**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.**

## Conclusion

While the number and types of services offered by AWS have increased dramatically, our philosophy on pricing has not changed. You pay as you go, pay for what you use, pay less as you use more, and pay even less when you reserve capacity. Projecting costs for a use case, such as web application hosting, can be challenging, because a solution typically uses multiple features across multiple AWS products, which in turn means there are more factors and purchase options to consider.

# EC2 Pricing Models

Lecture 24

1

## On Demand

Allows you to pay a fixed rate by the hour (or by the second) with no commitment.

2

## Reserved

Provides you with a capacity reservation, and offer a significant discount on the hourly charge for an instance. Contract Terms are 1 Year or 3 Year Terms.

3

## Spot

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

4

## Dedicated Hosts

Physical EC2 server dedicated for your use. Dedicated Hosts can help you reduce costs by allowing you to use your existing server-bound software licenses.

04:23 / 14:26 Transcript Browse Q&A Add B

## On Demand pricing is useful for;

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

## Reserved pricing is useful for;

- Applications with steady state or predictable usage
- Applications that require reserved capacity
- Users able to make upfront payments to reduce their total computing costs even further

## Reserved Pricing Types

1. **Standard Reserved instances**
   These offer up to 75% off on demand instances. The more you pay up front and the longer the contract, the greater the discount.
2. **Convertible Reserved Instances**
   These offer up to 54% off on demand capability to change the attributes of the RI as long as the exchange results in the creation of Reserved Instances of equal or greater value.
3. **Scheduled Reserved Instances**
   These are available to launch within the time windows you reserve. This option allows you to match your capacity reservation to a predictable recurring schedule that only requires a fraction of a day, a week, or a month.

## Spot pricing is useful for;

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

## Dedicated Hosts pricing is useful for;

- Useful for regulatory requirements that may not support multi-tenant virtualization.
- Great for licensing which does not support multi-tenancy or cloud deployments.
- Can be purchased On-Demand (hourly.)
- Can be purchased as a Reservation for up to 70% off the On-Demand price.

## EC2 Instance Types

| Family | Speciality | Use case |
|---|---|---|
| F1 | Field Programmable Gate Array | Genomics research, financial analytics, real-time video processing, big data etc |
| I3 | High Speed Storage | NoSQL DBs, Data Warehousing etc |
| G3 | Graphics Intensive | Video Encoding/ 3D Application Streaming |
| H1 | High Disk Throughput | MapReduce-based workloads, distributed file systems such as HDFS and MapR-FS |
| T3 | Lowest Cost, General Purpose | Web Servers/Small DBs |
| D2 | Dense Storage | Fileservers/Data Warehousing/Hadoop |
| R5 | Memory Optimized | Memory Intensive Apps/DBs |
| M5 | General Purpose | Application Servers |
| C5 | Compute Optimized | CPU Intensive Apps/DBs |
| P3 | Graphics/General Purpose GPU | Machine Learning, Bit Coin Mining etc |
| X1 | Memory Optimized | SAP HANA/Apache Spark etc |
| Z1D | High compute capacity and a high memory footprint. | Ideal for electronic design automation (EDA) and certain relational database workloads with high per-core licensing costs. |
| A1 | Arm-based workloads | Scale-out workloads such as web servers |
| U-6tb1 | Bare Metal | Bare metal capabilities that eliminate virtualization overhead |

## EC2 Instance Types - Mnemonic

ecture 24

- **F** - For FPGA
- **I** - For IOPS
- **G** - Graphics
- **H** - High Disk Throughput
- **T** - Cheap general purpose (think T2 Micro)
- **D** - For Density
- **R** - For RAM
- **M** - Main choice for general purpose apps
- **C** - For Compute
- **P** - Graphics (think Pics)
- **X** - Extreme Memory
- **Z** - Extreme Memory AND CPU
- **A** - Arm-based workloads
- **U** - Bare Metal

Important question above one

While launching instance, chose protect against accidental termination.

Providing additional tags like name of the server, department, who created would help us track the instance later on.

After login to instance yum update -y to update os based updates.

To install apache web server, use

Yum install httpd -y

Cd /var/www/html and create index.html here

Service httpd start => starts the server

Chkconfig on => will ensure if machine reboots this would start automatically

The Chkconfig command tool allows to configure services start and stop automatically in the /etc/rd.d/init.d scripts through command line.

- Termination Protection is **turned off** by default, you must turn it on.
- On an EBS-backed instance, the **default action is for the root EBS volume to be deleted** when the instance is terminated.
- EBS Root Volumes of your DEFAULT AMI's cannot be encrypted. You can also use a 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.

The information about instance can be obtained as below

Monitoring tab shows various graphs depicting cpu and memory usage etc

Tags section shows tags associated with this instance.

Security groups:

::/0 is for IPV6 addresses. 0.0.0.0/0 is for IPV4 addresses.

Any security group change is immediately effective. This is exam question

Security groups are stateful. So, when we create inbound rule to allow por 80 or 8080, it automatically allows outbound as well.

Network Access control Lists are stateless. So, we have to explicitly mention inbound and outbound rules separately.

We can assign more than one security group to an EC instance.

Actions -> Networking -> Change security groups

- Security Groups are **STATEFUL.**
- If you create an inbound rule allowing traffic in, that traffic is automatically allowed back out again.
- You cannot block specific IP addresses using Security Groups, instead use Network Access Control Lists.
- You can specify allow rules, but not deny rules.

By default everything is denied. We should allow with explicit rules.

If we want to specific deny rules, it is only possible with NACL

**Amazon Elastic Block Store (EBS) provides persistent block storage volumes for use with Amazon EC2instances in the AWS Cloud. Each Amazon EBS volume is automatically replicated within its Availability Zone to protect you from component failure, offering high availability and durability.**

## 5 Different Types of EBS Storage;

- General Purpose (SSD)
- Provisioned IOPS (SSD)
- Throughput Optimised Hard Disk Drive
- Cold Hard Disk Drive
- Magnetic

IOPs: Input output processing speed

| | Solid-State Drives (SSD) | | Hard disk Drives (HDD) | | |
|---|---|---|---|---|---|
| Volume Type | **General Purpose SSD** | **Provisioned IOPS SSD** | **Throughput Optimized HDD** | **Cold HDD** | **EBS Magnetic** |
| Description | General purpose SSD volume that balances price and performance for a wide variety of transactional workloads | Highest-performance SSD volume designed for mission-critical applications | Low cost HDD volume designed for frequently accessed, throughput-intensive workloads | Lowest cost HDD volume designed for less frequently accessed workloads | Previous generation HDD |
| Use Cases | **Most Work Loads** | **Databases** | **Big Data & Data Wharehouses** | **File Servers** | **Workloads where data is infrequently accessed** |
| API Name | gp2 | io1 | st1 | sc1 | Standard |
| Volume Size | 1 GiB - 16 TiB | 4 GiB - 16 TiB | 500 GiB - 16 TiB | 500 GiB - 16 TiB | 1 GiB-1 TiB |
| Max. IOPS**/ Volume | 16,000 | 64,000 | 500 | 250 | 40-200 |

API names are important for exam.

The EC2 instance and EBS volume will be in the same AZ. We can verify this by going to menu options of EC2 and EBS. Important question.

When we terminate EC2 instance, the EBS volume also gets deleted automatically.

We can add additional volumes to EC2 instance and also we can move from one type to another.

We can take a snapshot of existing volume, create an image out of it. And then launch new EC2 instances out of it. This way we can create a copy of existing data to another.

We can also copy image to another region and then launch instances in that region. This is a popular exam question. This needs to be tested. This helps in migrating data to different machine easily.

When we delete EC2 instance, the root volume gets deleted but all other volumes will remain intact. This is also important exam question.

- Volumes exist on EBS. Think of EBS as a virtual hard disk
- Snapshots exist on S3. Think of snapshots as a photograph of the disk.
- 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.

