**EBS:**

* It is a virtual hard drive that we can use with ec2.
* We use it because it allows us to keep the data separate from our computing instance.
* They are independent and we attach it to instance when needed.
* After attaching it to the instance we can use it like any other hard drive and store any kind of file system.
* When a compute instance fails no need to worry because it is housed on your block storage.
* They are around 1 gb to 1 tb in size.
* Data is replicated in that zone to avoid data loss due to failure of any single hardware component.
* Many volumes can be attached to a single instance.
* We can attach multiple volumes and increase input and throughput performance.
* Mainly helpful for data based intensive applications that frequently encounter many random read’s and writes across the dataset.

**Example of using EBS:**

If a web application we created is becoming a huge hit and is serving more than 2 million pages per day as well as processing 4 million transactions every month.

With this kind of demand we make sure that we have great performance for all of visitors. We use amazon EBS volumes and since the workload is IO-intensive you can leverage provisioned IOPS to significantly reduce latency and improve your visitors experience on your site.

**Create an EBS volume and attach it to a already running ec2 instance:**

Clink on ec2

Click volumes

Click create volume

Choose standard

Click yes create

Now we have new volume ready.

We will attach to ec2 instance right click and choose attach volume select ec2 instance and yes attach.

**Create a snapshot**: Replicates data in the volume.

**S3 or Simple Storage Service:**

* Backing up of data daily requires more storage space.
* Building and maintaining our own storage space is expensive and time consuming.
* We need to stack lot of racks and to get it up and running we need lot of dedicated staff.
* AWS S3 is safe secure object storage. Store any amount of data from ec2 or from anywhere on the web. Choose a region where the data to be stored create a bucket and press a button all our data is stored.
* Automatically copies the data on multiple regions.
* We actually pay for the storage actually we use there is no minimum fee and no setup cost.
* Data that is not required frequently the data is secured by archiving them to amazon glacier.
* Also provides flexibility to control that can access the data with IAM, Access control list, Bucket policies.

**IAM:**

* Provides security to AWS resources to user of AWS account.
* Using IAM we can manage AWS users and Groups.
* And assign permission to them to allow and deny their access to AWS resources.

**How do we manage IAM resource and their access?**

* Create users and assign them individual security credentials like access keys passwords and multi factor authentication devices or temporary access credentials to access AWS services and resources.
* We can manage permissions in order to control which operation user can perform.

**How are permissions configured?**

* Permission is configured by creating IAM policies.
* Policy specifies the actions to be performed on the resource that can be affected by those actions and whether the action is being allowed or denied.
* Permissions can be managed by attaching policies to resources, users, or groups.

**IAM roles can be used to provide temporary credentials to users who do not have access to AWS resources.**

We can define which entity can use the role.

For example: we can create a read only role to allow an application to view EC2 details by attaching AWS EC2 read only access policy to a role.

IAM allows users to control access to AWS services APIs and to specific resources.

Limiting access by restricting they’re originating IP address, requiring them to use SSL or enforcing them to use of multi factor authentication device.

We can create an active directory (AD) and create authentication to AWS management console AWS APIs and resources.

**Do we have to use Microsoft active directory for federated user access?**

Third party users can be supported using AWS Security Token Service (STS) assume Role request.

We can use AWS multi factor authentication (MFA), which is free required users to prove possession of an extra factor for verification.

**IAM is free.**

**How to create IAM Policy?**

* Sign in to AWS management console.
* Click services.
* Click IAM.
* Click users.
* Find user one and click on its name.
* As we can see that user is not a member of any group’s and has no policies associated with it.
* Click groups from left navigation pan.
* Click the group name for ec2 support .
* Scroll down to the permission section of the group detail page and we can see that this group as a policy associated with it named EC2 support policy.
* We can check what permissions are given inside the policy click edit

IAM policies are JSON documents each statement covers the core information about a single permission.

If there are multiple statements AWS applies a logical OR across the statements when evaluating the policy.

Example JSON format:

{

Resource: “\*” # specifies the scope of entities covered by policy group.

Action: “elastic cloud balancing: describe” #Specifies the AWS services

Effect: “Allow” # Desired result. Defines weather the

Statement will allow or deny the user

Request

**}**

**How to add a user to a group?**

* Adding a user to S3 admin group.
* Go to IAM service dashboard.
* Click groups.
* Select the group
* User sections click add users to group
* User 3 select and click the blue button add users this new user will obtain their permissions from the policy attached to the group.

**Lets update a user password?**

Select users and select the user and select manage password from user actions. Replace existing password with new password and click apply.

Click services and click IAM users sign in link copy and click users and select user 3 and top of the screen login with the token that we copied with credential. Now we can access the buckets copy the files and perform different actions on the bucket.