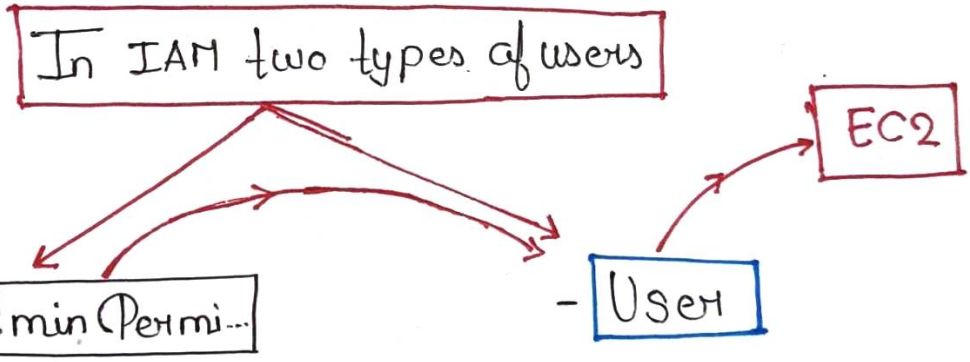
IAM - Identify Access Management

IAM is the Security Service in aws

- IAM is two type of approaches
  - 1. Console base (manual approach) (Root mail id Passwd
  - 2. Programmatical approach --- Access key and Secret keyID
    - Ex: aws manual activities create and delete manually by clicking the Service
    - Ex: Terraform, CLI, Python CDK



- By default having all Permissions
- Root can login both console and Programmatical

- IAM User Create by Root with Permissions
- Root will decide only console or both
- Create user and add ec2 Permissions and finally generate Credentials to user Either Console or Programmatical.

- In Programmatical approach

1. key (access key and Secret key)

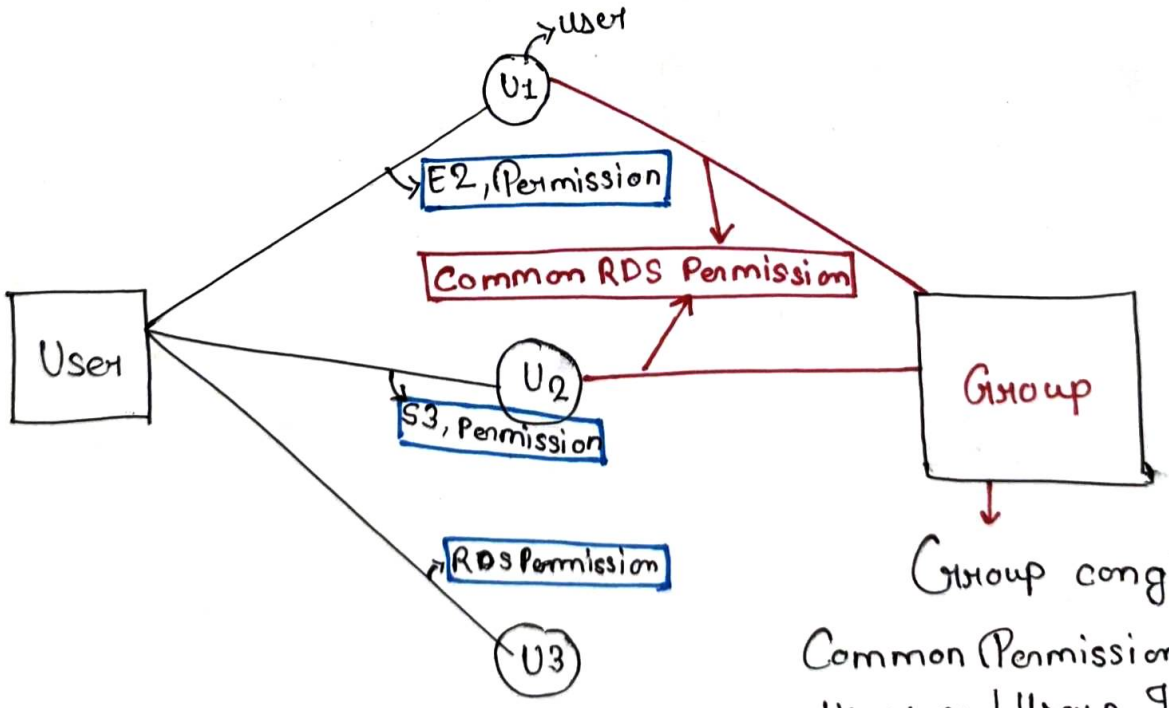
- AWS CLI
- AWS configure
- accesskey
- Secret keyid
- region : ap-south-1
- Output : json