- To create a snapshot for Amazon EBS volumes that serve as root devices, you should stop the instance before taking the snapshot.
- However you can take a snap while the instance is running.
- You can create AMI's from both Volumes and Snapshots
- You can change EBS volume sizes on the fly, including changing the size and storage type.
- Volumes will ALWAYS be in the same availability zone as the EC2 instance.

- To move an EC2 volume from one AZ to another, take a snapshot of it, create an AMI from the snapshot and then use the AMI to launch the EC2 instance in a new AZ.
- To move an EC2 volume from one region to another, take a snapshot of it, create an AMI from the snapshot and then copy the AMI from one region to the other. Then use the copied AMI to launch the new EC2 instance in the new region.

## You can select your AMI based on:

- Region (see Regions and Availability Zones)
- Operating system
- Architecture (32-bit or 64-bit)
- Launch Permissions
- Storage for the Root Device (Root Device Volume
  - Instance Store (**EPHEMERAL STORAGE**)
  - EBS Backed Volumes

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

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

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

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

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

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

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

Ephemeral – temporary

We can choose instance store volume AMI from community AMIs

When you launch an instance, the **root** device **volume** contains the image used to boot the instance. When we introduced Amazon **EC2**, all AMIs were backed by Amazon **EC2** instance store, which means the **root** device for an instance launched from the AMI is an instance store **volume** created from a template stored in Amazon S3.

Some Amazon Elastic Compute Cloud (Amazon EC2) instance types come with a form of directly attached, block-device storage known as the instance store. The instance store is ideal for temporary storage, because the data stored in instance store volumes is not persistent through instance stops, terminations, or hardware failures.

For data you want to retain longer, or if you want to encrypt the data, use Amazon Elastic Block Store (Amazon EBS) volumes instead. EBS volumes preserve their data through instance stops and terminations, can be easily backed up with EBS snapshots, can be removed from one instance and reattached to another, and support full-volume encryption.

Cloudwatch:

**CloudWatch can monitor things like**

- Compute
  - EC2 Instances
  - Autoscaling Groups
  - Elastic Load Balancers
  - Route53 Health Checks
- Storage & Content Delivery
  - EBS Volumes
  - Storage Gateways
  - CloudFront

**Host Level Metrics Consist of:**

- CPU
- Network
- Disk
- Status Check

**AWS CloudTrail increases visibility into your user and resource activity by recording AWS Management Console actions and API calls. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred.**

- CloudWatch monitors performance.
- CloudTrail monitors API calls in the AWS platform.

**Remember;**

- CloudWatch is used for monitoring performance.
- CloudWatch can monitor most of AWS as well as your applications that run on AWS.
- CloudWatch with EC2 will monitor events every 5 minutes by default.
- You can have 1 minute intervals by turning on detailed monitoring.
- You can create CloudWatch alarms which trigger notifications.
- CloudWatch is all about performance. CloudTrail is all about auditing.

**What Can I do With CloudWatch?**

- Dashboards - Creates awesome dashboards to see what is happening with your AWS environment.
- Alarms - Allows you to set Alarms that notify you when particular thresholds are hit.
- Events - CloudWatch Events helps you to respond to state changes in your AWS resources.
- Logs - CloudWatch Logs helps you to aggregate, monitor, and store logs.

97 people bookmarked this moment.

We have to create monitoring alert in cloudwatch.

For logs, an agent should be installed on eC2 instances

For events, the target could be queue etc

Accessing AWS using CLI. While creating user, enable access key based authentication.

Then after we login to EC2 instance, use aws configure command.

This will ask access key id and secret key and region.

Then we should be able to issue commands like aws s3 ls

This will give us the list of buckets.

Similarly we can create buckets using aws s3 mb s3://bucketname

Here aws is command line tool and service name.

```
[root@ip-172-31-25-137 ec2-user]# aws configure
AWS Access Key ID [None]: AKIAIFS3YQEAYE5ROILA
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]:
[root@ip-172-31-25-137 ec2-user]# aws s3 ls
2019-01-07 14:40:13 acloudguru-2019-ryan-kroonenburg3141
2019-01-08 10:11:21 acloudguruversioning2019rjk
2019-01-08 12:05:29 crossregionreplication542322423
[root@ip-172-31-25-137 ec2-user]# aws s3 mb s3://testbucket
```

- You can interact with AWS from anywhere in the world just by using the command line (CLI).
- You will need to set up access in IAM
- Commands themselves are not in the exam, but some basic commands will be useful to know for real life.

If we go to home directory of the ec2 machine, and goto hidden directory .aws, open credentials file, we will see the access id and secret key are stored over here. This is a security risk. So, recommendation is to use roles.

- Roles are more secure than storing your access key and secret access key on individual EC2 instances.
- Roles are easier to manage.
- Roles can be assigned to an EC2 instance after it is created using both the console & command line.
- Roles are universal — you can use them in any region.

Roles can be created and assigned to EC2 instances using following menu item.

We can provide startup script for EC2 instance. These are called bootstrap scripts.

From EC2 instance, we can copy the files to s3 bucket using aws s3 cp command.

```
#!/bin/bash
yum update -y
yum install httpd -y
service httpd start
chkconfig httpd on
cd /var/www/html
echo "<html><h1>Hello Cloud Gurus Welcome To My Webpage</h1></html>" > index.html
aws s3 mb s3://YOURBUCKETNAMEHERE
aws s3 cp index.html s3://YOURBUCKETNAMEHERE
```

instance data:

we can look at the user data

curl http://....ip../latest/user-data

we can also look at the instance data

curl http://,..../latest/instance-data

```
instance-action
instance-id
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
```

curl http://169.254.169.254/latest/meta-data/local-ipv4

- Used to get information about an instance (such as public ip)
- curl http://169.254.169.254/latest/meta-data/
- curl http://169.254.169.254/latest/user-data/

**Amazon Elastic File System (Amazon EFS) is a file storage service for Amazon Elastic Compute Cloud (Amazon EC2) instances. Amazon EFS is easy to use and provides a simple interface that allows you to create and configure file systems quickly and easily. With Amazon EFS, storage capacity is elastic, growing and shrinking automatically as you add and remove files, so your applications have the storage they need, when they need it.**

Multiple EC2 instances can share EFS but not EBS

We need not provision storage space up front like EBS. It can grow and shrink automatically.

We can find EFS under storage option.

WE can update an html file in EFS so that all instances can get the latest content of it

We first create EFS. And then using command line utility, we mount this EFS into a directory in EC2.

- Supports the Network File System version 4 (NFSv4) protocol
- You only pay for the storage you use (no pre-provisioning required.)
- Can scale up to the petabytes
- Can support thousands of concurrent NFS connections
- Data is stored across multiple AZ's within a region
- Read After Write Consistency

A cluster placement group is a grouping of instances within a single Availability Zone. Placement groups are recommended for applications that need low network latency, high network throughput, or both.

Only certain instances can be launched in to a Clustered Placement Group.

## Two Types of Placement Groups;

- Clustered Placement Group
- Spread Placement Group

A cluster placement group is a grouping of instances within a single Availability Zone. Placement groups are recommended for applications that need low network latency, high network throughput, or both.

Only certain instances can be launched in to a Clustered Placement Group.

A spread placement group is a group of instances that are each placed on distinct underlying hardware.
Spread placement groups are recommended for applications that have a small number of critical instances that should be kept separate from each other.

- A clustered placement group can't span multiple Availability Zones.
- A spread placement group can.
- 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 Optimized, GPU, Memory Optimized, Storage Optimized)
- AWS recommend homogenous instances within placement groups.
- You can't merge placement groups.
- You can't move an existing instance into a placement group. You can create an AMI from your existing instance, and then launch a new instance from the AMI into a placement group.

