySQL
  + PostgreSQL
  + Oracle
  + Aurora – Amazon Propriety Database.
  + MariaDB
* DynamoDB – No SQL
* RedShift – OLAP (Online Analytics Processing) – it is used for data ware housing.
* Elasticache – In Memory Caching
  + Memcached
  + Redis
* DMS

**53 - Backups, Multi-AZ & Read Replicas**

**Automated Backups**

* There are two different types of Backups for AWS.
  + Automated Backups
  + Database Snapshots
* Automated Backups allow you to recover your database to any point in time within a “retention period”.
* Retention period can be between one and 35 days.
* Automated backups will take a full daily snapshot and will also store transaction logs throughout the day.
* When you do a recovery, AWS will first choose the most recent daily backup, and then apply transaction logs relevant to that day.
* This allows you to do a point in time recovery down to a second, within the retention period.
* Automated backup are enabled by default. The backup data is stored in S3 and you get free storage space equal to size of your database. SO if you have an RDS instance of 10G, you will get 10G worth of storage.
* Backup are taken within a defined window. During the backup window, Storage I/O may be suspended while your data is being backed up and you may experience elevated latency.

**Snapshots**

* DB Snapshots are done manually (i.e. they are user initiated). They are stored even after deleted the original RDS instances, unlike automated backups.

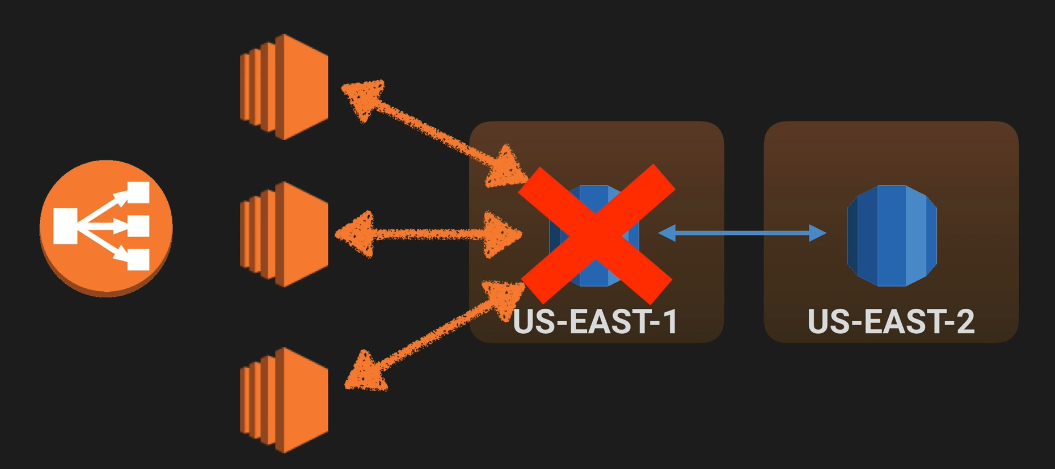
**Restoring Backups**

* Whenever you restore either an Automatic Backup or a manual Snapshot, the restored version of the databases will be a new RDS instances with a new end point.

**Encryption**

* Encryption at rest is supported for MySQL, Oracle, SQL Server, PostgreSQL & MariaDB.
* Encryption is done using the AWS key Management Service (KMS) service.
* Once your RDS instance is encrypted the data stored at rest in the underlying storage is encrypted, as are its automated backups, read replicas, and snapshots.
* At the present time, encryption an existing DB instance is not supported. To use Amazon RDS encryption for an existing databases, create a new DB instance with encryption enabled and migrate your data into it.

