**TERRAFORM ASSIGNMENT 3 (BUILD INFRASTRUCTURE)**

Microsoft Windows [Version 10.0.22000.675]

(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Downloads\terraform\_1.1.9\_windows\_amd64>aws configure

AWS Access Key ID [\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*AUIW]: AKIAYSTMH7S64MKBAUIW

AWS Secret Access Key [\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*9yne]: 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

Default region name [None]: ap-south-1

Default output format [None]: JSON

C:\Users\Administrator\Downloads\terraform\_1.1.9\_windows\_amd64>terraform init

Initializing the backend...

Initializing provider plugins...

- Finding hashicorp/aws versions matching "~> 4.14.0"...

- Installing hashicorp/aws v4.14.0...

- Installed hashicorp/aws v4.14.0 (signed by HashiCorp)

Terraform has created a lock file .terraform.lock.hcl to record the provider

selections it made above. Include this file in your version control repository

so that Terraform can guarantee to make the same selections by default when

you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see

any changes that are required for your infrastructure. All Terraform commands

should now work.

If you ever set or change modules or backend configuration for Terraform,

rerun this command to reinitialize your working directory. If you forget, other

commands will detect it and remind you to do so if necessary.

C:\Users\Administrator\Downloads\terraform\_1.1.9\_windows\_amd64>terraform apply

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:

# aws\_instance.app\_server will be created

+ resource "aws\_instance" "app\_server" {

+ ami = "ami-079b5e5b3971bd10d"

+ 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\_termination = (known after apply)

+ ebs\_optimized = (known after apply)

+ get\_password\_data = false

+ host\_id = (known after apply)

+ id = (known after apply)

+ instance\_initiated\_shutdown\_behavior = (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 = (known after apply)

+ monitoring = (known after apply)

+ outpost\_arn = (known after apply)

+ password\_data = (known after apply)

+ placement\_group = (known after apply)

+ placement\_partition\_number = (known after apply)

+ primary\_network\_interface\_id = (known after apply)

+ private\_dns = (known after apply)

+ private\_ip = (known after apply)

+ public\_dns = (known after apply)

+ public\_ip = (known after apply)

+ secondary\_private\_ips = (known after apply)

+ security\_groups = (known after apply)

+ source\_dest\_check = true

+ subnet\_id = (known after apply)

+ tags = {

+ "Name" = "aws\_instance"

}

+ tags\_all = {

+ "Name" = "aws\_instance"

}

+ tenancy = (known after apply)

+ user\_data = (known after apply)

+ user\_data\_base64 = (known after apply)

+ user\_data\_replace\_on\_change = false

+ vpc\_security\_group\_ids = (known after apply)

+ capacity\_reservation\_specification {

+ capacity\_reservation\_preference = (known after apply)

+ capacity\_reservation\_target {

+ capacity\_reservation\_id = (known after apply)

+ capacity\_reservation\_resource\_group\_arn = (known after apply)

}

}

+ ebs\_block\_device {

+ delete\_on\_termination = (known after apply)

+ device\_name = (known after apply)

+ encrypted = (known after apply)

+ iops = (known after apply)

+ kms\_key\_id = (known after apply)

+ snapshot\_id = (known after apply)

+ tags = (known after apply)

+ throughput = (known after apply)

+ volume\_id = (known after apply)

+ volume\_size = (known after apply)

+ volume\_type = (known after apply)

}

+ enclave\_options {

+ enabled = (known after apply)

}

+ ephemeral\_block\_device {

+ device\_name = (known after apply)

+ no\_device = (known after apply)

+ virtual\_name = (known after apply)

}

+ maintenance\_options {

+ auto\_recovery = (known after apply)

}

+ metadata\_options {

+ http\_endpoint = (known after apply)

+ http\_put\_response\_hop\_limit = (known after apply)

+ http\_tokens = (known after apply)

+ instance\_metadata\_tags = (known after apply)

}

+ network\_interface {

+ delete\_on\_termination = (known after apply)

+ device\_index = (known after apply)

+ network\_card\_index = (known after apply)

+ network\_interface\_id = (known after apply)

}

+ root\_block\_device {

+ delete\_on\_termination = (known after apply)

+ device\_name = (known after apply)

+ encrypted = (known after apply)

+ iops = (known after apply)

+ kms\_key\_id = (known after apply)

+ tags = (known after apply)

+ throughput = (known after apply)

+ volume\_id = (known after apply)

+ volume\_size = (known after apply)

+ volume\_type = (known after apply)

}

}

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.