If a spot instance is terminated by amazon, we will not be charged for partial hour.

- All Inbound traffic is blocked by default.
- All Outbound traffic is allowed.
- Changes to Security Groups take effect immediately.
- You can have any number of EC2 instances within a security group.
- You can have multiple security groups attached to EC2 Instances.

- Security Groups are **STATEFUL.**
- If you create an inbound rule allowing traffic in, that traffic is automatically allowed back out again.
- You cannot block specific IP addresses using Security Groups, instead use Network Access Control Lists.
- You can specify allow rules, but not deny rules.

- Volumes exist on EBS. Think of EBS as a virtual hard disk
- Snapshots exist on S3. Think of snapshots as a photograph of the disk.
- 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.

- To create a snapshot for Amazon EBS volumes that serve as root devices, you should stop the instance before taking the snapshot.
- However you can take a snap while the instance is running.
- You can create AMI's from both Volumes and Snapshots
- You can change EBS volume sizes on the fly, including changing the size and storage type.
- Volumes will ALWAYS be in the same availability zone as the EC2 instance.

- To move an EC2 volume from one AZ to another, take a snapshot of it, create an AMI from the snapshot and then use the AMI to launch the EC2 instance in a new AZ.
- To move an EC2 volume from one region to another, take a snapshot of it, create an AMI from the snapshot and then copy the AMI from one region to the other. Then use the copied AMI to launch the new EC2 instance in the new region.

- Snapshots of encrypted volumes are encrypted automatically.
- Volumes restored from encrypted snapshots are encrypted automatically.
- You can share snapshots, but only if they are unencrypted.
- These snapshots can be shared with other AWS accounts or made public.

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

31 people bookmarked this moment.

- Create a Snapshot of the unencrypted root device volume
- Create a copy of the Snapshot and select the encrypt option
- Create an AMI from the encrypted Snapshot
- Use that AMI to launch new encrypted instances

**Remember;**

- CloudWatch is used for monitoring performance.
- CloudWatch can monitor most of AWS as well as your applications that run on AWS.
- CloudWatch with EC2 will monitor events every 5 minutes by default.
- You can have 1 minute intervals by turning on detailed monitoring.
- You can create CloudWatch alarms which trigger notifications.
- CloudWatch is all about performance. CloudTrail is all about auditing.

## What Can I do With CloudWatch?

- Dashboards - Creates awesome dashboards to see what is happening with your AWS environment.
- Alarms - Allows you to set Alarms that notify you when particular thresholds are hit.
- Events - CloudWatch Events helps you to respond to state changes in your AWS resources.
- Logs - CloudWatch Logs helps you to aggregate, monitor, and store logs.

- You can interact with AWS from anywhere in the world just by using the command line (CLI).
- You will need to set up access in IAM
- Commands themselves are not in the exam, but some basic commands will be useful to know for real life.

- Roles are more secure than storing your access key and secret access key on individual EC2 instances.
- Roles are easier to manage.
- Roles can be assigned to an EC2 instance after it is created using both the console & command line.
- Roles are universal — you can use them in any region.

- Bootstrap scripts run when an EC2 instance first boots.
- Can be a powerful way of automating software installs and updates.

- Used to get information about an instance (such as public ip)
- curl http://169.254.169.254/latest/meta-data/
- curl http://169.254.169.254/latest/user-data/

- Supports the Network File System version 4 (NFSv4) protocol
- You only pay for the storage you use (no pre-provisioning required.)
- Can scale up to the petabytes
- Can support thousands of concurrent NFS connections
- Data is stored across multiple AZ's within a region
- Read After Write Consistency

- A clustered placement group can't span multiple Availability Zones.
- A spread placement group can.
- 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 Optimized, GPU, Memory Optimized, Storage Optimized)
- AWS recommend homogenous instances within placement groups.
- You can't merge placement groups.
- You can't move an existing instance into a placement group. You can create an AMI from your existing instance, and then launch a new instance from the AMI into a placement group.

Databases:

RDS: Relational Database Services

**Relational databases on AWS;**

- SQL Server
- Oracle
- MySQL Server
- PostgreSQL
- Aurora
- MariaDB

## RDS has two key features;

- Multi-AZ - For Disaster Recovery
- Read Replicas - For Performance

Failover is automatic between AZs

We can have 5 read replicas

## Non Relational Databases are as follows;

- Collection = Table
- Document = Row
- Key Value Pairs = Fields

**Used for business intelligence. Tools like Cognos, Jaspersoft, SQL Server Reporting Services, Oracle Hyperion, SAP NetWeaver.**

**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)**

Data Warehousing databases use different type of architecture both from a database perspective and infrastructure layer.

Amazon's Data Warehouse Solution Is Called Redshift

ElastiCache is a web service that makes it easy to deploy, operate, and scale an in-memory cache in the cloud. The service improves the performance of web applications by allowing you to retrieve 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:

ElastiCache supports two open-source in-memory caching engines:

- Memcached
- Redis

MEMCACHED

redis

## RDS (OLTP)

- SQL
- MySQL
- PostgreSQL
- Oracle
- Aurora
- MariaDB

## DynamoDB (No SQL)

## Red Shift OLAP

## Redshift for Business Intelligence or Data Warehousing

**Remember the following points;**

- RDS runs on virtual machines
- You cannot log in to these operating systems however.
- Patching of the RDS Operating System and DB is Amazon's responsibility
- RDS is NOT Serverless
- Aurora Serverless IS Serverless

We typically create an RDS instance and provide its connection string, user and password in the EC2 instance. We should enable the security group of the EC2 instance in the RDS inbound rules.

There are two different types of Backups for RDS:

- Automated Backups
- Database Snapshots

Automated Backups allow you to recover your database to any point in time within a "retention period". The 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 back up, 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 Backups are enabled by default. The backup data is stored in S3 and you get free storage space equal to the size of your database. So if you have an RDS instance of 10Gb, you will get 10Gb worth of storage.

Backups 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.

DB Snapshots are done manually (ie they are user initiated.) They are stored even after you delete the original RDS instance, unlike automated backups.

Whenever you restore either an Automatic Backup or a manual Snapshot, the restored version of the database will be a new RDS instance with a new DNS endpoint.

Encryption at rest is supported for MySQL, Oracle, SQL Server, PostgreSQL, MariaDB & Aurora. 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.

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 Instance failure, or an Availability Zone failure, Amazon RDS will automatically failover to the standby so that database operations can resume quickly without administrative intervention.

## Multi-AZ is available for the following databases

- SQL Server
- Oracle
- MySQL Server
- PostgreSQL
- MariaDB

Multi AZ is only for disaster recovery not for performance.

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 Replicas are available for the following databases

- MySQL Server
- PostgreSQL
- MariaDB
- Aurora

## Things to know about Read Replicas;

- Used for scaling, not for DR!
- Must have automatic backups turned on in order to deploy a read replica.
- You can have up to 5 read replica copies of any database.
- You can have read replicas of read replicas (but watch out for latency.)

## Things to know about Read Replicas;

- Each read replica will have its own DNS end point.
- You can have read replicas that have Multi-AZ.
- You can create read replicas of Multi-AZ source databases.
- Read replicas can be promoted to be their own databases. This breaks the replication.
- You can have a read replica in a second region.

## Read Replicas

- Can be Multi-AZ.
- Used to increase performance.
- Must have backups turned on.
- Can be in different regions.
- Can be Aurora or MySQL.
- Can be promoted to master, this will break the Read Replica

## MultiAZ

- Used For DR.
- You can force a failover from one AZ to another by rebooting the RDS instance.

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

0:36 / 3:51 Transcript Browse Q&A

**The basics of DynamoDB** are as follows;

- Stored on SSD storage
- Spread across 3 geographically distinct data centres
- 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 updated 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.

