

We at The Data Monk hold the vision to make sure everyone in the IT industry has an equal stand to work in an open domain such as analytics. Analytics is one domain where there is no formal under-graduation degree and which is achievable to anyone and everyone in the World.

We are a team of 30+ mentors who have worked in various product-based companies in India and abroad, and we have come up with this idea to provide study materials directed to help you crack any analytics interview.

Every one of us has been interviewing for at least the last 6 to 8 years for different positions like Data Scientist, Data Analysts, Business Analysts, Product Analysts, Data Engineers, and other senior roles. We understand the gap between having good knowledge and converting an interview to a top product-based company.

Rest assured that if you follow our different mediums like our blog cum questions-answer portal www.TheDataMonk.com , our youtube channel - [The Data Monk](#), and our e-books, then you will have a very strong candidature in whichever interview you participate in.

There are many blogs that provide free study materials or questions on different analytical tools and technologies, but we concentrate mostly on the questions which are asked in an interview. We have a set of 100+ books which are available both on Amazon and on [The Data Monk e-shop page](#)

We would recommend you to explore our website, youtube channel, and e-books to understand the type of questions covered in our articles. We went for the question-answer approach both on our website as well as our e-books just because we feel that the best way to go from beginner to advance level is by practicing a lot of questions on the topic.

We have launched a series of 50 e-books on our website on all the popular as well as niche topics. Our range of material ranges from SQL, Python, and Machine Learning algorithms to ANN, CNN, PCA, etc.

We are constantly working on our product and will keep on updating it. It is very necessary to go through all the questions present in this book.

Give a rating to the book on Amazon, do provide your feedback and if you want to help us grow then please subscribe to our Youtube channel.

Amazon Web Services (AWS)

AWS is a vast subject. Each service in AWS is a whole new individual topic. what the aim of this article is to introduce the main services like E2,RDS, S3,sage maker with basic steps of creating them with your account and some key properties and features each element has like the security, permissions, tags, policies, etc to give you an overall idea of each topic .

1) What does AWS do?

Aws is a cloud technology service where you can host your website, store the database, convert your machine learning models to web applications without actually having any physical servers or application devices owned by you.

2) What are various services AWS offers?

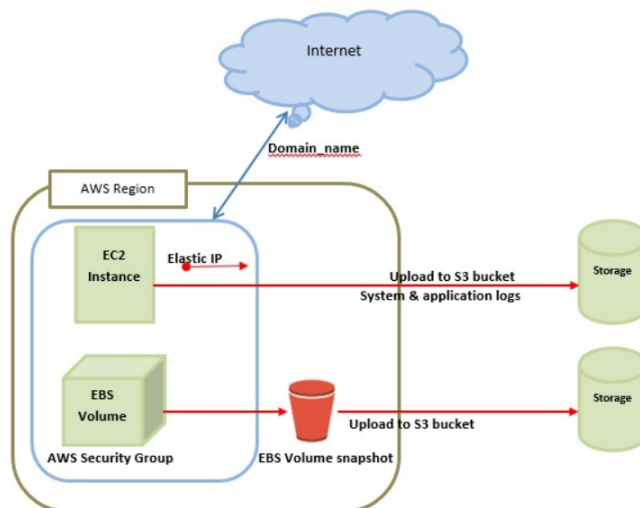
- 1) Web services
- 2) Computer services
- 3) Network services
- 4) Storage services
- 5) Data base services
- 6) Analytics services
- 7) Application services

3) What is cloud computing?

Cloud computing is a computing service which enables the user to get access to remote servers through his system. The computation requirement of the user does not depend on the capacity of his system/device.

Organizations and users can use this to share computing resources and storing rather than setting up the whole thing by themselves.

4) What is the basic structure of AWS?



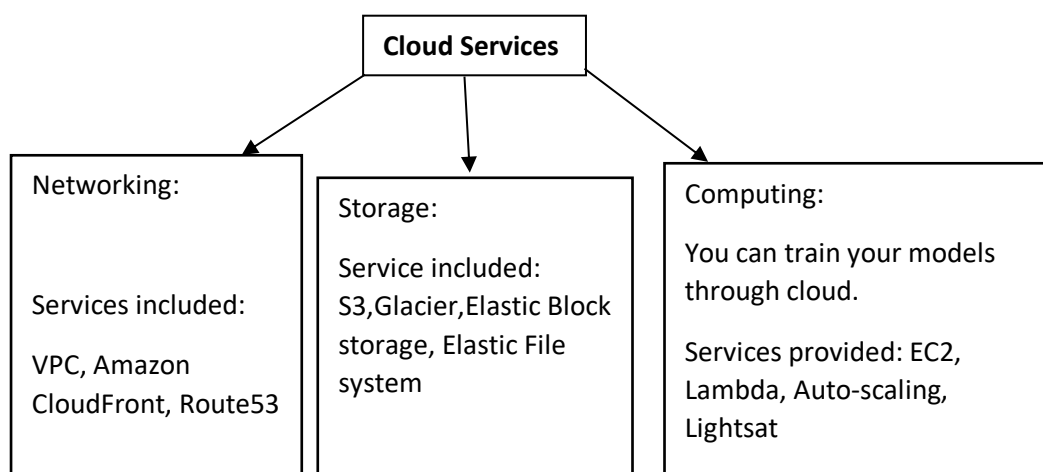
This is the basic architecture of the AWS system.

EC2 stands for elastic compute cloud

S3 stands for simple storage service

EBS-elastic block store

5) What are the types of cloud services AWS provides using which all the major products are built?



6) What is EC2 in AWS architecture?

Ec2(elastic compute cloud) is a web service that provides virtual machines to users that have security, resizable compute capacity based on their capacity, utilization, and requirement.

EC2 instances provide a plethora of types of instances to adapt different use cases.

These are the factors that each type of instance is built on:

- CPU

- Memory storage
- Networking capacity

We can mix these applications based on our necessity.

Instance	vCPU*	CPU Credits/hour	Mem (GiB)	Storage	Network Performance (Gbps)
t3.nano	2	6	0.5	EBS-Only	Up to 5

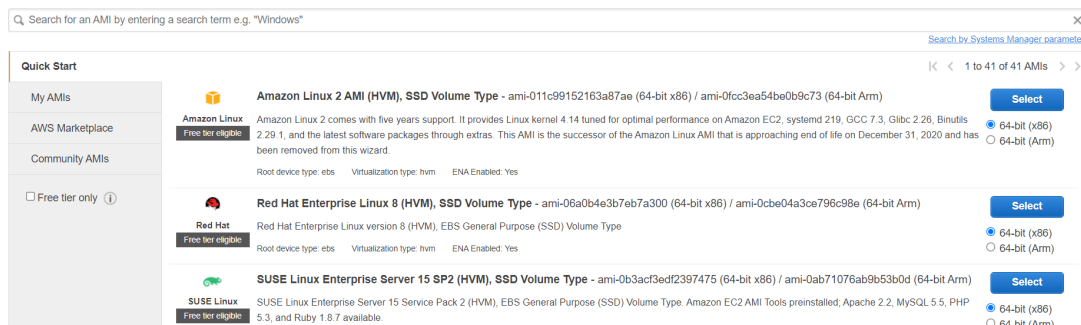
The above diagram explains how an EC2 instance is described.

