IAM

1) Identify and Access Management

- We can control the entire AWS by using proper permission to the IAM uses.
- IAM is global service.
- The person who provides the card details he is the Root User.
- The Users who are using from that account they are called as IAM users.
- IAM- Access control to AWS services.

2) IAM Resources:

- User
- Group
- Polies
- Roles
- Identity Providers

3) There are two types of Access.

1) Console Access (Id, Pass)
2) Programmatic Access (Access key and Secrete key)

4) Difference between Access key and secrete key

Access Key	Secrete Key
1) AWS access key ID is a form of unique account Identification	1) AWS secrete key is like private key.

5) Difference between Account 10 and Canonical ID

Account ID	Canonical JD
1) A 12-digit numbers, that uniquely identify an Aws Account.	1) A canonical user ID is an Alpha-numeric identify an AWS Account.
2) Account ID can be known by ARN, Amazon Resources Name.	2) canonical ID can be known by Canonical User ID.

6) Task:

1) create 10 users:

a) 5 console Access
b) Programmatic Access

2) Different permission to all users
3) Create Group and Set Permission
4) Add user in group

- **Console Access**

- IAM - Add User → Username → enable Console Access → next →custom password→give password→unlock→ Users must create new password at next sign in → next → Attach policies directly → old permission → Add new tag → create user.

- Go inside Username → Security credentials → copy console sign in link → paste it if new tab → enter → account ID comes automatically → enter username → Password.

- Now try to launch EC2 instance if policy attach to user.

- **Programmatic Access**

- Go inside User → Security credential → create access key → copy access key and secrete key on notepad.
- Now launch one instance in root user → connect to that instance → Sudo -2 →
  AWS configure
  Paste access key from notepad
  Paste secrete key from notepad
  Enter
  Enter

Now you have access of user by using programmatic console.

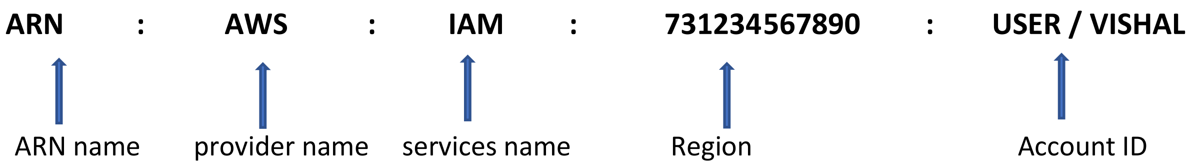
7) Now try to add User

- AWS IAM create-user→ username → kunal
- AWS IAM list uses
- User will be added to IAM – Users

8) Now Add a bucket in user using console

- Go to CLI → AWS S3 is buckets will be list.

9) ARN (Amazon Recourse Name)



ARN : AWS : IAM : 731234567890 : USER / VISHAL

ARN name provider name services name Region Account ID

1)ARN Name

2) Provider Name

3) Service Name

4) Region

5) Account ID

6) resource type and resource ID

- **MFA → Multi factor Authentication**
- Assign MPA to the uses

Steps – 1) go to I AM → users → inside user → assign MFA--> Mobile devices push notification→ Scan QR code by mobile → Download google Authentication → scan → continue → type opt in AWS → type 2nd option in AWS → ok

- 2) While IAM user logs in then account will ask for MFA code.

- **Policies**

1) Default service
2) Customer Service Policy

- **Two types of Policy editors**

1) Visual editor
2) JSON

- Create one IAM user → Give Programmatic Access → and try to take access in ubuntu machine
- Install ubuntu from Microsoft store.
- If is show error after installation like
  WS Region Distribution filled with error : 0*8007019C
  Then go to window PowerShell → then type two commands

  1) Enable Virtual Machine :- dism.exe/ online/ enable feature / feature name : Virtual Machine Platform / all / norestart

2) Enable window subsystem for Linux : dism.exe / online/ enable -feature / feature name : Microsoft – window - subsystem – Linux / all / no restart
3) Now restart system

Create a new user in ubuntu

- ➢ Sudo -1

- Now download package
- ➢ Curl → AWS CLI package name

- Now download unzip command
- ➢ apt instead unzip -y

- Now unzip AWS CLI VZ.zip package
- ➢ Unzip awsclivz.zip

- Now install AWS CLI VZ
- ➢ Sudo / aws / install

- Now check version
- ➢ Aws – version

- Now configure AWS IAM user account
- ➢ Aws configure
  Give access key
  Give private key
  Enter
  Enter

- Now check buckets created by IAM user
- ➢ Aws s3 ls

- To see content inside the buckets
- ➢ Aws ls s3 bucket name

10) AWS Users

1) Root user
2) IAM user
3) Federated user

11) Cloud Service Models Layers

- → Manage by you
- ▪ → Manage by Vender

On- Premise	IAAS	PAAS	SAAS
Application Data Runtime Middleware O/s Virtualization Servers Storage Networking	Application Data Runtime Middleware OS Virtualization Servers Storage Networking	Application Data Runtime Middleware OS Virtualization Servers Storage Networking	Application Data Runtime Middleware OS Virtualization Servers Storage Networking

12) ARN (Amazon Resource Name)

- It is a string that uniquely identifies on AWS resource, such as EC2 instance, S3 buckets, accounts, lambda functions.
- AWS requires an ARN when you want specify a resource not to open to more that one interpretation access all to AWS, such as in IAM Policies, Amazon relational database service (Amazon RDS) and API calls.