## The basics of DynamoDB are as follows;

- Stored on SSD storage
- Spread across 3 geographically distinct data centres
- Eventual Consistent Reads (Default)
- Strongly Consistent Reads

**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 $1,000 per terabyte per year, less than a tenth of most other data warehousing solutions.**

**Data Warehousing databases use different type of architecture both from a database perspective and infrastructure layer.**

**Redshift can be configured as follows**

- 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.

**Advanced Compression:**

Columnar data stores can be compressed much more than row-based data stores because similar data is stored sequentially on disk. Amazon Redshift employs multiple compression techniques and can often achieve significant compression relative to traditional relational data stores. In addition, Amazon Redshift doesn't require indexes or materialized views, and so uses less space than traditional relational database systems. When loading data into an empty table, Amazon Redshift automatically samples you most appropriate compression scheme.

57 people bookmarked this moment.

2:56 / 6:32 Transcript Browse Q&A Add Bookmark

**Massively Parallel Processing (MPP):**

Amazon Redshift automatically distributes data and query load across all nodes. Amazon Redshift makes it easy to add nodes to your data warehouse and enables you to maintain fast query performance as your data warehouse grows.

## Backups

- Enabled by default with a 1 day retention period.
- Maximum retention period is 35 days.
- Redshift always attempts to maintain at least three copies of your data (the original and replica on the compute nodes and a backup in Amazon S3).
- Redshift can also asynchronously replicate your snapshots to S3 in another region for disaster recovery.

## Redshift is priced as follows;

- Compute Node Hours (total number of hours you run across all your compute nodes for the billing period. You are billed for 1 unit per node per hour, so a 3-node data warehouse cluster running persistently for an entire month would incur 2,160 instance hours. You will not be charged for leader node hours; only compute nodes will incur charges.)
- Backup
- Data transfer (only within a VPC, not outside it)

4:59 / 6:32 Transcript Browse Q&A Add Bookmark

## Security Considerations:

- Encrypted in transit using SSL
- Encrypted at rest using AES-256 encryption
- By default RedShift takes care of key management.
    - Manage your own keys through HSM
    - AWS Key Management Service

## Redshift Availability:

- Currently only available in 1 AZ
- Can restore snapshots to new AZs in the event of an outage.

Amazon Aurora is a MySQL-compatible, relational database engine that combines the speed and availability of high-end commercial databases with the simplicity and cost-effectiveness of open source databases. Amazon Aurora provides up to five times better performance than MySQL at a price point one tenth that of a commercial database while delivering similar performance and availability.

0:23 / 7:12 Transcript Browse Q&A

## Things to know about Aurora;

- Start with 10GB, Scales in 10GB increments to 64TB (Storage Autoscaling)
- Compute resources can scale up to 32vCPUs and 244GB of Memory.
- 2 copies of your data is contained in each availability zone, with minimum of 3 availability zones. 6 copies of your data.

So, Aurora is only available in regions that have 3 AZs.

## Scaling Aurora;

- Aurora is designed to transparently handle the loss of up to two copies of data without affecting database write availability and up to three copies without affecting read availability.
- Aurora storage is also self-healing. Data blocks and disks are continuously scanned for errors and repaired automatically.

## Two Types of Aurora Replicas are available;

- Aurora Replicas (currently 15)
- MySQL Read Replicas (currently 5)

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

Automatic failover is not possible for read replication enabled mysql db. It is however possible for Aurora.

## Backups With Aurora

- Automated backups are always enabled on Amazon Aurora DB Instances. Backups do not impact database performance.
- You can also take snapshots with Aurora. This also does not impact on performance.
- You can share Aurora Snapshots with other AWS accounts.

- 2 copies of your data is contained in each availability zone, with minimum of 3 availability zones. 6 copies of your data.
- You can share Aurora Snapshots with other AWS accounts.
- 2 types of replicas available. Aurora Replicas and MySQL replicas. Automated failover is only available with Aurora Replicas.
- Aurora has automated backups turned on by default. You can also take Snapshots with Aurora. You can share these snapshots with other AWS accounts.

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

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

- Use Elasticache to increase database and web application performance.
- Reddis is Multi-AZ
- You can do back ups and restores of Reddis
- If you need to scale horizontally, use Memcached

**Remember the following points;**

- RDS runs on virtual machines
- You cannot log in to these operating systems however.
- Patching of the RDS Operating System and DB is Amazon's responsibility
- RDS is NOT Serverless
- Aurora Serverless IS Serverless

Performance can be improved either by having read replicas or using elastic cache

## Read Replicas

- Can be Multi-AZ.
- Used to increase performance.
- Must have backups turned on.
- Can be in different regions.
- Can be Aurora or MySQL.
- Can be promoted to master, this will break the replication with the Read Replica

## MultiAZ

- Used For DR.
- You can force a failover from one AZ to another by rebooting the RDS instance.

Route 53:

DNS port is 53

If you've used the internet, you've used DNS. DNS is used to convert human friendly domain names (such as http://acloud.guru) into an Internet Protocol (IP) address (such as http://82.124.53.1).

IP addresses are used by computers to identify each other on the network. IP addresses commonly come in 2 different forms, IPv4 and IPv6.

## IPv4 Addresses are running out...

The IPv4 space is a 32 bit field and has over 4 billion different addresses (4,294,967,296 to be precise).

IPv6 was created to solve this depletion issue and has an address space of 128bits which in theory is

340,282,366,920,938,463,463,374,607,431,768,211,456 addresses or 340 undecillion addresses.

If we look at common domain names such as google.com, bbc.co.uk. acloud.guru etc., you will notice a string of characters separated by dots (periods). The last word in a domain name represents the "top level domain". The second word in a domain name is known as a second level domain name (this is optional though and depends on the domain name).

.com
.edu
.gov
.co.uk
.gov.uk
.com.au

04:24 / 14:18 Transcript Browse Q&A Add Bookmark Continue

These top level domain names are controlled by the Internet Assigned Numbers Authority (IANA) in a root zone database which is essentially a database of all available top level domains. You can view this database by visiting:

**http://www.iana.org/domains/root/db**

https://www.iana.org/domains/root/db

Because all of the names in a given domain name have to be unique there needs to be a way to organize this all so that domain names aren't duplicated. This is where domain registrars come in. A registrar is an authority that can assign domain names directly under one or more top-level domains. These domains are registered with InterNIC, a service of ICANN, which enforces uniqueness of domain names across the Internet. Each domain name becomes registered in a central database known as the WhoIS database.

Popular domain registrars include

Amazon, GoDaddy.com, 123-reg.co.uk etc.

**Start Of Authority Record (SOA)**

## The SOA record stores information about:

- The name of the server that supplied the data for the zone.
- The administrator of the zone.
- The current version of the data file.
- The default number of seconds for the time-to-live file on resource records.

## NS stands for Name Server Records

They are used by Top Level Domain servers to direct traffic to the Content DNS server which contains the authoritative DNS records.

## What's an A record?

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

## What's an TTL?

The length that a DNS record 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.

## What's a CName?

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 the name http://mobile.acloud.guru to resolve to this same address.

## Alias Records

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

Alias records work like a CNAME record in that you can map one DNS name (www.example.com) to another 'target' DNS name (elb1234.elb.amazonaws.com).

## Alias Records

Key difference - A CNAME can't be used for naked domain names (zone apex record.) You can't have a CNAME for http://acloud.guru, it must be either an A record or an Alias.

## Route53 Exam Tips

- ELBs do not have pre-defined IPv4 addresses; you resolve to them using a DNS name.
- Understand the difference between an Alias Record and a CNAME.
- Given the choice, always choose an Alias Record over a CNAME.

## Common DNS Types

- SOA Records
- NS Records
- A Records
- CNAMES
- MX Records
- PTR Records

https://computer.howstuffworks.com/dns.htm/printable

Route53 is under network.