Even the big companies like Facebook, Netflix use these EC2 instances for their storage and computational purposes.

AWS is the only cloud provider that supports macOS.

7) What are the steps involved in creating an EC2 instance?

STEP 1 Choose an AMI (amazon machine images): an AMI is a software configuration framework that contains operating system, application server, applications required to create your instance.



STEP 2 selecting an instance type: these instances are like the virtual servers that run your application. In this step you choose constraints like CPU, memory, Instance Storage, Network Performance, EBS-Optimized availability.

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, ~, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="radio"/>	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="radio"/>	t2	t2.micro <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate	Yes
<input type="radio"/>	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes

We only have one type for free rest types are charged based on the constraints they provide.

STEP 3 configure instance: mainly in this section we select number of instances needed, we can launch multiple instances from same AMI

Number of instances ⓘ [Launch into Auto Scaling Group ⓘ](#)

Purchasing option ⓘ ☐ Request Spot instances

Network ⓘ [Create new VPC](#)

Subnet ⓘ [Create new subnet](#)

Auto-assign Public IP ⓘ

Placement group ⓘ ☐ Add instance to placement group

Capacity Reservation ⓘ

Domain join directory ⓘ [Create new directory](#)

IAM role ⓘ [Create new IAM role](#)

Shutdown behavior ⓘ

Stop - Hibernate behavior ⓘ ☐ Enable hibernation as an additional stop behavior

Enable termination protection ⓘ ☐ Protect against accidental termination

STEP 4 Add Storage: This step deals with storage device setting. AWS offers 30 GB default. We can extend this by adding additional EBS volume.

Volume Type ⓘ	Device ⓘ	Snapshot ⓘ	Size (GiB) ⓘ	Volume Type ⓘ	IOPS ⓘ	Throughput (MB/s) ⓘ	Delete on Termination ⓘ	Encryption ⓘ
Root	/dev/sda1	snap-0feaf82256fa4c01f	<input type="text" value="30"/>	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

STEP 5 Tags: This is mainly helpful when you have resources of the same type that help you categorize these resources. It can also be used as a meta data to each resource.

Each **Tag** consists of a key and an optional value, both need to be defined by the user.

Key (128 characters maximum)	Value (256 characters maximum)	Instances ⓘ	Volumes ⓘ	Network Interfaces ⓘ
<input type="text" value="hero"/>	<input type="text" value="12"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)

STEP 6 Configure security group: In this step we frame firewall rules that control the traffic for our E2 instance.

Assign a security group: ☒ Create a new security group ☐ Select an existing security group

Security group name:

Description:

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
RDP	TCP	3389	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop


[Add Rule](#)

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.


STEP 7 review: final step, this stage you are to check all the previous decisions or settings made all along this way.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

 **Improve your instances' security.** Your security group, launch-wizard-1, is open to the world.
Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only.
You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

▼ AMI Details

 **Microsoft Windows Server 2019 Base with Containers - ami-0320b895e6e18ca3**

Free tier eligible

Microsoft Windows 2019 Datacenter edition with Containers. [English]
Root Device Type: ebs Virtualization type: hvm
If you plan to use this AMI for an application that benefits from Microsoft License Mobility, fill out the [License Mobility Form](#). [Don't show me this again](#)

Edit AMI

▼ Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

Edit instance type

▼ Security Groups

Edit security groups

Cancel

Previous

Launch

For the choice I made for 6 steps this is how my review page looked like.

8) What are restrictions of a tag for EC2 instances?

There are certain rules that need to be followed in creating a tag for an EC2 instance.

- For an allocated instance maximum number of tags allowed are 50.
- All the tags under one instance need to be unique and each key can only have one value.
- Coming to the storage constraints of key and value pairs:
Key: maximum length -128 Unicode characters in UTF-8
Value: max value length -256 Unicode characters in UTF-8
- Tags and values are case sensitive.
- If we have “aws:” prefix then we cannot edit key or value and these tags do not come under the maximum tag limit.

9) What is the difference between terminating and stopping an EC2 instance?

Intuitively it might seem that both are similar but there is a huge difference between these two terms when referred under EC2 instance. When we stop an EC2 instance, it performs a normal shutdown and moves to a stopped state.

However, when you terminate an EC2 instance, it is first transferred to a stopped state, and the EBS volume attached to it is deleted and cannot be recovered.

10) What does Elastic Load Balancing in AWS do?

Previously we learnt that EC2 instances are responsible for having a virtual machine with all features to a user. Now, the question is what makes this happen, part is responsible for efficient allocation of specific instances.

ELB (elastic load balance) distributes the traffic into various EC2 instances across available resources.

ELB comes in 4 different types that takes all the important features into consideration like high availability, automatic scaling, robust security.

ELB types:

- Application load balancer
- Gateway load balancer
- Network load balancer
- Classic load balancer

It can dynamically change its growth or shrink based on the traffic requirement.

11) What is S3 in AWS?

S3 -simple storage service. S3 is basically a storage space used in AWS to store data. Amazon S3 provides a basic storage interface in the cloud to retrieve and store practically any amount of data.

S3 stores in storage resources called buckets (they are like the storage tankers in AWS).

12) What are the advantages of AWS S3?

- Buckets: storage resources in S3 are called buckets, they act as storage facility units in S3.
- Data storage: we can upload as much data as we want, practically infinite but each object contains 5TB of data. And for security reasons each time an action is performed, AWS expects to enter the assigned key to perform the function (data retrieval or uploading).
- Downloading data: no interruption for the free flow of downloading data at any time.
- Permissions: they are strict security protocols for data retrieval and data uploading. Grant upload or download are of 3 types to the users.

- Standard interfaces: Using standards-based REST and SOAP interfaces allows to work with any internet-development toolkit.

13) What are the various attributes of AWS storage objects?

- **Key**: it is the name of the object in the bucket. Each object in a bucket has a unique key.
- **Value**: it is the data we store in buckets.
- **Version Id**: used for version control.
- **Meta data**: it is data about the data (like what are the datatypes of the data or frontline information about the data)
- **Permissions or Access control lists**: giving permissions to who can have access to the data (retravels or download).

