

Write Terraform script to create a custom VPC and deploy two EC2 VMs on AWS using Terraform.

The code should be broken into three different parts:

Networking (define the VPC and all of its components)

SSH-Key (dynamically create an SSH-key pair for connecting to VMs)

EC2 (deploy a VM in the public subnet, and deploy another VM in a private subnet)

NGINX should be accessed for all the internet

Automate Terraform Deployment with Jenkins Pipelines.

```
terraform-aws-project/
├── Jenkinsfile
├── ec2
│   ├── main.tf
│   ├── outputs.tf
│   └── variables.tf
├── main.tf
├── networking
│   ├── main.tf
│   ├── outputs.tf
│   └── variables.tf
├── outputs.tf
├── ssh-key
│   ├── main.tf
│   └── outputs.tf
├── terraform.tfstate
├── terraform.tfstate.backup
└── variables.tf
```

```
sigmoid@sigmoid-ThinkPad-T450:~/Desktop/terraform-aws-project$ terraform apply
module.ssh-key.tls_private_key.ssh_key: Refreshing state... [id=cfc0b3cc9b7f34cc3781e75847d0077139a3069c]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

# module.ec2.aws_instance.private_vm will be created
+ resource "aws_instance" "private_vm" {
  + ami                  = "ami-01184db239e4c756c"
  + arn                  = (known after apply)
  + associate_public_ip_address = (known after apply)
  + availability_zone     = (known after apply)
  + cpu_core_count       = (known after apply)
  + cpu_threads_per_core = (known after apply)
  + disable_api_stop      = (known after apply)
  + disable_api_termination = (known after apply)
  + ebs_optimized         = (known after apply)
  + get_password_data     = false
  + host_id               = (known after apply)
  + host_resource_group_arn = (known after apply)
  + iam_instance_profile  = (known after apply)
  + id                    = (known after apply)
  + instance_initiated_shutdown_behavior = (known after apply)
  + instance_lifecycle    = (known after apply)
  + instance_state        = (known after apply)
  + instance_type         = "t2.micro"
  + ipv6_address_count    = (known after apply)
  + ipv6_addresses        = (known after apply)
  + key_name              = "generated-key"
  + monitoring            = (known after apply)
  + outpost_arn           = (known after apply)
  + password_data         = (known after apply)
```

```

module.ec2.aws_instance.private_vm: Creating...
module.networking.aws_vpc.custom_vpc: Still creating... [10s elapsed]
module.networking.aws_subnet.public: Still creating... [10s elapsed]
module.networking.aws_vpc.custom_vpc: Creation complete after 15s [id=vpc-0a9886a5785ad0ddb]
module.networking.aws_internet_gateway.igw: Creating...
module.networking.aws_subnet.private_subnet: Creating...
module.networking.aws_route_table.public_rt: Creating...
module.networking.aws_subnet.public_subnet: Creating...
module.ec2.aws_instance.private_vm: Still creating... [10s elapsed]
module.networking.aws_subnet.private_subnet: Creation complete after 1s [id=subnet-074addfd6e8683cf5]
module.networking.aws_internet_gateway.igw: Creation complete after 1s [id=igw-09943db4920a4fc2f]
module.networking.aws_subnet.public: Creation complete after 12s [id=subnet-0342e882ffc11fc4a]
module.ec2.aws_instance.public_vm: Creating...
module.networking.aws_route_table.public_rt: Creation complete after 1s [id=rtb-0eb8a4cdfbe80948d]
module.networking.aws_route.public_internet_route: Creating...
module.networking.aws_route.public_internet_route: Creation complete after 2s [id=r-rtb-0eb8a4cdfbe80948d1080289494]
module.ec2.aws_instance.private_vm: Creation complete after 15s [id=i-0bcae09e8da6c0f00]
module.networking.aws_subnet.public_subnet: Still creating... [10s elapsed]
module.ec2.aws_instance.public_vm: Still creating... [10s elapsed]
module.networking.aws_subnet.public_subnet: Creation complete after 12s [id=subnet-0ecc8d7d02f7dc88b]
module.networking.aws_route_table_association.public_subnet_association: Creating...
module.networking.aws_route_table_association.public_subnet_association: Creation complete after 1s [id=rtbassoc-079366ea43260fc65]
module.ec2.aws_instance.public_vm: Still creating... [20s elapsed]
module.ec2.aws_instance.private_vm: Still creating... [30s elapsed]
module.ec2.aws_instance.public_vm: Still creating... [40s elapsed]
module.ec2.aws_instance.public_vm: Still creating... [50s elapsed]
module.ec2.aws_instance.public_vm: Creation complete after 56s [id=i-0f5b3bce6e7119c75]

Apply complete! Resources: 13 added, 0 changed, 0 destroyed.

Outputs:

private_vm_private_ip = "10.0.2.162"
public_vm_public_ip = "52.90.5.171"
signold@signold-ThinkPad-T450:~/Desktop/terraform-aws-project$

```

Search (CTRL+K)

admin

log out

Dashboard > terraform Deployment > #12

Status

Changes

Console Output

Edit Build Information

Timings

Git Build Data

Pipeline Overview

Pipeline Console

Thread Dump

Pause/resume

Replay

Pipeline Steps

Workspaces

Previous Build

Console Output

Download

Copy

View as plain text

Started by user admin

[Pipeline] Start of Pipeline

[Pipeline] node

Running on Jenkins in /var/lib/jenkins/workspace/terraform Deployment

[Pipeline] {

[Pipeline] withCredentials

WARNING: Unknown parameter(s) found for class type 'com.cloudbees.jenkins.plugins.awscredentials.AmazonWebServicesCredentialsBinding': keyIdVariable,secretVariable

WARNING: Unknown parameter(s) found for class type 'com.cloudbees.jenkins.plugins.awscredentials.AmazonWebServicesCredentialsBinding': keyIdVariable,secretVariable

Masking supported pattern matches of \$AWS_ACCESS_KEY_ID or \$AWS_SECRET_ACCESS_KEY

[Pipeline] {

[Pipeline] withEnv

[Pipeline] {

[Pipeline] stage

[Pipeline] { (Clone Repository)

[Pipeline] git

The recommended git tool is: NONE

No credentials specified

Cloning the remote Git repository

← → ↻

http://localhost:8080/job/terraform%20Deployment/12/console

☆

⌵

Dashboard > terraform Deployment > #12

```

[[@private_vm_private_ip = "10.0.2.218"
public_vm_public_ip = "54.175.10.36"
(Pipeline) }
(Pipeline) // script
(Pipeline) }
(Pipeline) // stage
(Pipeline) stage
(Pipeline) { (Declarative: Post Actions)
(Pipeline) cleanWs
[WS-CLEANUP] Deleting project workspace...
[WS-CLEANUP] Deferred wipeout is used...
[WS-CLEANUP] done
(Pipeline) }
(Pipeline) // stage
(Pipeline) }
(Pipeline) // withEnv
(Pipeline) }
(Pipeline) // withCredentials
(Pipeline) }
(Pipeline) // node
(Pipeline) End of Pipeline
Finished: SUCCESS

```