Each region EC2 should have its own security group and keypair

- You can buy domain names directly with AWS.
- It can take up to 3 days to register depending on the circumstances.

## The Following Routing Policies Are Available With Route53:

- Simple Routing
- Weighted Routing
- Latency-based Routing
- Failover Routing
- Geolocation Routing
- Geoproximity Routing (Traffic Flow Only)
- Multivalue Answer Routing

These are various ip addresses related to all EC2 instances.

We can remove an EC2 instance if it fails health check. It happens automatically. For this, we need to create health check as below.

This option is available from left hand side menu in the route53 itself.

It takes the total weight percentages we gave and divides the configured instances proportionally. If we have 20 and 30, the traffic gets divided into 40 and 60.

## Failover Routing Policy

Failover routing policies are used when you want to create an active/passive set up. For example, you may want your primary site to be in EU-WEST-2 and your secondary DR Site in AP-SOUTHEAST-2.

Route53 will monitor the health of your primary site using a health check.

A health check monitors the health of your end points.

## Geolocation Routing Policy

Geolocation routing lets you choose where your traffic will be sent based on the geographic location of your users (ie the location from which DNS queries originate). For example, you might want all queries from Europe to be routed to a fleet of EC2 instances that are specifically configured for your European customers. These servers may have the local language of your European customers and all prices are displayed in Euros.

## Geoproximity Routing (Traffic Flow Only)

Geoproximity routing lets Amazon Route 53 route traffic to your resources based on the geographic location of your users and your resources. You can also optionally choose to route more traffic or less to a given resource by specifying a value, known as a bias. A bias expands or shrinks the size of the geographic region from which traffic is routed to a resource.

**To use geoproximity routing, you must use Route 53 traffic flow.**

This is very complicated and not asked in the exam.

## Multivalue Answer Policy

Multivalue answer routing lets you configure Amazon Route 53 to return multiple values, such as IP addresses for your web servers, in response to DNS queries. You can specify multiple values for almost any record, but multivalue answer routing also lets you check the health of each resource, so Route 53 returns only values for healthy resources.

**This is similar to simple routing however it allows you to put health checks on each record set.**

## Route53 Exam Tips

- ELBs do not have pre-defined IPv4 addresses; you resolve to them using a DNS name.
- Understand the difference between an Alias Record and a CNAME.
- Given the choice, always choose an Alias Record over a CNAME.

VPCS:

5 to 10 questions will come on exam.

This is virtual data center in the cloud.

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the Amazon Web Services (AWS) Cloud where you can launch AWS resources in a virtual network that you define. You have complete 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 webservers that has access to the Internet, and place your backend systems such as databases or application servers 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 we 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
- Instance 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 instance has both a public and private IP address.

## VPC Peering

- Allows you to connect one VPC with another via a direct network route using private IP addresses.
- 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!!!
- You can peer between regions.

Here we need to explicitly setup peering between b and c. b cannot go thru a to talk to C

DNS: Domain Name System

you just connect through a **domain name server**, also called a **DNS server** or **name server**, which manages a massive database that maps domain names to IP addresses.

you're probably more likely to remember "howstuffworks.com" when you want to return later. In addition, a website's IP address can change over time, and some sites associate multiple IP addresses with a single domain name.

Without DNS servers, the internet would shut down very quickly. But how does your computer know what DNS server to use? Typically, when you connect to your home network, internet service provider (ISP) or WiFi network, the modem or router that assigns your computer's network address also sends some important network configuration information to your computer or mobile device. That configuration includes one or more DNS servers that the device should use when translating DNS names to IP address.

With so much to handle, DNS servers rely on network efficiency and internet protocols. Part of the IP's effectiveness is that each machine on a network has a unique IP address in both the IPV4 and IPV6 standards managed by the Internet Assigned Numbers Authority (IANA). Here are some ways to recognize an IP address:

- An IP address in the IPV4 standard has four numbers separated by three decimals, as in: 70.74.251.42
- An IP address in the IPV6 standard has eight hexadecimal numbers (base-16) separated by colons, as in 2001:0cb8:85a3:0000:0000:8a2e:0370:7334. Because IPV6 is still a

very new standard, we'll concentrate on the more common IPV4 for this article.

Certain addresses and ranges are designated by the IANA as **reserved IP addresses**, which means they have a specific job in IP. For example, the IP address 127.0.0.1 is reserved to identify the computer you're currently using. So, talking to 127.0.0.1 is just talking to yourself

Where does your computer's IP address come from? If we're talking about your desktop or laptop computer, it probably comes from a Dynamic Host Configuration Protocol (DHCP) server on your network. The job of a DHCP server is to make sure your computer has the IP address and other network configuration it needs whenever you're online. Because this is "dynamic," the IP address for your computer will probably change from time to time, such as when you shut down your computer for a few days. As the user, you'll probably never notice all this taking place. See the sidebar on this page for hints on where to find the IP address assigned to your computer or mobile device.

Web servers and other computers that need a consistent point of contact use **static IP addresses**. This means that the same IP address is always assigned to that system's network interface when it's online. To make sure that interface always gets the same IP address, IP associates the address with the Media Access Control (MAC) address for that network interface. Every network interface, both wired and wireless, has a unique MAC address embedded in it by the manufacturer.

Now, let's look at the other side of the DNS equation: domain names.

Note that if you're on a home or small local network, your address will probably be in the form 192.168.x.x, 172.16.x.x or 10.x.x.x (where x is a number between 0 and 255). These are reserved addresses used on each local network, and a router on that network then connects you to the internet

You'll recognize domain names as having strings of characters separated by dots (periods). The last word in a domain name represents a **top-level domain**. These top-level domains are controlled by the IANA in what's called the Root Zone Database, which we'll examine more closely later. There are more than 1,000 top-level domains, and here are some of the most common:

- COM — commercial websites, though open to everyone
- NET — network websites, though open to everyone

- ORG — non-profit organization websites, though open to everyone
- EDU — restricted to schools and educational organizations
- MIL — restricted to the U.S. military
- GOV — restricted to the U.S. government
- US, UK, RU and other two-letter country codes — each is assigned to a domain name authority in the respective country

DNS servers that perform the first task are normally managed by your internet service provider (ISP). As mentioned earlier, the ISP's DNS server is part of the network configuration you get from DHCP as soon as you go online. These servers reside in your ISP's data centers, and they handle requests as follows:

- If it has the domain name and IP address in its database, it resolves the name itself.
- If it doesn't have the domain name and IP address in its database, it contacts another DNS server on the internet. It may have to do this multiple times.
- If it has to contact another DNS server, it caches the lookup results for a limited time so it can quickly resolve subsequent requests to the same domain name.
- If it has no luck finding the domain name after a reasonable search, it returns an error indicating that the name is invalid or doesn't exist.

A DNS server that manages a specific domain is called the **start of authority** (**SOA**) for that domain. Over time, the results from looking up hosts at the SOA will propagate to other DNS servers, which in turn propagate to other DNS servers, and so on across the internet.

This propagation is a result of each DNS server caching the lookup result for a limited time, known as its **Time To Live** (TTL), ranging from a few minutes to a few days. People managing a DNS server can configure its TTL, so TTL values will vary across the Internet. So, each time you look up "www.howstuffworks.com," it's possible that the DNS server for your ISP will find the lookup results "70.42.251.42" in its own cache if you or someone else using that server looked for it before within the server's TTL.

Each new configuration you add is called a record, and the following are the most common types of records you can configure for your DNS server

- Host (A) — This is the basic mapping of IP address to host name, the essential component for any domain name.
- Canonical Name (CNAME) — This is an alias for your domain. Anyone accessing that alias will be automatically directed to the server indicated in the A record.
- Mail Exchanger (MX) — This maps e-mail traffic to a specific server. It could indicate another host name or an IP address. For example, people who use Google for the e-mail for their domain will create an MX record that points to ghs.google.com.
- Name Server (NS) — This contains the name server information for the zone. If you configure this, your server will let other DNS servers know that yours is the ultimate authority (SOA) for your domain when caching lookup information on your domain from other DNS servers around the world.
- Start of Authority (SOA) — This is one larger record at the beginning of every zone file with the primary name server for the zone and some other information. If your registrar or hosting company is running your DNS server, you won't need to manage this. If you're managing your own DNS, you can peek at tips for doing so using this article on the structure of a DNS SOA Record.