Note: - After configure AWS .AWS folder is going to create in your local laptop user here it will store keys details and region details

- AWS CLI

IAM "Group"

Contains number of users



Group congim

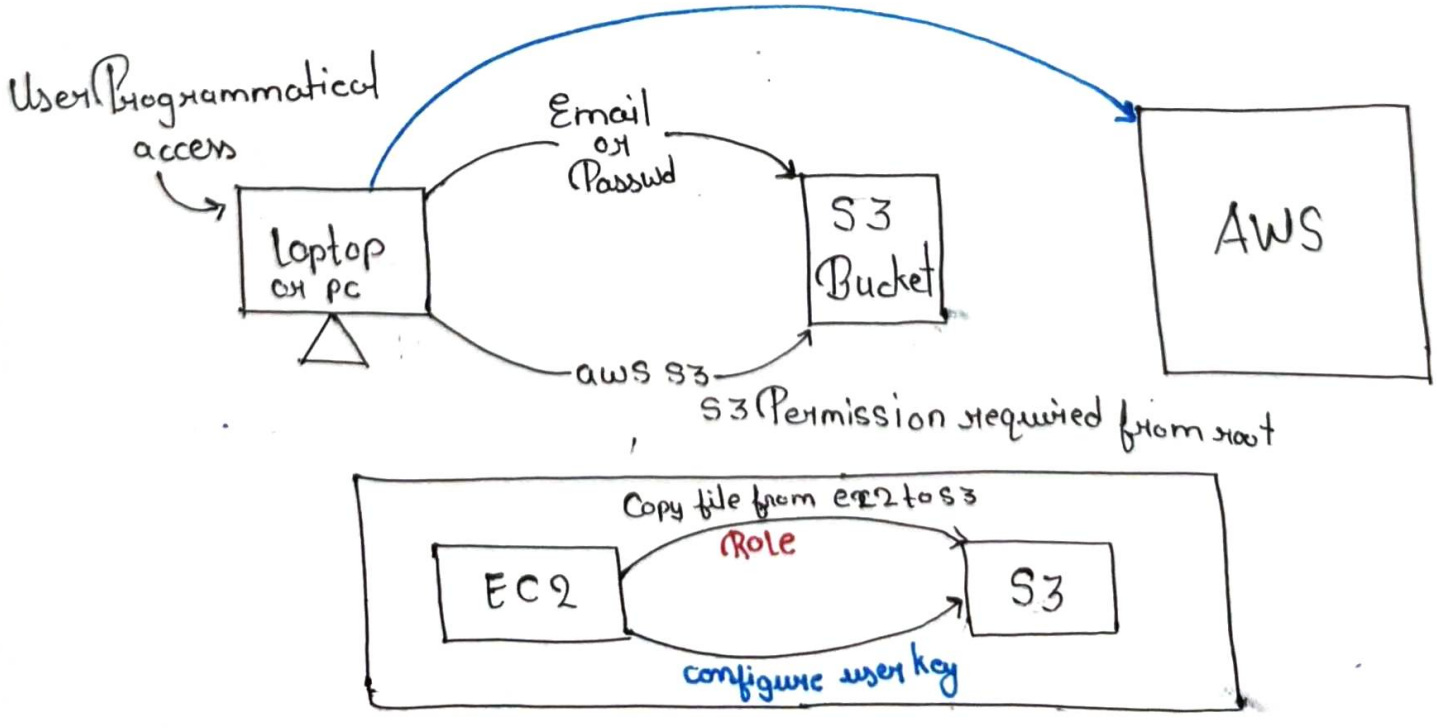
Common Permission to User 1 and User 2, Individual he cannot share own permission E2, S3, only share RDS Permission.

AWS managed Policies - Standalone policy created and **administered** by AWS

Customer managed Policies - Policy you create for **Specific use case**, and you change or update them as often as you like.

Inline Policies - Policy created for a Single IAM (Identity **user group**, or **role**) that maintains a Strict one-to-one relation Ship between a policy and an identity.

IAM Roles



- IAM Role: - An IAM Role is a **temporary permission** Card that **AWS Services or users** can wear to get Permissions.
  - A role has Permissions
  - A role does **not have a** **username** and **Password**
  - Anyone (Service or user) who "assumes" that role gets its **permissions temporarily** //