**What is a Multi-AZ?**



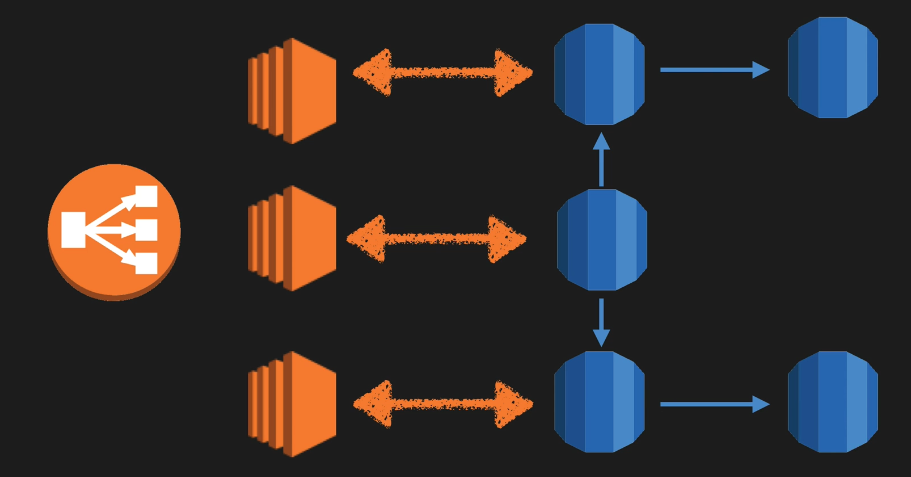
**What is Multi-AZ RDS?**

* Multi-AZ allows you to have an exact copy of your production database in another Availability Zone. AWS handles the replication for you, so when your production database is written to, this write will automatically be synchronized to the stand by database.
* In the event of planned database maintenance, DB Instances failure, or an Availability Zone failure, Amazon RDS will automatically failover to the standby so that database operation can resume quickly without administrative intervention.
* Multi-AZ for Disaster Recovery only. It is not primarily used for improving performance. For performance improvement you need Read Replicas.

**Multi-AZ Databases**

* + SQL Server
  + Oracle
  + MySQL Server
  + PostgreSQL
  + MariaDB

**What is read replicas?**



* Read replicas allow you to have a read only copy of your production database.
* This is achieved by using Asynchronous replication from the primary RDS instance to the read replica.
* You use read replicas primarily for very read-heavy database workloads.

**Read Replica Database**

* Supported Read replicas:
  + MySQL server
  + PostgreSQL
  + MariaDB
* Used for Scaling!!! Not for DR!
* Must have automatic backups turned on in order to deploy a read replicas.
* You can have up to 5 read replicas copies of any databases
* You can have read replicas of read replicas(but watch out for latency)
* Each read replicas will have its own DNS end point.
* You cannot have Read Replicas that have Multi-AZ
* You can create Read Replicas of Multi-AZ source database however.
* Read Replicas can be promoted to be their own databases. This breaks the replication.
* Read Replicas can be promoted to ve their own databases. This breaks the replication.
* Read Replica in a second region for MySQL and MariaDB. Not for PostgreSQL.

**DynamoDb vs RDS**

* DynamoDB offers “push button” scaling, meaning that you can scale your database on the fly, without any down time.
* RDS is not so easy and you usually have to use a bigger instance size or to add a read replica.

54 – DynamoDB

What is DynamoDB?

Amazon DynamoDB is a fast and flexible NoSQL database service for all application that need consistent, single-digit millisecond latency at any scale. It is fully managed database and supports both document and key-value data models. Its flexible data model and reliable performance make it a great fit for mobile, web, gaming, ad-tech, IoT, and many other applications.

Stored on SSD storage

Spread across 3 geographically distinct data centers

Eventual Consistent Reads (Default)

Strongly Consistent Reads

Eventual Consistent Reads

Consistency across all copies of data is usually reached within a second. Repeating a read after a short time should return the update data. (Best Read Performance)

Strongly Consistent Reads

A strongly consistent read returns a result that reflects all writes that received a successful response prior to the read.

DynamoDB Pricing

Provisioned Throughput Capacity

Write Throughput $0.0065 per hour for every 10 Units

Read Throughput $0.0065 per hour for every 50 Units

Storage costs of $0.25Gb per month.

55 – RedShift

What is Redshift?

Amazon Redshift is a fast and powerful, fully managed, petabyte-scale data warehouse service in the cloud. Customers can start small for just $0.25 per hour with no commitments or upfront costs and scale to a petabyte or more for $1000 per terabyte per year, less than a tenth of most other data warehousing solutions.

Redshift Configuration

Single Node (160Gb)

Multi-Node

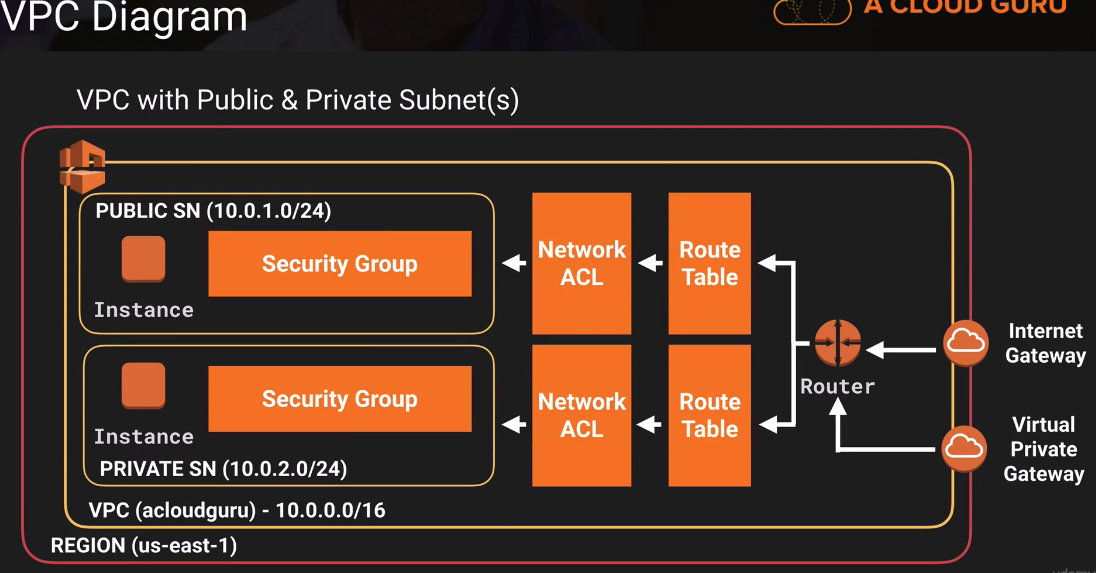
Leader node (manages client connections and receives queries)