Typical users will probably get the most use out of MX and CNAME records. The MX records allows you to point your mail services somewhere other than your hosting company if you choose to use something like Google Apps for your domain. The CNAME records let you point host names for your domain to various other locations. This could include setting google.example.com to redirect to google.com, or setting up a dedicated game server with its own IP address and pointing it to something like gameserver.example.com.

Developed by Cisco, Network Address Translation is used by a device (firewall, router or computer that sits between an internal network and the rest of the world.

\

When a new VPC is created, by default following components are created.

What is CIDR address

Subnet?

One subnet is for one avialabity zone -> important question

There could be only one internet gateway per VPC

CIDR notation: Classless Inter Domain Routing

For IPv4, CIDR notation is an alternative to the older system of representing networks by their starting address and the subnet mask, both written in dot-decimal notation. *192.168.100.0/24* is equivalent to *192.168.100.0/255.255.255.0*.

The number of addresses of a subnet may be calculated as $2^{\text{address length} - \text{prefix length}}$, where *address length* is 128 for IPv6 and 32 for IPv4. For example, in IPv4, the prefix length */29* gives: $2^{32-29} = 2^3 = 8$ addresses.

In common usage, the first address in a subnet, all binary zero in the host identifier, is reserved for referring to the network itself, while the last address, all binary one in the host identifier, is used as a broadcast address for the network; this reduces the number of addresses available for

hosts by 2. As a result, a */31* network, with one binary digit in the host identifier, is rarely used, as such a subnet would provide no available host addresses after this reduction.

Any address starting with 127 is a loop back address. Ie., pinging localmachine itself.

- IP classes provide a default mask based on the number in the first octet

| Class | 1st Octet range | Default Mask |
|---|---|---|
| Class A | 1 – 126 | 255.0.0.0 |
| Class B | 128 – 191 | 255.255.0.0 |
| Class C | 192 – 223 | 255.255.255.0 |
| Loopback | 127 | |
| Class D | 224 – 239 | Multicasting |
| ~~Class E~~ | ~~240 – 255.255.255.254~~ | ~~Experimental~~ |

## Class C

C 192 - 223 255.255.255.0

192.168.1.0 255.255.255.0

128 64 32 16 | 8 4 2 1

Subnet # 1 - 192.168.1.0/26 255.255.255.192

Broadcast - 192.168.1.63

Subnet # 2 - 192.168.1.64/28 255.255.255.240

Broadcast - 192.168.1.79

Subnet # 3 - 192.168.1.80/28 255.255.255.240

Broadcast - 192.168.1.95

Requirements:

Create 1 subnet that contains at least 50 usable ip addresses.

Create 2 subnets that contain 14 usable IP addresses each

Do not waste any address space

# THE PURPOSE OF A ROUTER

- Purpose: Route traffic from one network to another network
- **IP Router**: the most widely used type of Router
  - Routes IP Packets from one network to another
  - Routers are the network gateway device
  - Connects Local Area Networks (LANs) to Wide Area Networks (WANs)

# HOME/SOHO ROUTER

- Home routers are normally combo devices
  - Router
  - Switch
  - Wireless Access Point

VLAN helps us to manage same switch maintain multiple logically separated LANs

Switch

Router

Data is put into a packet with source, destination addresses. Then it is put into frame along with gateway address as destination. Once gateway receives it, it puts into a new frame and sends it to destination.

VPC Lab:

Without any configuration from developers' end, we can see some pre-configured options as below

If we create a new VPC, we are automatically creating Route table, acl, security group

We should create subnet explicitly

One subnet will be only in one AZ

Cidr.xyz tells us how many ip addresses are available and starting/ending ip

Reserved ip addresses

- `10.0.0.0`: Network address.
- `10.0.0.1`: Reserved by AWS for the VPC router.
- `10.0.0.2`: Reserved by AWS. The IP address of the DNS server is always the base of the VPC network range plus two; however, we also reserve the base of each subnet range plus two. For VPCs with multiple CIDR blocks, the IP address of the DNS server is located in the primary CIDR. For more information, see Amazon DNS Server.
- `10.0.0.3`: Reserved by AWS for future use.
- `10.0.0.255`: Network broadcast address. We do not support broadcast in a VPC, therefore we reserve this address.

There can only be one gateway for one VPC

After creating A VPC, we can provision EC2 instances. We can associate VPC and Subnet as below

To be able to connect from public subnet to private subnet, a new security group is required as below

ICMP is for ping

NAT: Network Address Translation

**NAT Instances**. You can use a network address translation (**NAT**) **instance** in a public subnet in your VPC to enable **instances** in the private subnet to initiate outbound IPv4 traffic to the Internet or other AWS services, but prevent the**instances** from receiving inbound traffic initiated by someone on the Internet.

For the VPC subnet we created earlier, the private subnet cannot access internet to download and install software etc. NAT should help us here

NAT instance is EC2 instance

NAT gateway is highly available gateway that can do same thing as NAT instance

To create NAT instance, go to community AMIs and search for NAT

## Disabling Source/Destination Checks

Each EC2 instance performs source/destination checks by default. This means that the instance must be the source or destination of any traffic it sends or receives. However, a NAT instance must be able to send and receive traffic when the source or destination is not itself. Therefore, you must disable source/destination checks on the NAT instance.

You can disable the `SrcDestCheck` attribute for a NAT instance that's either running or stopped using the console or the command line.

**To disable source/destination checking using the console**

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the navigation pane, choose **Instances**.
3. Select the NAT instance, choose **Actions**, **Networking**, **Change Source/Dest. Check**.
4. For the NAT instance, verify that this attribute is disabled. Otherwise, choose **Yes, Disable**.
5. If the NAT instance has a secondary network interface, choose it from **Network interfaces** on the **Description** tab and choose the interface ID to go to the network interfaces page. Choose **Actions**, **Change Source/Dest. Check**, disable the setting, and choose **Save**.

**To disable source/destination checking using the command line**

You can use one of the following commands. For more information, see Accessing Amazon VPC.

- modify-instance-attribute (AWS CLI)
- Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

## Nat Instances Exam Tips

- When creating a NAT instance, Disable Source/Destination Check on the Instance.
- NAT instances must be in a public subnet.
- There must be a route out of the private subnet to the NAT instance, in order for this to work.
- The amount of traffic that NAT instances can support 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 AZs, and a script to automate failover.
- Behind a Security Group.

80 people book

## Nat Gateways

- Redundant inside the Availability Zone
- Preferred by the enterprise
- Starts at 5Gbps and scales currently to 45Gbps
- No need to patch
- Not associated with security groups
- Automatically assigned a public ip address
- Remember to update your route tables.
- No need to disable Source/Destination Checks

## Nat Gateways

- If you have resources in multiple Availability Zones and they share one NAT gateway, in the event that the NAT gateway's Availability Zone is down, resources in the other Availability Zones lose internet access. To create an Availability Zone-independent architecture, create a NAT gateway in each Availability Zone and configure your routing to ensure that resources use the NAT gateway in the same Availability Zone.

We can create ACLs for inbound and outbound separately

Here, the last one is called ephemeral

# Ephemeral port

From Wikipedia, the free encyclopedia

An **ephemeral port** is a short-lived transport protocol port for Internet Protocol (IP) communications. Ephemeral ports are allocated automatically from a predefined range by the IP stack software. An ephemeral port is typically used by the Transmission Control Protocol (TCP), User Datagram Protocol (UDP), or the Stream Control Transmission Protocol (SCTP) as the port assignment for the client end of a client–server communication to a well-known port on a server.