13) What is Service?

- A system or organization that provides the public with something that it needs; or the job that organizations does.

14) What is Resource?

-The resource is what you create inside the service.

15) Different Between cache and cookies.

Cache	**Cookies**
- A System uses caches for storing content from a website and application.	- A website or application uses cookies to store the user's activities and identity. Their trials of preference.
- Cache stores JavaScript, CSS, html pages, media	- Cookies stores temporary data for tracking. Such as history browsing sessions.

- The cache stores the website content only on a user browser.	- Cookies stores their content on both a server as well as browser.
- The cache needs to delete manually. It does not expire automatically.	- Cookies does not needs to delete manually. The cookies expires after a fixed amount of time.

16) Resources in AWS

- EC2 instance
- IAM user
- IAM Group
- IAM policy
- S3 Buckets
- Cloud formation Stalk

17) What is wavelength?

➔ Wavelength is a new type of AWS infrastructure design to run workloads that require ultra-low latency over mobile networks

18) What is API ?

→- Application Programming Interface

- API is mechanism of communication between two software components with each other using set of definition and protocols.

19) Difference between Authentication and Authorization

Authentication	Authorization
- In Authentication process user are verified	-In Authorization process users are validated.
- Authentication need users login details	- Authorization needs the users privilege or security level.
- Authentication determine whether the person is user or not	- Authorization determines what permission does the user have
- The user authentication is visible at user end.	- The user authorization is not visible at user end
- Popular authentication technique - → password base authentication - → multi factor authentification	- Popular authorization technique - → role base access (RBAC) - → SAMC Authorization.

- → single sign- on (SSO**)**	

20) What is Granular Permissions?

- Users with the full access that can toggle. The access level for any user data.
- Granules user can grant different permissions to different peoples for different resources.

21) What is boundary Permissions?

-→ - Boundary permissions is an advance feature for using managed policy to set the maximum permissions to on IAM user.

Policy

1) What is TAM policy in Aws ?

- IAM policies defines permissions for and action of the method that you use to perform the operation.

2) Types of Policy

1) Identify based Policy :
- you can attach managed and inline policies to IAM identities (user, group and roles)

2) Resource Based Policy :
- We can attach inline policies to resource some AWS services.

3) Organizations SCP's :
- We can use an AWS organizations service controls policies (SCP) to apply a permissions boundary to an AWS organization.
-

4) Access control List (ACLs) :
- ACLs are similar to resource based policies ACL does not use the JSON policy documents structure.

3) There are two types of policies :

1) Manage Policy
2) Custom policy or Inline Policy

1) Manage Policy : created and manage by AWS.

2) Inline or custom policy : created and manage by customer.

4) These are two types of policy generations

- Visual Editor
- JSON

Note : Also we can generate policy using AWS policy generator

- Go to AWS policy generator --. Select policy type → effect → allow / deny → AWS service → Actions → ARN → Add statement → Generate Policy.

5) Two ways to Access AWS

- Console Access
- Programmatic Access

1) Console Access:

- A) IAM username
- B) IAM user Password

2) Programmatic Access :

- A) Access Key
- B) Secrete Key

- Login with keys (Access key , secrete key)
- Key are user specific , individuals IAM user have their own keys
- Every IAM user can have max set of 2 keys
- Once key is lost. It is lost, you cannot get the some keys back but you can re-generate n number of times.
- If you re-generate, you will get new keys, you cannot get the old keys back.

IAM Group

➢ IAM groups = Collection of IAM Users

- Groups under group are not possible / nested group are not possible.
- It is possible to attach multiple policies to the IAM users and groups also, max 10
- We can attach and detach policies to the IAM user and group anytime
- If you attach any user to the group, his/ her individual policies remains some and the new permissions will be inherited to the IAM user.
- You cannot assign / create keys to the group.
- IAM groups are used to assign policies to the bunch of IAM users at the same time.
- Policies → Policy → policy document contains permissions.

IAM Roles

1) **Roles :** Temporary access without credentials / credit/ debit.

- If we user the roles, we no need to configure keys on the machine.
- Based on the permissions you have attached to the role those permissions are available from the machine.
- EC2 instance can have only 1 role attached at same time,
- 1 role can be attach to multiple EC2 instance at the same time.

Steps :- Go to IAM → Roles -→ Create Role → select trusted Entity→ select use cases (EC2 or S3) → use cases for other AWS service l→ select S3 or any → next→ select permissions policy→ S3 full access or another → next→ give role name → give tags→ create Role.

- We have created role for EC2 instance
- Now go to EC2 instance → select instance→ Action → security→ Modify IAM role→ select IAM role → update IAM role

- From now we can get access of instance without using credential (Access key / secrete key)

Identify Providers/ federation

1) **SSO – single sign on**

- Like login with Gmail login with LinkedIn

2) **What is Identity Federation?**

- Identity federation is a way to log in to one site using credential from another.
- This way you only need to remember one set of login information and don't have to worry about remembering and don't have to worry about remembering multiple username and passwords. Instead user can use a single credential to access all their online accounts.

- The most common identity provides are social media sites likes Facebook and google.

IAM Tags

- Tags are key value pair
- Tags are used for identification purpose
- Tags are used for automation purpose
- Tags are also used for cost optimization
- Tags are not IAM specific, it is throughout AWS
- Per Resource = 50 tags.
-