Compute Node (store data and perform queries and computations). Up to 128 Compute Nodes.

Columnar Data Storage: Instead of storing data as a series of rows, Amazon Redshift organizes the data by column. Unlike row-based system are ideal for data warehousing and analytics, where queries often involve aggregates performed over large data sets. Since only the column involved in the queries are processed and columnar data is stored sequentially on the storage media, column-based systems require far fewer I/Os, greatly improving query performance.

**VPC Overview**

* It’s a logical data centre.
* Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the Amazon Web Service (AWS) Cloud where you can launch AWS resource in a virtual network that you define. You have completed control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.
* You can easily customize the network configuration for your Amazon Virtual Private Cloud. For example, you can create a public-facing subnet for your web servers that have the access to the internet, and place your backend systems such as databases or application server in a private-facing subnet with no internet access. You can leverage multiple layers of security, including security groups and network access control lists, to help control access to Amazon EC2 instances in each subnet.
* Additionally, you can create a Hardware Virtual Private Network (VPN) connection between your corporate datacenter and your VPC and leverage the AWS cloud as an extension of your corporate datacenter.



**What can you do with a VPC?**

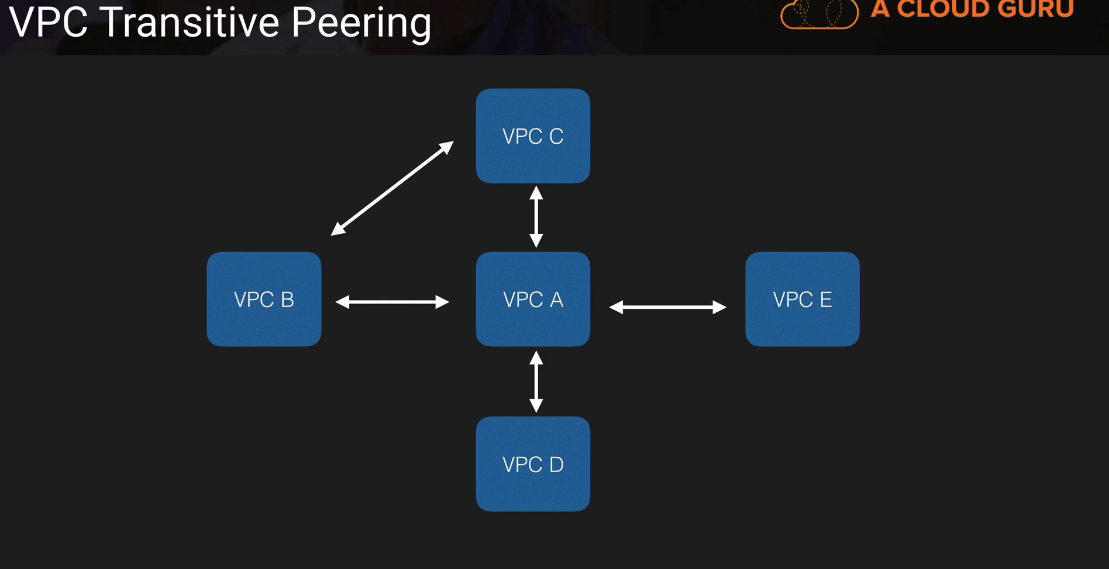
* Launch instances into a subnet of your choosing
  + Assign Custom IP address ranges in each subnet.
  + Configure route tables between subnets.
  + Create internet gateway and attach it to our VPC
  + Much better security control over your AWS resources
  + Instances security groups.
  + Subnet network access control lists (ACLS)

**Default VPC vs. Custom VPC**

* + Default VPC is user friendly, allowing you to immediately deploy instances.
  + All Subnets in default VPC have a route out to the internet
  + Each EC2 instances has both a public and private IP address.
  + If you delete the default VPC the only way to get it back is to Contact AWS.