Some of these attributes like bucket + key + version id can be used to map uniquely to object.

14) What are the important factors considered while selecting a S3 storage class?

Some of the factors could be namely:

- Based on frequency of accessing data
- Automatically optimizing with changing or unknown access patterns
- Based on infrequently accessing data objects
- For archiving objects

15) What are the popular or frequently used storage classes?

1. Amazon S3 standard known as **S3 standard**: It is the default storage class you get when you do not specify any class.

It is used for wide variety of purposes such as:

- Content distribution
- Gaming applications
- Big data analytics

2. **S3 standard-infrequent access**:

This is specifically used when you do not access the data frequently but when you access you need a high retrieval rate.

It is used for long term file storage, when you need to sync old data with present data, disaster recovery file.

3. Reduced Redundancy storage:

This stores data at low redundancy levels than S3 standard or infrequent access.

It is mainly used to store non critical data (data without which the business can survive).

It does not replicate objects as many times as standard S3.

4. Amazon S3 glacier:

This class is used for those who prioritize security. when the data involves competitive secrets or something like that glacier is preferred.

To keep the cost low with all these features, you get 3 retrieval options that range from a few minutes to hours.

It is used mainly for data archiving.

16) What are the main features of S3 standard storage class?

feature	S3 standard
Designed for durability	99.999999..%(11 9 s)
Designed for availability	99.9%
Availability SLA (service level agreement)	99.9%
zones	≥3
First byte latency	Milli seconds
Retrieval fee	None

17) Show Step by step guide in creating a S3 bucket?

Initially, when we enter the console and go towards creating a bucket. We get to see something like this:

Create bucket

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name

Bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

AWS Region

Asia Pacific (Mumbai) ap-south-1

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

We gave a name and region, choose the nearest region so that retrieval rate could be faster.

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning

☒ Disable

☐ Enable

Tags (0) - optional

Track storage cost or other criteria by tagging your bucket. [Learn more](#)

No tags associated with this bucket.

Add tag

Default encryption

Automatically encrypt new objects stored in this bucket. [Learn more](#)

Server-side encryption

☒ Disable

These are some of the other credentials you need to choose before creating an actual bucket.

After you create a bucket, you see something like this on your dashboard:

Amazon S3

Account snapshot

Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

View Storage Lens dashboard

Buckets (1)

Copy ARN

Empty

Delete

Create bucket

Buckets are containers for data stored in S3. [Learn more](#)

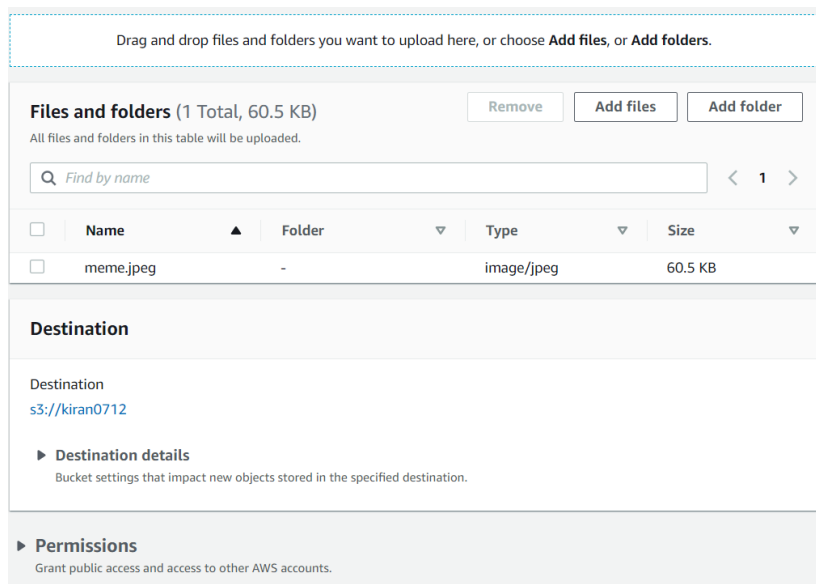
Q kiran

1 match

< 1 > ⚙

	Name	AWS Region	Access	Creation date
<input type="radio"/>	kiran0712	Asia Pacific (Mumbai) ap-south-1	Bucket and objects not public	June 23, 2021, 12:17:38 (UTC+05:30)

Now I can upload my data to the bucket.



See here I have added an image into my bucket(meme.jpeg)

It also displays destinations.

Choose the required storage class while uploading the data because you cannot change the class once it is uploaded.

	Storage class	Designed for	Availability Zones	Min storage duration	
<input checked="" type="radio"/>	Standard	Frequently accessed data	≥ 3	-	-
<input type="radio"/>	Intelligent-Tiering	Long-lived data with changing or unknown access patterns	≥ 3	30 days	-
<input type="radio"/>	Standard-IA	Long-lived, infrequently accessed data	≥ 3	30 days	1
<input type="radio"/>	One Zone-IA	Long-lived, infrequently accessed, non-critical data	1	30 days	1
<input type="radio"/>	Glacier	Long-term data archiving with retrieval times ranging from minutes to hours	≥ 3	90 days	-
<input type="radio"/>	Glacier Deep Archive	Long-term data archiving with retrieval times within 12 hours	≥ 3	180 days	-
<input type="radio"/>	Reduced redundancy	Frequently accessed, non-critical data	≥ 3	-	-

Previously we discussed only a few classes but there are other classes too.

If you want to deep dive into the storage classes and its features:

<https://aws.amazon.com/s3/storage-classes/>

follow this link, it's from the official website.

Tags - *optional*
Track storage cost or other criteria by tagging your objects. [Learn more](#)

No tags associated with this resource.

Add tag

Metadata - *optional*
Metadata is optional information provided as a name-value (key-value) pair. [Learn more](#)

No metadata associated with this resource.

Add metadata

You can also add metadata and tags to your buckets.

18) What are all the configurations offered to you when you create a S3 bucket?

- Access control lists (ACL): read, write permissions for object owners and others.
- Storage classes: default standard, you can alter it while creating a bucket.
Remember: you cannot alter the bucket name or region after creating the bucket.
- Server-side encryption
- Tags
- Version
- Meta data: information about the data you stored in the bucket.

19) Is S3 a global service? If yes, why would we still mention the region name while creating a bucket?

The naming schema is global but data storage and S3 runs in each AWS region. you are suggested to choose the region that is near to you considering the factors like latency rate, minimal costs, address regulatory requirements.

20) What is the maximum number of buckets you can create in an account?

