EC2 Deployment with Terraform

Project Overview:

This project involved deploying an Amazon EC2 instance using Terraform. The goal was to learn infrastructure as code (IaC) practices and understand the full lifecycle of cloud resources.

Tools & Technologies Used:

- Terraform v1.x
- AWS CLI
- Amazon Web Services (EC2)
- Specific AMI (Amazon Machine Image)

Step-by-Step Execution:

1. AWS Provider Configuration:
   - Defined the AWS provider in a Terraform configuration file.
   - Set the region (e.g., us-east-1) and authenticated using AWS credentials.
2. EC2 Instance Definition:
   - Wrote a Terraform configuration to launch a 't2.micro' EC2 instance.
   - Specified an AMI ID for the desired OS (e.g., Ubuntu).
3. Terraform Commands:
   a. terraform init:
      - Initialized the Terraform working directory.
      - Downloaded the necessary provider plugins.
   b. terraform plan:
      - Reviewed the execution plan showing changes Terraform would make.
   c. terraform apply:
      - Applied the configuration and deployed the EC2 instance.
   d. terraform destroy:

- Cleaned up and deleted the deployed EC2 instance to avoid charges.

Security Best Practices:

- Avoid committing sensitive data like .tfstate files to version control.

- Use .gitignore to exclude sensitive and system-specific files.

Lessons Learned:

- Improved understanding of Terraform's declarative approach.

- Learned the full lifecycle management of AWS resources.

- Understood the importance of cleanup and cost management.