Immutable Infrastructure on Ubuntu with Packer and Terraform

Step 1: Install Prerequisites

Update and Upgrade System Packages

bash

```
sudo apt-get update

sudo apt-get upgrade -y
```

Install Packer

bash

```
sudo apt-get install -y unzip

wget https://releases.hashicorp.com/packer/1.7.8/packer_1.7.8_linux_amd64.zip

unzip packer_1.7.8_linux_amd64.zip

sudo mv packer /usr/local/bin/
```

Install Terraform

bash

```
wget https://releases.hashicorp.com/terraform/1.0.11/terraform_1.0.11_linux_amd64.zip

unzip terraform_1.0.11_linux_amd64.zip
```

```
sudo mv terraform /usr/local/bin/
```

Install AWS CLI

bash

```
sudo apt-get install -y awscli
```

Step 2: Create a Packer Template

Create a New Directory for Your Packer Template

bash

```
mkdir packer-templates

cd packer-templates
```

Create a File Named `ubuntu.json`

Replace `ami-0c55b159cbfafe1f0` with the valid AMI ID `ami-0c819f65440d5f1d1`.

json

```
{

"variables": {

"aws_access_key": "{{env `AWS_ACCESS_KEY_ID`}}",

"aws_secret_key": "{{env `AWS_SECRET_ACCESS_KEY`}}"

},
"builders": [{
"type": "amazon-ebs",
"access_key": "{{user `aws_access_key`}}",
"secret_key": "{{user `aws_secret_key`}}",
"region": "us-east-1",
"source_ami": "ami-0c819f65440d5f1d1",
"instance_type": "t2.micro",
"ssh_username": "ubuntu",
"ami_name": "packer-example {{timestamp}}"
}],
"provisioners": [{
"type": "shell",
"inline": [
"sudo apt-get update",
"sudo apt-get upgrade -y",
"sudo apt-get install -y nginx"
]
}]
}
```

Step 3: Build the Image with Packer

Export Your AWS Credentials (if not already set)

bash

```
export AWS_ACCESS_KEY_ID=your_access_key_id
export AWS_SECRET_ACCESS_KEY=your_secret_access_key
```

Build the Image Using Packer

bash

```
packer build ubuntu.json
```

Step 4: Deploy the Image with Terraform

Create a New Directory for Your Terraform Configuration

bash

```
mkdir terraform-config

cd terraform-config
```

Create a File Named `main.tf`

hcl

```
provider "aws" {
region = "us-east-1"
}

variable "ami_id" {}

resource "aws_instance" "example" {
ami           = var.ami_id
instance_type = "t2.micro"
tags = {

Name = "packer-example-instance"

}

}
```

Initialize Terraform

bash

```
terraform init
```

Apply the Terraform Configuration

bash

```
terraform apply -var="ami_id=ami-0c819f65440d5f1d1"
```

Step 5: Test and Verify

Verify That the Instance Is Running and Accessible

bash

```
aws ec2 describe-instances --filters "Name=tag:Name,Values=packer-example-instance"
```

Access the Instance

bash

```
ssh ubuntu@your_instance_ip
```

You should see that Nginx is installed and running.

Step 6: Clean Up

Destroy the Terraform-Managed Infrastructure

bash

```
terraform destroy
```

Remove the AMI Created by Packer If No Longer Needed

bash

```
aws ec2 deregister-image --image-id ami-0c819f65440d5f1d1
```

This setup provides a basic workflow for creating immutable infrastructure on AWS using Ubuntu, Packer, and Terraform. Adjust the `provisioners` in your Packer template and the `resource` configurations in Terraform as needed for your specific use case.