Inherently, a user with an account is allowed to have 100 buckets. If you need additional buckets, it can be increased to up to 1000 buckets by

submitting a server limit increase. However, you can create any number of objects in the bucket.

One point to note is that users will not experience any slow down because of multiple buckets. Slowdown is independent of the number of buckets in an account.

21) What is versioning in S3 bucket? What is the purpose of using versioning in S3 bucket?


Versioning is what allows you to keep multiple varied copies of the data objects in a bucket. Versioning allows you to recover a past data which you have now overwritten.

22) What are the different states of versioning in S3 buckets?



Bucket versioning comes in 3 types:

- Unversioned (default state)
- Versioning enabled
- Versioning suspended

Initially you are in the version disable state and you can change it to enable after creating the bucket but once you click enable at creating the bucket you cannot return to the unversioned state. However, you can suspend it. For each version the user is provided with a unique name.

<input type="checkbox"/>	Version ID	Type	Last modified	Size	Storage class
<input type="checkbox"/>	 null (Current version)	jpeg	June 23, 2021, 12:29:31 (UTC+05:30)	60.5 KB	Standard

When you are in an unversioned state, your version id would be null.

<input type="checkbox"/>	Name	Type	Version ID	Last modified	Size	Storage class
<input type="checkbox"/>	 meme.jpeg	jpeg	DH69L1ydJCIDqyo1JHoy8u1CLJQxsZzJ	June 23, 2021, 18:33:00 (UTC+05:30)	60.5 KB	Standard
<input type="checkbox"/>	 meme.jpeg	jpeg	null	June 23, 2021, 12:29:31 (UTC+05:30)	60.5 KB	Standard

I uploaded the same file again, now it appears as a version and you also get a version id.

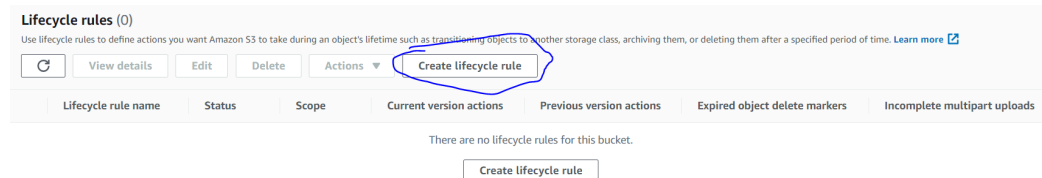
23) What is the life cycle in S3 bucket?

Life cycle in S3 is a service provided by AWS where you get to choose the path your data travels over the time. Path here means the storage class it enters after specific number of days automatically without you changing it and it also controls how versions should react with change in storage class over time like should all the previous versions be deleted when you change the class type or should all of them change the storage

class and many other options regarding version adaptability with classes over time.

24) How to create a life cycle for a file in S3?

Firstly, go through the management section and you see an icon named “create life cycle”.



After clicking this, you will see set of arguments you can set for life cycle lets see them one by one:

Lifecycle rule configuration

Lifecycle rule name

Up to 255 characters.

Choose a rule scope

☒ Limit the scope of this rule using one or more filters

☐ This rule applies to *all* objects in the bucket

Filter type

You can filter objects by prefix, object tags, or a combination of both.

Prefix

Add filter to limit the scope of this rule to a single prefix.

Don't include the bucket name in the prefix. Using certain characters in key names can cause prob protocols.

You can also create a cycle for a file or object in a bucket. Life cycle need not be for all the files in the bucket.

Lifecycle rule actions

Choose the actions you want this rule to perform. Per-request fees apply. [Learn more](#) or see [Amazon S3 pricing](#)

- ☐ Transition *current* versions of objects between storage classes
- ☐ Transition *previous* versions of objects between storage classes
- ☐ Expire *current* versions of objects
- ☐ Permanently delete *previous* versions of objects
- ☐ Delete expired delete markers or incomplete multipart uploads

When a lifecycle rule is scoped with tags, these actions are unavailable.

Previously, we said in the Life cycle, we get to control the change of version controls with respect to storage classes. The above is how the options are displayed to us for that specific purpose.

Now we move to the main feature of the life cycle. The reason why we use the life cycle process in the first place, Here we get to choose what storage class you need after a certain time interval.

Observe in the below picture, I intentionally entered the days as 29 to show how it raises objection, below you see it says, you need a minimum of 30 days

There are certain rules for storage classes which specify a certain amount of days you need to spend at least in any other class in order to enter that class. For example: the standard-IA selected here needs us to be at standard S3 for at least 30 days.

Transition current versions of objects between storage classes

Storage class transitions	Days after object creation	
Standard-IA	29	Remove transition

A minimum of 30 days is required before transitioning to Standard-IA.

Transition current versions of objects between storage classes

Storage class transitions	Days after object creation	
Standard-IA	31	Remove transition
Glacier Deep Archive	182	Remove transition

Add transition

This is how you proceed: you give continuous classes with time specified for each and create a route map for your objects which indeed is the life cycle of your bucket.

25) What are some of the data protection services provided by AWS?

- **Data redundancy:** amazon has multiple storage facilities around the world which provides high durability for mission critical storage data. The bucket objects are redundantly stored across multiple storage facilities in amazon s3 regions.
- **Versioning:** amazon provides you versioning. Users can use it to store every version of every object and retrieve all objects with the same latency.

- **Server-side encryption:** AWS kms services provides you key for data encryption.
- **Client-side encryption:** In this case, client or user is responsible for encrypting data and AWS just stores your encrypted data.

We have 2 ways to proceed with client-side encryption:

- 1) Creating a customer master key and storing it in AWS key management service.
- 2) Storing the key within the application.

26) What is encryption?

Encryption is a process where you convert the human readable text to some complicated incomprehensible text to secure your data. This incomprehensible data is also called cipher text. To encrypt data, you need an encryption key, key is nothing but a logic or a set of mathematical values which can convert your normal data to cipher text. In AWS you can have an encryption key to your bucket objects by using the service AWS KMS (key management system). Let's know more about it as we go further into the article.

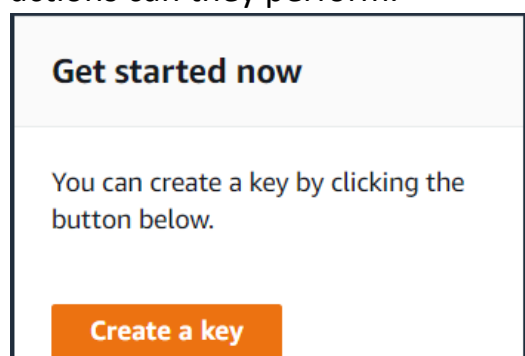
27) What is data encryption server-side security service provided by AWS?

KMS-Key Management Service is a server-side data encryption service provided by AWS.