**VPC Peering**

* + Allow you to connect one VPC with another via a direct network route using private IP address.
  + Instances behave as if they were on the same private network.
  + You can peer VPC’s with other AWS accounts as well as with other VPCs in the same account.
  + Peering is in a star configuration, ie 1 central VPC peers with 4 others. NO TRANSITIVE PEERING!!



**Summary:**

* + Think of a VPC as logical datacenter in AWS
  + Consists of IGW’s ( Or Virtual Private Gateways), Route Tables, Network Access Control lists, Subnets, Security Groups.
  + 1 Subnet = 1 Availability Zone
  + Security Groups are Stateful, Network Access Control Lists are Stateless
  + NO TRANSITIVE PEERI NG

**Build your own custom VPC**

* When we create VPC, then by default, it will create “Route table”, “Network ACL” and “security Group”

Steps for creating the vpc

* Create VPC
* Create subnet (1 subnet = 1 Availability)
* First 4 and last IP address is not available to use in subnet.
* Create a Internet Gateway
  + - * This is use for providing the public access.
      * 1VPC = 1 Internet Gateway
* Create a new route table to provide the access to the internet
* Associate the subnet with it
* Auto assign the public IP

**Network Address Translation (NAT)**

* + There are 2 ways to provide the public access to the private instances.
    - NAT instance
    - NAT Gateway (Preferred way)

**NAT instances**

* + When creating a NAT instance, Disable Source/Destination check on the instance.
  + NAT instance must be in a public subnet
  + There must be a route out of the private subnet to the NAT instance, in order for this to work
  + The amount of traffic that NAT instances supports depends on the instance size. IF you are bottlenecking, increase the instance size
  + You can create high availability using Autoscaling Groups, multiple subnets in different AZ’s and a script to automate failover.
  + NAT instance always behind a security groups.

**NAT Gateway**

* + Scale automatically up to 10Gbps
  + Automatically assign a public ip address.
  + Not associated with security groups
  + Remember to update your route tables
  + No Need to disable Source/Destination checkes.

**Network Access Control Lists vs Security Groups**

* + Your VPC automatically comes a default network ACL and by default it allows all outbound and inbound traffic.
  + You can create a custom network ACL. By default, each custom network ACL denies all inbound and outbound traffic until you add rules.
  + Each subnet in your VPC must be associated with a network ACL. If you do not explicitly associate a subnet with a network ACL, the subnet is automatically associated with the default network ACL.
  + You can associate a network ACL with multiple subnets; however, a subnet can be associated with only one network ACL at a time. When you associate a network ACL with a subnet, the previous associated is removed.
  + A network ACL contains a numbered list of rules that is evaluated in order, starting with the lowest numbered rule.
  + A network ACL has separated inbound and outbound rules, and each rule can either allow or deny traffic.
  + Network ACLs are stateless; responses to allow inbound traffic are subject to the rules for outbound traffic (and vice versa)
  + Block IP Address using network ACL’s not security Groups

**65 - NATs vs Bastions**

* A NAT id used to provide internet traffic to EC2 instances in private subnets.
* A Bastion is used to securely administer EC2 instances in private subnets. We also call it jump host.

**67 - VPC Clean Up**

Delete all the instances running under the VPC

Then deleted the VPC

**68 – Summary**

**69 – SQS**

* + Amazon SQS is a web service that gives you to access to a message queue that can be used to store messages while waiting for a computer to process them.
  + Amazon SQS is a distributed queue system that enables web service application to quickly and reliably queue message that one component in the application generates to be consumed by another component. A queue is a temporary repository for message that is waiting processing.
  + Using Amazon SQS, you can decouple the components of an application so they run independently, with Amazon SQS easing message management between components. Any component of a distributed application can store up to 256 KB of text in any format. Any component can later retrieve the message programmatically using the Amazon SQS API.
  + The queue acts as a buffer between the component producing and saving data, and the component receiving the data for processing.
  + Amazon SQS ensures delivery of each meaage at least once, and supports multiple readers and writers interacting with the same queue
  + Does not offer FIIFO
  + 12 hours visibility time out
  + 256kb message size now available
  + Billed at 64Kb “Chunks”
  + A 256kb message will be 4 x 64kb “Chunks”
  + First 1 million Amazon SQS Requests per month are free.
  + $0.50 per 1 million Amazon SQS requests per month thereafter.
  + A Single requests can have from 1 to 10 message, up to a maximum total payload 256KB
  + Each 64KB ‘chunk’ of payload is billed as 1 request. For example, a single API call with a 256KB payload will be billed as four requests.

70 – SWF