**Infrastructure as Code with Terraform**

**Overview**

This project automates the deployment of a web server using Terraform on AWS. It includes:

* A **Managed Instance Group (MIG)** for automatic scaling
* A **Layer 7 Load Balancer** for high availability
* **Route 53 DNS configuration** for domain resolution
* **Terraform** for Infrastructure as Code (IaC) management

**Architecture**

The infrastructure follows a scalable and secure architecture:

* **EC2 Instances**: Deployed within an Auto Scaling Group (MIG)
* **Application Load Balancer (ALB)**: Distributes traffic across instances
* **Route 53 Hosted Zone**: Manages DNS entries for the web application
* **Terraform**: Automates resource provisioning

**Prerequisites**

Before running this project, ensure you have:

1. **AWS Account** with required IAM permissions (EC2, Route 53, ALB, IAM, S3)
2. **Terraform installed** ([Install Guide](https://developer.hashicorp.com/terraform/tutorials/aws-get-started/install-cli))
3. **AWS CLI installed** ([Install Guide](https://docs.aws.amazon.com/cli/latest/userguide/install-cliv2.html))
4. **A registered domain** in AWS Route 53 (or another registrar)

**Setup & Deployment**

**Step 1: Clone the Repository**

git clone https://github.com/yourusername/aws-terraform-project.git

cd aws-terraform-project

**Step 2: Configure AWS Credentials**

Ensure your AWS CLI is set up:

aws configure

Enter your **AWS Access Key**, **Secret Key**, **Default Region**, and **Output Format**.

**Step 3: Initialize Terraform**

terraform init

This command initializes Terraform, downloads required providers, and sets up backend configurations.

**Step 4: Plan the Deployment**

terraform plan

This previews the changes Terraform will make in AWS.

**Step 5: Apply Changes & Deploy**

terraform apply -auto-approve

Terraform provisions the infrastructure. This may take a few minutes.

**Step 6: Retrieve Load Balancer URL**

Once deployed, get the **Load Balancer DNS name**:

echo $(terraform output load\_balancer\_dns)

Use this URL to access your application.

**Step 7: (Optional) Configure a Custom Domain**

* Go to **AWS Route 53 > Hosted Zones**
* Create an **A Record** pointing to the Load Balancer

**Destroying the Infrastructure**

To remove all resources created by Terraform, run:

terraform destroy -auto-approve

**Project Structure**

aws-terraform-project/

│── aws/console

│── main.tf

│── change-batch.json

│── terraform.tfstate

│── README.md

**Troubleshooting**

* If AWS CLI commands fail, check credentials using:
* aws sts get-caller-identity
* If Terraform fails, ensure IAM permissions allow resource creation.

**Author:** Tafadzwa Shingirai Mwerenga