There is 2 level of encryption in KMS

First the data is encrypted by a key provided by kms and then at level 2 the key is also encrypted using the master key which rotates regularly.

We can also bring our own keys and import them in KMS. We can also set usage policies that determine which users can use our keys and what actions can they perform.



After we click on the create key. We need to choose between 2 types of keys:

- 1) **Symmetric**: single key for both encryption and decryption.
- 2) **Asymmetric**: A more secure way than the previous key type, here we will have 2 types of keys: public and a private one for encrypt/decrypt or sign/verify options.

Configure key

Key type [Help me choose](#)

☒ **Symmetric**
A single encryption key that is used for both encrypt and decrypt operations

☐ **Asymmetric**
A public and private key pair that can be used for encrypt/decrypt or sign/verify operations

Later in advanced settings, you are also provided to choose between regionality. This allows you to either have the same key in every region or a different key in different regions.

At the next stage you select key administrative permissions.

Step 1
Configure key

Step 2
Add labels

Step 3
Define key administrative permissions

Step 4
Define key usage permissions

Step 5
Review

Define key administrative permissions

Key administrators
Choose the IAM users and roles who can administer this key through the KMS API. You may need to add additional permissions for the users or roles to administer this key from this console. [Learn more](#)

☒

AWSServiceRoleForSupport

/aws-service-role/support.amazonaws.com/

Role

☐

AWSServiceRoleForTrustedAdvisor

/aws-service-role/trustedadvisor.amazonaws.com/

Role

Key deletion

☒ Allow key administrators to delete this key.

Cancel

Previous

Next

After you finish all the steps, you can see that the key is created

Customer managed keys (1)						Key actions	Create key
<input type="text"/>						< 1 >	
<input type="checkbox"/>	Aliases	Key ID	Status	Key spec	Key usage		
<input type="checkbox"/>	keys_123	6827a6b1-02ec-462b-b13d-aa8b883beb0b	Enabled	SYMMETRIC_DEFAULT	Encrypt and decrypt		

28) What is the default view for an S3 bucket?

Initially, when you create an s3 instance all the objects, buckets and sub resources like website configuration and life cycle configuration are set to private. Only the resource owner has the access to it by default. However, if we wish to give access to other users, it is in the hands of the resource owner who can set access permissions by writing an access policy.

29) What do we mean by access policies in S3? What are the types of access policies?

Initially buckets are in private, if we want others to access it, first, we need to set the bucket to public and then set the objects in it to public for all the users to see but if there are like 20,000 files in it and we can not go making each file public all day, it is a tedious task. So AWS allows us to give some policies to attach to our buckets which gives public mode to all the objects in the bucket at a time, not just this there are many other use cases or features for access policies.

Mainly there are 2 types of access policies:

- Resource-based policies
- User based policies

Bucket policies and access control lists come under resource-based policies. These are attached to buckets hence the name resource-based policies.

We do have an option of giving policies to user accounts instead of resources. These are called user policies. We can create policies combining both resource and access policies.

30) What is auto-scaling with respect to AWS?

Auto-scaling is a function that provides access to new instances whenever there is a serious demand for extension or decrement of resources based on demand. This saves a lot of money for users who have flexible, dynamic data stored in AWS.

31) What happens when you just created a bucket and added 2 files and you wish to change one of the objects to public access?

You get an error message. If you wish to change an object to public the error message says, you first need to change bucket permissions.

See my bucket is not public, now I try to change the object access settings and see what happens

Buckets (1) Refresh Copy ARN Empty Delete Create bucket

Buckets are containers for data stored in S3. [Learn more](#)

Find buckets by name

Name	AWS Region	Access	Creation date
kiran0712	Asia Pacific (Mumbai) ap-south-1	Bucket and objects not public	June 23, 2021, 12:17:38 (UTC+05:30)

This is how the error message looks like, when you wish to change the object settings.

Failed to edit public access
For more information, see the Error column in the Failed to edit table below.

Make public: status Close

Public access is blocked because **Block Public Access** settings are turned on for this bucket
To determine which settings are turned on, check your [Block Public Access settings](#) for this bucket. [Learn more about using Amazon S3 Block Public Access](#)

The information below will no longer be available after you navigate away from this page.

Summary

Source s3://kiran0712	Successfully edited public access 0 objects	Failed to edit public access 1 object, 43.3 KB
--------------------------	--	---

32) What are the block public access settings available for a bucket?

This is how the default setting looks like, initially, all objects are refrained from public access.

Block all public access

On

- Block public access to buckets and objects granted through *new* access control lists (ACLs)

On
- Block public access to buckets and objects granted through *any* access control lists (ACLs)

On
- Block public access to buckets and objects granted through *new* public bucket or access point policies

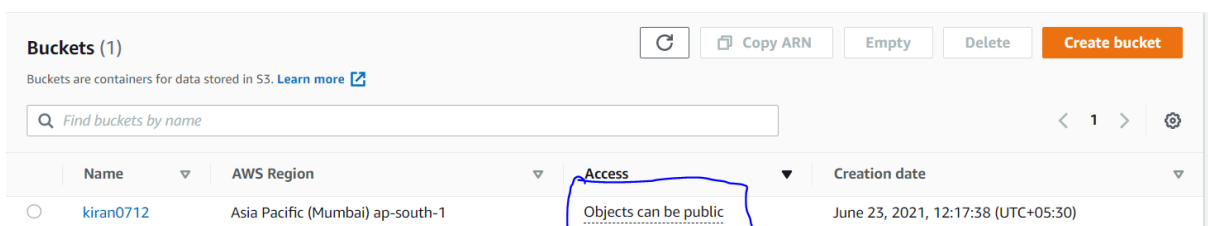
On
- Block public and cross-account access to buckets and objects through *any* public bucket or access point policies

On

After we turn off the “block all public access”. We can select the access among the 4 options.

- Block public access to buckets and objects granted through *new* access control lists (ACLs):** If you enable this setting, then all the previous files that are public or private stays the same way but from now on any new file will be refrained from public access.

- 2) **Block public access to buckets and objects granted through *any* access control lists (ACLs):** here S3 will ignore all the previous and later ACLs which grant public access.
- 3) **Block public access to buckets and objects granted through *new* public bucket or access point policies:** S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- 4) **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies:** This is mainly to prevent public access from cross-account access.



The screenshot shows the AWS S3 'Buckets' page. At the top, there are buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'. Below these is a search bar labeled 'Find buckets by name'. A table lists the buckets. The first bucket is 'kiran0712' in the 'Asia Pacific (Mumbai) ap-south-1' region, created on 'June 23, 2021, 12:17:38 (UTC+05:30)'. The 'Access' column for this bucket shows 'Objects can be public', which is circled in blue.