On servers, ephemeral ports may also be used as the port assignment on the server end of a communication. This is done to continue communications with a client that initially connected to one of the server's well-known service listening ports. File Transfer Protocol (FTP) and Remote Procedure Call (RPC) applications are two protocols that can behave in this manner. Note that the term "server" here includes workstations running network services that receive connections initiated from other clients (e.g. Remote Desktop Protocol).

The allocations are temporary and only valid for the duration of the communication session. After completion (or timeout) of the communication session, the ports become available for reuse.[note 1] Since the ports are used on a per request basis they are also called **dynamic ports**.

The example network ACL in the preceding section uses an ephemeral port range of 32768-65535. However, you might want to use a different range for your network ACLs depending on the type of client that you're using or with which you're communicating.

The client that initiates the request chooses the ephemeral port range. The range varies depending on the client's operating system. Many Linux kernels (including the Amazon Linux kernel) use ports 32768-61000. Requests originating from Elastic Load Balancing use ports 1024-65535. Windows operating systems through Windows Server 2003 use ports 1025-5000. Windows Server 2008 and later versions use ports 49152-65535. A NAT gateway uses ports 1024-65535. For example, if a request comes into a web server in your VPC from a Windows XP client on the Internet, your network ACL must have an outbound rule to enable traffic destined for ports 1025-5000.

If an instance in your VPC is the client initiating a request, your network ACL must have an inbound rule to enable traffic destined for the ephemeral ports specific to the type of instance (Amazon Linux, Windows Server 2008, and so on).

In practice, to cover the different types of clients that might initiate traffic to public-facing instances in your VPC, you can open ephemeral ports 1024-65535. However, you can also add rules to the ACL to deny traffic on any malicious ports within that range. Ensure that you place the DENY rules earlier in the table than the ALLOW rules that open the wide range of ephemeral ports.

If we want to deny a particular ip address, keep at the top

| Rule # | Type | Protocol | Port Range | Source | Allow / Deny |
|---|---|---|---|---|---|
| 99 | HTTP (80) | TCP (6) | 80 | 107.16.110.187/32 | DENY |
| 100 | HTTP (80) | TCP (6) | 80 | 0.0.0.0/0 | ALLOW |
| 200 | HTTPS (443) | TCP (6) | 443 | 0.0.0.0/0 | ALLOW |
| 300 | SSH (22) | TCP (6) | 22 | 0.0.0.0/0 | ALLOW |
| * | ALL Traffic | ALL | ALL | 0.0.0.0/0 | DENY |
| * | ALL Traffic | ALL | ALL | ::/0 | DENY |

Exam tip: Understand how these allow and deny rules work and their priorities

A subnet can only have one ACL at any time

ACL can be assigned to multiple subnets

Network ACLs evaluated first then security groups.

**Remember the following for your exam;**

- Your VPC automatically comes a default network ACL, and by default it allows all outbound and inbound traffic.
- You can create custom network ACLs. 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 don't explicitly associate a subnet with a network ACL, the subnet is automatically associated with the default network ACL.
- Block IP Addresses using network ACLs not Security Groups

**Remember the following for your exam;**

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

Exam tip: If we have to ELB, we need to have atleast 2 public facing subnets.

## Default Network ACL

The default network ACL is configured to allow all traffic to flow in and out of the subnets with which it is associated. Each network ACL also includes a rule whose rule number is an asterisk. This rule ensures that if a packet doesn't match any of the other numbered rules, it's denied. You can't modify or remove this rule.

The following is an example default network ACL for a VPC that supports IPv4 only.

| **Inbound** |
| --- |

| 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** | | | | | |
| **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 |

## What Are VPC Flow Logs?

VPC Flow Logs is a feature that enables you to capture information about the IP traffic going to and from network interfaces in your VPC. Flow log data is stored using Amazon CloudWatch Logs. After you've created a flow log, you can view and retrieve its data in Amazon CloudWatch Logs.

**Flow logs can be created at 3 levels;**

- VPC
- Subnet
- Network Interface Level

- You cannot enable flow logs for VPCs that are peered with your VPC unless the peer VPC is in your account.
- You cannot tag a flow log.
- After you've created a flow log, you cannot change its configuration; for example, you can't associate a different IAM role with the flow log.

Flow logs allows you to put the accepted/rejected traffic info in either cloudwatch logs or S3 bucket

**Not all IP Traffic is monitored;**

- Traffic generated by instances when they contact the Amazon DNS server. If you use your own DNS server, then all traffic to that DNS server is logged.
- Traffic generated by a Windows instance for Amazon Windows license activation.
- Traffic to and from 169.254.169.254 for instance metadata.
- DHCP traffic.
- Traffic to the reserved IP address for the default VPC router.

**A Bastion Host:**

A bastion host is a special purpose computer on a network specifically designed and configured to withstand attacks. The computer generally hosts a single application, for example a proxy server, and all other services are removed or limited to reduce the threat to the computer. It is hardened in this manner primarily due to its location and purpose, which is either on the outside of a firewall or in a demilitarized zone (DMZ) and usually involves access from untrusted networks or computers.

## Remember the following;

- A NAT Gateway or NAT Instance is used to provide internet traffic to EC2 instances in a private subnets.
- A Bastion is used to securely administer EC2 instances (Using SSH or RDP). Bastions are called Jump Boxes in Australia.
- You cannot use a NAT Gateway as a Bastion host.

## Direct Connect

AWS Direct Connect is a cloud service solution that makes it easy to establish a dedicated network connection from your premises to AWS. Using AWS Direct Connect, you can establish private connectivity between AWS and your datacenter, office, or colocation environment, which in many cases can reduce your network costs, increase bandwidth throughput, and provide a more consistent network experience than Internet-based connections.

**Remember the following;**

- Direct Connect directly connects your data center to AWS
- Useful for high throughput workloads (ie lots of network traffic)
- Or if you need a stable and reliable secure connection.

## What Is A VPC Endpoint?

### A VPC Endpoint:

A VPC endpoint enables you to privately connect your VPC to supported AWS services and VPC endpoint services powered by PrivateLink without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC do not require public IP addresses to communicate with resources in the service. Traffic between your VPC and the other service does not leave the Amazon network.

Endpoints are virtual devices. They are horizontally scaled, redundant, and highly available VPC components that allow communication between instances in your VPC and services without imposing availability risks or bandwidth constraints on your network traffic.

**An interface endpoint is an elastic network interface with a private IP address that serves as an entry point for traffic destined to a supported service. The following services are supported:**

- Amazon API Gateway
- AWS CloudFormation
- Amazon CloudWatch
- Amazon CloudWatch Events
- Amazon CloudWatch Logs
- AWS CodeBuild
- AWS Config
- Amazon EC2 API
- Elastic Load Balancing API
- AWS Key Management Service
- Amazon Kinesis Data Streams
- Amazon SageMaker and Amazon SageMaker Runtime
- Amazon SageMaker Notebook Instance
- AWS Secrets Manager
- AWS Security Token Service
- AWS Service Catalog
- Amazon SNS
- Amazon SQS
- AWS Systems Manager
- Endpoint services hosted by other AWS accounts
- Supported AWS Marketplace partner services

01:43 / 11:02 Transcript Browse Q&A Add Bookmark Continue

This allows the VPC subnet machines to communicate with AWS services using AWS private lines instead of going through internet.

