Security & IAM Management

IAM Role & Policy Management – Secure AWS resources using IAM policies.

Step 1: Create an IAM User

- Login to AWS Console → Go to IAM.
- Click Users → Add User.
- Enter a Username and enable Programmatic Access (for CLI/API).
- Click Next: Permissions and Attach existing policies directly.
- Choose AdministratorAccess (or create a custom policy).
- Click Next → Review → Create User.
- Download the Access Key ID and Secret Access Key.

Step 2: Create a Custom IAM Policy

- In the IAM Console, navigate to Policies → Create Policy. Select JSON and define a custom policy (e.g., S3 Read-Only Access): json

```
{ "Version": "2012-10-17",
        "Statement": [
        {
        "Effect": "Allow",
         "Action": "s3:ListBucket",
         "Resource": "arn:aws:s3:::example-bucket"
        },
        {
         "Effect": "Allow",
        "Action": "s3:GetObject",
         "Resource": "arn:aws:s3:::example-bucket/*"
           }
         ]
         }
```

- Click Next → Name Policy (e.g., S3ReadOnlyPolicy) → Create Policy.

Step 3: Create an IAM Role

- In the IAM Console, go to Roles → Create Role.
- Choose AWS Service and select EC2 (or another service).
- Click Next and attach the S3ReadOnlyPolicy.
- Name the role (e.g., S3ReadOnlyRole) and create it.

Step 4: Assign the Role to an EC2 Instance

- Go to EC2 Console → Select an instance.
- Click Actions → Security → Modify IAM Role.
- Select S3ReadOnlyRole and apply changes.

Step 5: Verify IAM Role & Policy SSH into the EC2 instance and run: aws s3 ls s3://example-bucket

- If successful, the IAM role is working correctly.
- Try an unauthorized action, such as deleting an object, to confirm restricted access.

This project demonstrated secure AWS resource access using IAM roles and policies. we ensured controlled access to AWS services, improving security and compliance.