	Name	AWS Region	Access	Creation date
<input type="radio"/>	kiran0712	Asia Pacific (Mumbai) ap-south-1	Objects can be public	June 23, 2021, 12:17:38 (UTC+05:30)

Now I edited bucket permissions and it is reflected on the main page.

33) What is Amazon Sage Maker?

Amazon sage maker is an AWS service, which helps data scientists and other data professionals to build, train and deploy machine learning models and monitor their performance later on. Since this is cloud computing, the user need not worry about his system computational limitations.

34) What are key steps in Sage Maker?

The three steps while utilizing sage maker resources are build, train and deploy.

Build: This service allows you to build machine learning models varying from linear regression with 10 mb dataset to advanced neural networks algorithm with Gb's of data.

AWS hosts these services using Jupyter notebooks or Jupyter labs. Which are just like the ones we use in our system with anaconda.

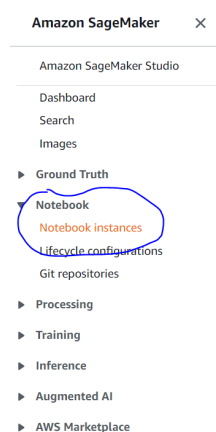
We can create notebooks which already have most of the algorithms pre-installed, which are optimized so that they can perform 10 times faster than when we run the algorithm elsewhere.

Train: clicking on run, with having the code lines to generate a model takes a few seconds to train a model. In order to speed and simplify the training process, it can automatically tune the model to get best accuracy.

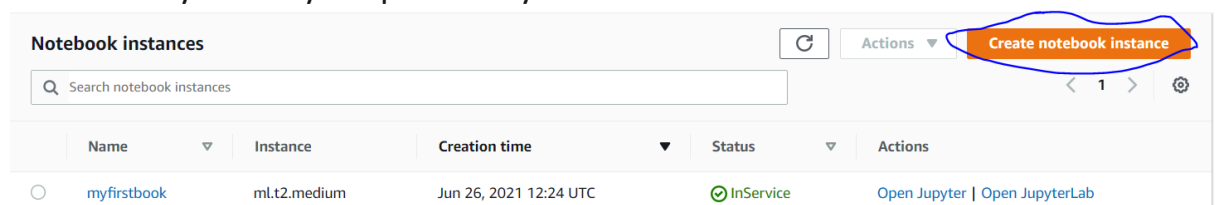
Deploy: After we build, tune and train the model it is time to bring it to actual production. The model created by us can be hosted on Amazon EC2 available in multiple zones for high performance and availability. AWS KMS-based Encryption is also available in this service to help in training and deploying.

35) Create a notebook instance.

When you enter Amazon Sage Maker services dashboard on the left side you see these options select notebook instance.



After you do that, you see a create notebook instance on the right side and below you see your previously created instances.



“myfirstbook” was already created so it appears below. Observe the status it is in “inService”.

Next, you enter instance settings that you need to set like notebook name, instance type(ml.t2.medium comes under free tier so we use this for now).We can also set network, tags, Permissions and encryptions.

Create notebook instance

Amazon SageMaker provides pre-built fully managed notebook instances that run Jupyter notebooks. The notebook instances include example code for common model training and hosting exercises. [Learn more](#)

Notebook instance settings

Notebook instance name

Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region.

Notebook instance type

► Additional configuration

Initially after you select it takes few seconds to be active, till then, your status would be pending. Observe the “mysecondnotebook” status after it got completed you see Inservice.

Notebook instances					
<input type="text" value="Search notebook instances"/>					
	Name	Instance	Creation time	Status	Actions
<input type="radio"/>	mysecondnotebook	ml.t2.medium	Jun 26, 2021 13:06 UTC	Pending	-
<input type="radio"/>	myfirstbook	ml.t2.medium	Jun 26, 2021 12:24 UTC	InService	Open Jupyter Open JupyterLab

Now you can click on Open Jupyter and start coding in python.

36) How to Create an S3 bucket from a Jupyter notebook instance?

After we enter the Open Jupyter follow me with this code

```
'''import necessary libraries'''
```

Boto3 is used for creating s3 buckets, sagemaker contains all the ml algorithms.

In []:

```
import boto3
```

```
import sagemaker
```

```
from sagemaker.session import s3_input, Session
```

```
'''we write name for bucket and find out our default region '''
```

In [12]:

```
bucket_name='dataset0712'
my_region=boto3.session.Session().region_name
print(my_region)

(output)
ap-south-1
```

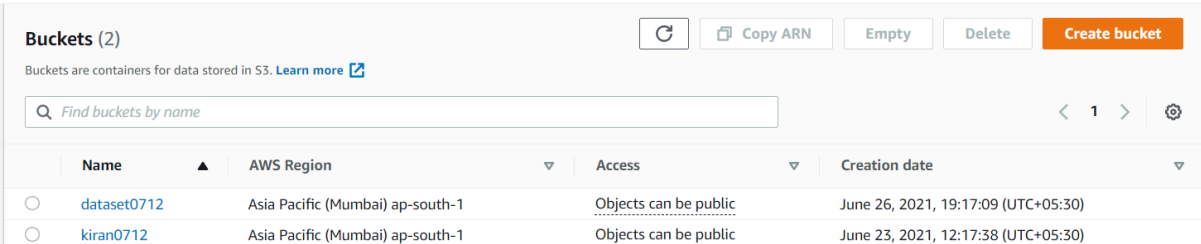
In [13]:

```
""" now we actually create an s3 bucket from jupyter"""
s3=boto3.resource('s3')
try:
    if my_region== 'ap-south-1':

s3.create_bucket(Bucket=bucket_name,CreateBucketConfiguration={'Location
Constraint':'ap-south-1'})#this step creates an s3 bucket
    print('bucket successfully created')
except Exception as e:
    print('S3 error: ',e)
```

(output)
bucket successfully created

changes are reflected in the bucket list



The screenshot shows the AWS S3 Buckets console. At the top, there's a header 'Buckets (2)' with a refresh button, a 'Copy ARN' button, an 'Empty' button, a 'Delete' button, and a prominent orange 'Create bucket' button. Below the header, a search bar says 'Find buckets by name'. A pagination bar shows '< 1 >' and a settings gear icon. The main content is a table with columns: Name, AWS Region, Access, and Creation date. Two buckets are listed: 'dataset0712' and 'kiran0712', both in the 'Asia Pacific (Mumbai) ap-south-1' region with 'Objects can be public' access.