**There are two types of VPC endpoints:**

- Interface Endpoints
- Gateway Endpoints

**Currently Gateway Endpoints Support**

- Amazon S3
- DynamoDB

**Remember the following;**

- Think of a VPC as a logical datacenter in AWS.
- Consists of IGWs (Or Virtual Private Gateways), Route Tables, Network Access Control Lists, Subnets, and Security Groups
- 1 Subnet = 1 Availability Zone
- Security Groups are Stateful; Network Access Control Lists are Stateless
- NO TRANSITIVE PEERING

## Remember the following;

- When you create a VPC a default Route Table, Network Access Control List (NACL) and a default Security Group.
- It won't create any subnets, nor will it create a default internet gateway.
- US-East-1A in your AWS account can be a completely different availability zone to US-East-1A in another AWS account. The AZ's are randomized.
- Amazon always reserve 5 IP addresses within your subnets.
- You can only have 1 Internet Gateway per VPC.
- Security Groups can't span VPCs.

## Nat Instances Exam Tips

- When creating a NAT instance, Disable Source/Destination Check on the Instance.
- NAT instances must be in a public subnet.
- There must be a route out of the private subnet to the NAT instance, in order for this to work.
- The amount of traffic that NAT instances can support 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 AZs, and a script to automate failover.
- Behind a Security Group.

## Nat Gateways

- Redundant inside the Availability Zone
- Preferred by the enterprise
- Starts at 5Gbps and scales currently to 45Gbps
- No need to patch
- Not associated with security groups
- Automatically assigned a public ip address
- Remember to update your route tables.
- No need to disable Source/Destination Checks

## Nat Gateways

- If you have resources in multiple Availability Zones and they share one NAT gateway, in the event that the NAT gateway's Availability Zone is down, resources in the other Availability Zones lose internet access. To create an Availability Zone-independent architecture, create a NAT gateway in each Availability Zone and configure your routing to ensure that resources use the NAT gateway in the same Availability Zone.

**Remember the following for your exam;**

- Your VPC automatically comes a default network ACL, and by default it allows all outbound and inbound traffic.
- You can create custom network ACLs. 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 don't explicitly associate a subnet with a network ACL, the subnet is automatically associated with the default network ACL.
- Block IP Addresses using network ACLs not Security Groups

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

- You need a minimum of two public subnets to deploy an internet facing loadbalancer.

- You cannot enable flow logs for VPCs that are peered with your VPC unless the peer VPC is in your account.
- You cannot tag a flow log.
- After you've created a flow log, you cannot change its configuration; for example, you can't associate a different IAM role with the flow log.

- Traffic generated by instances when they contact the Amazon DNS server. If you use your own DNS server, then all traffic to that DNS server is logged.
- Traffic generated by a Windows instance for Amazon Windows license activation.
- Traffic to and from 169.254.169.254 for instance metadata.
- DHCP traffic.
- Traffic to the reserved IP address for the default VPC router.

- A NAT Gateway or NAT Instance is used to provide internet traffic to EC2 instances in a private subnets.
- A Bastion is used to securely administer EC2 instances (Using SSH or RDP). Bastions are called Jump Boxes in Australia.
- You cannot use a NAT Gateway as a Bastion host.

**Remember the following;**

- Direct Connect directly connects your data center to AWS
- Useful for high throughput workloads (ie lots of network traffic)
- Or if you need a stable and reliable secure connection.

**A VPC Endpoint:**

A VPC endpoint enables you to privately connect your VPC to supported AWS services and VPC endpoint services powered by PrivateLink without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC do not require public IP addresses to communicate with resources in the service. Traffic between your VPC and the other service does not leave the Amazon network.

Endpoints are virtual devices. They are horizontally scaled, redundant, and highly available VPC components that allow communication between instances in your VPC and services without imposing availability risks or bandwidth constraints on your network traffic.

Load Balancer:

**Load Balancer Types**

1. Application Load Balancer
2. Network Load Balancer
3. Classic Load Balancer

**Application Load Balancers** are best suited for load balancing of HTTP and HTTPS traffic. They operate at Layer 7 and are application-aware. They are intelligent, and you can create advanced request routing, sending specified requests to specific web servers.

App load balancers are intelligent in the sense that they can look at the request attributes and for example load balance among France language based web servers etc.

**Network Load Balancers** are best suited for load balancing of TCP traffic where extreme performance is required. Operating at the connection level (Layer 4), Network Load Balancer are capable of handling millions of requests per second,while maintaining ultra-low latencies.

Use for extreme performance!

**Classic Load Balancers** are the legacy Elastic Load Balancers. You can load balance HTTP/HTTPS applications and use Layer 7-specific features, such as X-Forwarded and sticky sessions. You can also use strict Layer 4 load balancing for applications that rely purely on the TCP protocol.

**Classic Load Balancers** -if your application stops responding, the ELB (Classic Load Balancer) responds with a 504 error.
This means that the application is having issues. This could be either at the Web Server layer or at the Database Layer.
Identify where the application is failing, and scale it up or out where possible.

46 people bookmarked this moment.

X-Forwarded for contains the actual ip address of the customer's device.

- 504 Error means the gateway has timed out. This means that the application not responding within the idle timeout period.
- Trouble shoot the application. Is it the Web Server or Database Server?

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

There will be 10 questions on ELB. So, read FAQs

For ELB Lab,

Classic load balancer,

Create two EC2 instances in different available zones,

Create a load balancer

We need to configure health check

DNS name will be given for LB

After LB is created, EC2 instances will be in OutOfService and after some time, they become InService.

For Application LB, first we need to create Target Group. Target group will group instances based on their characteristics for eX: all European instances, US instances etc.

After target group is created, add targets to it.

Then create application load balancer and add target group to it.

We can customize the rules for the LB we created (if, then etc)

For network LB, we should use it when we have fixed ip addresses or need ultra high performance at tcp level.

Classic LB costs less.

Above is exam question. If an EC2 instance is not receiving any traffic, disable sticky session.

Without crosss zone load balancing, the single instance in second zone receives 50% traffic.

**What Are Path Patterns?**

You can create a listener with rules to forward requests based on the URL path. This is known as path-based routing. If you are running microservices, you can route traffic to multiple back-end services using path-based routing. For example, you can route general requests to one target group and requests to render images to another target group.

Autoscaling group:

This will allow to bring up the instances as and when needed. We can specify how many min and max instances are needed and their AZs. Then it automatically monitors the instances and ensures that min and max are respected.

While creating ACG, we can use the image created from other EC2 instance under My AMIs

HA Archiecture:

Very critical for exam

# Simian Army Projects

- Chaos Monkey
- Chaos Gorilla
- Chaos Kong
- Janitor Monkey
- Doctor Monkey
- Compliance Monkey
- Latency Monkey
- Security Monkey

goo.gl/UR2nzM

**Scenario: You have a website that requires a minimum of 6 instances and it must be highly available. You must also be able to tolerate the failure of 1 Availability Zone. What is the ideal architecture for this environment while also being the most cost effective?**

- 2 Availability Zones with 2 instances in each AZ.
- 3 Availability Zones with 3 instances in each AZ.
- 1 Availability Zone with 6 instances in each AZ.
- 3 Availability Zones with 2 instances in each AZ.

Answer is B

## Remember the following

- Always Design for failure.
- Use Multiple AZ's and Multiple Regions where ever you can.
- Know the difference between Multi-AZ and Read Replicas for RDS.
- Know the difference between scaling out and scaling up.
- Read the question carefully and always consider the cost element.
- Know the different S3 storage classes.

Making all objects in S3 public

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PublicReadGetObject",
            "Effect": "Allow",
            "Principal": "*",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::BUCKET_NAME/*"
            ]
        }
    ]
}
```

CloudFormation:

This is under Management & Governance

We can also choose from existing stacks like sharepoint, wordpress etc based on which organization offers those solutions.

We can also build our own templates

Elastic Beanstalk:

1 or 2 questions

Select all compute services is a popular question

Just choose type of application and it automatically creates all required infra for us. We can then configure it to have autoscaling etc. We can also upload our source code while creating the application.

SQS appears in exam