	Name ▲	AWS Region ▼	Access ▼	Creation date ▼
<input type="radio"/>	dataset0712	Asia Pacific (Mumbai) ap-south-1	Objects can be public	June 26, 2021, 19:17:09 (UTC+05:30)
<input type="radio"/>	kiran0712	Asia Pacific (Mumbai) ap-south-1	Objects can be public	June 23, 2021, 12:17:38 (UTC+05:30)

After you run this code, go back to AWS s3 and see the updated list of buckets, you find a bucket that you named in your program (here I named it as “datasert0712”).

37) What is RDS?

AWS RDS -relational database service. It is an AWS database service which makes it easy to set up and operate the huge cloud data present in AWS.

Some of the main features of RDS are as follows:

- Having a server, yourself, you are restricted by fixed amounts of CPU, memory, storage, IOPS (input output operation per second) This sometimes or most of time be a problem because some websites need high IOPS but can manage with low storage for ex: a university exam conducting software takes less memory (for question papers and login credentials for students in campus) but it cannot afford for having low IOPS because then students cannot write exam comfortably. Considering situations like this in RDS each constraint is independent and it can be chosen individually
- Amazon RDS manages backups, software patching, automatic failure detection, and recovery.
- RDS does not provide shell access to DB instances.
- RDS allows users to use database products that are already present in the market like: MySQL, MariaDB, PostgreSQL, Oracle, Microsoft SQL Server.
- We can use IAM (identity and access management) to set permissions or define access to the database.

38) What are DB instances?

DB instances are basic elementary units or building blocks of Amazon RDS. Users can modify or create a DB instance using the AWS Command Line interface, RDS, Management console. Every Db instance runs a DB instance. Every Db engine has some constraints or parameters that monitor the databases that it is responsible for.

DB instances are of 3 types:

- Magnetic
- General purpose (SSD)
- Provisional IOPS

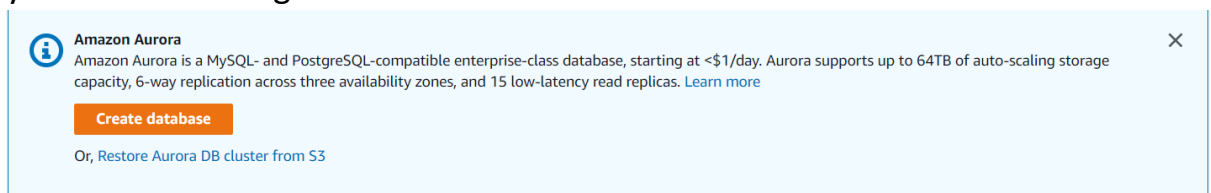
These categories are based on performance characteristics and prices. DB instances will have a particular maximum and minimum storage requirements depending on the storage types and the DB engine it belongs to.

39) How to reduce setting so many configurations while creating a DB instance?

In order to create a DB instance first we have to enable “easy create” so that some configurations are set to default otherwise we need to set many configurations.

40) How to create a Database in AWS?

When you login and enter the console, search for RDS and going further you see something like the below



Clicking on the RDS you see several parameters or settings for your database.

Initially, you were asked to choose between two creation method

1) Standard create:

In this we are supposed to select all the configuration options like availability, backup service, security and maintenance. This can be tedious if you want to set up a simple database service. So AWS provides another creation method.

2) **Easy create:** it sets most of the configurations for you and gives the flexibility to change after the database is created.

After you select the creation method. you now choose engine options.

- Amazon Azure
- MySQL
- MarioDB
- PostgreSQL
- Oracle
- Microsoft SQL server

Now, we move on to Capacity type

- Provisioned: In this user manages the server instance and its sizes.
- Serverless: this is preferred if the traffic is unpredictable to the user. User specifies maximum and minimum resources needed.

Next, Templates

- Production utility conditions: fast, consistent performance, higher availability.

- Dev/Test: used for non-production environments.
- Free Tier: you can use up to 20 GB of storage for database service.

Then we move on to selecting Db instance class

db.r5.large	2 vCPUs	16 GiB RAM	Network: 4,750 Mbps
db.r5.xlarge	4 vCPUs	32 GiB RAM	Network: 4,750 Mbps
db.r5.2xlarge	8 vCPUs	64 GiB RAM	Network: 4,750 Mbps
db.r5.4xlarge	16 vCPUs	128 GiB RAM	Network: 4,750 Mbps
db.r5.8xlarge	32 vCPUs	256 GiB RAM	Network: 6,800 Mbps
db.r5.12xlarge	48 vCPUs	384 GiB RAM	Network: 9,500 Mbps
db.r5.16xlarge	64 vCPUs	512 GiB RAM	Network: 13,600 Mbps

These are some of the instance types provided by AWS.

Finally, we move on to connectivity and database authentication.

Here I covered all the main configurations, there are many others if you go deep. For example: if you choose to enable auto scaling, even if you purchase for like 100Gb and use only 40 GB, you are charged only for 40 GB.

Other settings could be enabling automatic OS updates or patches and provision of data, backup retention period, setting logs for database changes (update, delete or add).

41) What is a database backup snapshot?

Creating a RDS storage volume and backing up the entire Db instance instead of just individual databases is called database snapshot.

This process can also be done manually, where you create a DB snapshot. The initial snapshot user creates contain the whole database instance, while the next subsequent or following snapshots are cumulative or per se incremental, so the changed parts are only visible in the latest snapshots.

One advantage of snapshots over automated backup is they won't expire like the automated backup.

42) What are automated backups?

RDS gives users this flexibility of creating and saving automated backup instances during the backup window of your DB instance.

Points to remember while enabling automated backups

- The DB instance for it should be in AVAILABLE state for automated backups.

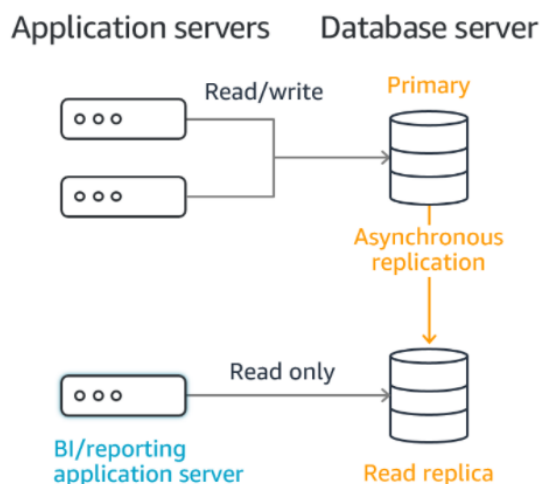
- When a copy is already running in the same region for the same DB instance, we cannot automate backups.
- Set up data backup retention period to non-zero number in order to enable it.

43) What is an amazon read replica?

There are 2 types of storages in the RDS database system.

- 1) Master or primary: which has read/write access
- 2) Read replica: these are mere asynchronous replication copies, which has read only access from application servers.

Read replica improves performance since by adding these elements, it reduces load on main storage by routing the queries to the read replica. For even further enhancing the performance RDS for just MYSQL provides additional table indexes without actually being in master. They can also be promoted to master status. Users can also create read replicas with encrypting data using AWS key management service.

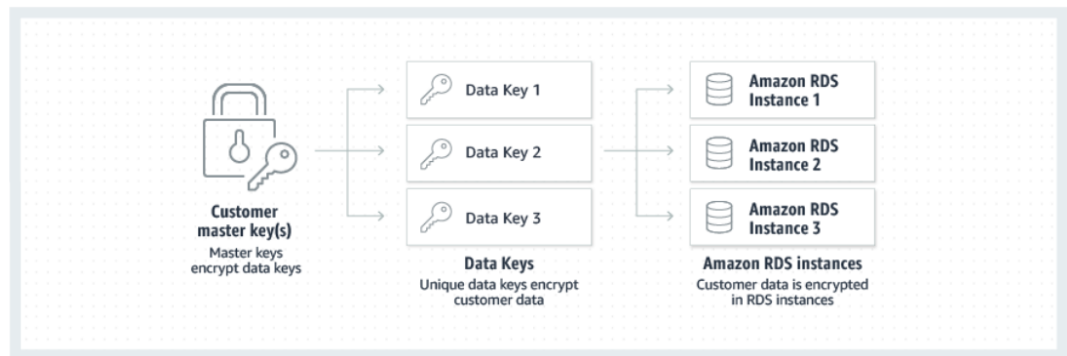


44) What are the security actions taken by AWS RDS?

There are mainly 3 services provided for security regarding AWS

- Amazon Aurora or RDS provides your database the service to run it on Virtual Private Cloud (VPC) for network level isolation.
- Creating security groups to provide permissions that determine who can access the database. This built-in firewall prevents any database except through rules users specify.

- Using Secure Socket Layer/ transport layer security connections to encrypt the data in transit. Kms key management services can be used to encrypt the databases

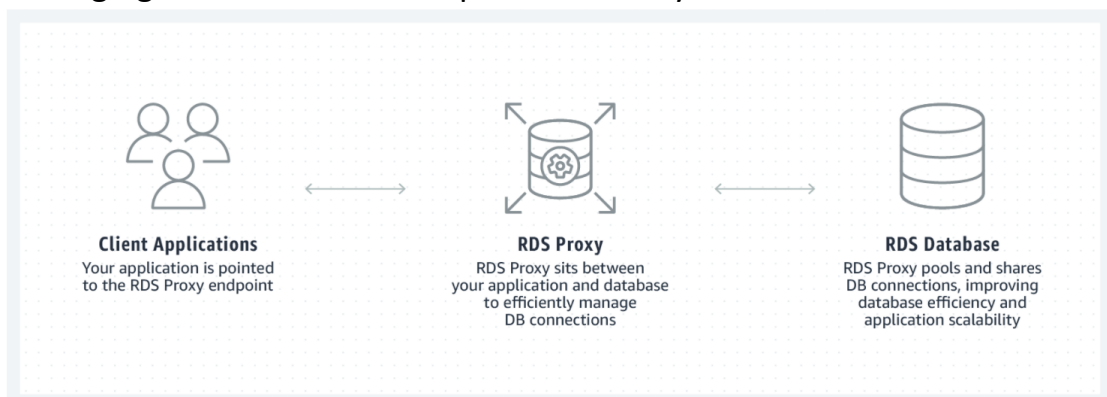


45) What is an Amazon RDS proxy?

Large databases used by companies often need a large number of open connections to the database server to run their day-to-day operations. These databases open and close connections at a high rate, exhausting database resources, memory.

To solve this issue RDS came up with proxy services. Amazon RDS proxy allows applications to pool and share connections established with the database, thus improving database efficiency and application stability. RDS proxy reduces fail-over times upto 66%. The price you pay for per vCPU includes proxy services. Amazon RDS Proxy is available for Aurora MySQL, Aurora PostgreSQL, RDS MySQL and RDS PostgreSQL.

RDS proxy acts as a mediator between application and database managing connections and improves stability of connections.



46) What is the advantage of Amazon Cloud Watch?

In operating real time websites with millions or thousands of viewers it is important to have constant updates regarding website metrics like conversion rate, click through, Search bar using frequency etc.

Amazon Cloud watch helps to serve this purpose. It helps you to monitor.

- Schedule events for database updating or resource extension
- Auto-scaling lifecycle events
- AWS API calls
- Console sign-in events
- State changes in Amazon EC2

47) What are region-specific AWS services?

It is important for account users to be aware of this otherwise if they fail to set the region to local region accessing latency might increase or might face extra charges in some instances.

- IAM
- CloudFront
- Route 53
- Web application Firewall

48) How do we configure CloudWatch to recover EC2 instance?

Step1: create an alarm using Amazon CloudWatch

Step2: In the alarm go to define alarm-> Action Tab

Step3: select the recover instance option in the action tab.

49) Key differences between S3 and EBS

Feature	AWS S3	AWS EBS
Paradigm	Object store system	File store system
Performance	Fast	Superfast
Redundancy	Across data centres	Within a single data centre

Security	Using private or public key	Can be used only with EC2
----------	-----------------------------	---------------------------

50) What are the different types of load balancers in AWS?

There are 3 types of load balancers that are supported by Elastic load Balancing.

- Application load balancer
- Network load balancer
- Classic load balancer

Summary

Initially, we started our discussion with, what is cloud computing, what services do AWS provide, then we saw the key elements from the basic architecture diagram like EC2 instance, buckets, security groups.

Now, we have moved our interest to specific services. Firstly, we learned about EC2 instances and saw how to create one and the restrictions on tags while creating an EC2 instance.

Next, we move on to the main service element that is S3 bucket. We understand the importance of the service, know about storage classes and when to choose a particular storage class and what are features involved in a storage class, create a s3 bucket using free s3 standard storage type, further we move on to the main aspects of s3 like life cycle how to set a life cycle for a s3 bucket.

As we move on further, we see about data protection services provided by AWS. Next, we see permission settings for s3 buckets and complications involved in it.

The next service element we covered is sage maker, used typically for machine learning modelling and deployment. As an exercise we created a Jupyter notebook instance and saw how to create a s3 bucket using python and boto3 module.

Final element we cover in this article is RDS relational database service. We see the backup arrangements provided and know about creating a database instance and some of the key features like proxy, security groups with respect to RDS are also covered.