Only 'yes' will be accepted to approve.

Enter a value: yes

aws\_instance.app\_server: Creating...

aws\_instance.app\_server: Still creating... [10s elapsed]

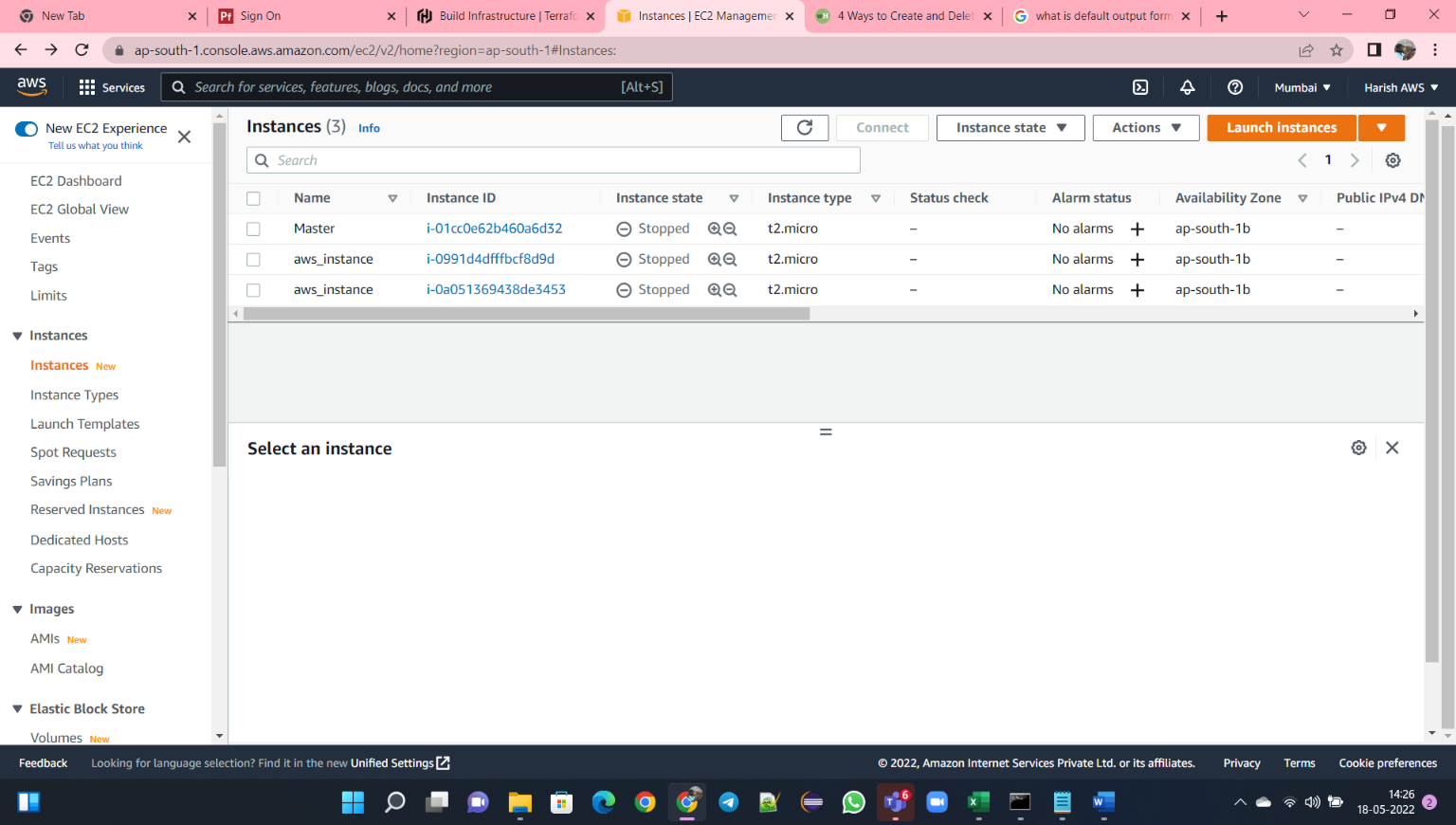
aws\_instance.app\_server: Still creating... [20s elapsed]

aws\_instance.app\_server: Still creating... [30s elapsed]

aws\_instance.app\_server: Creation complete after 32s [id=i-0a051369438de3453]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

C:\Users\Administrator\Downloads\terraform\_1.1.9\_windows\_amd64>



C:\Users\Administrator\Downloads\terraform\_1.1.9\_windows\_amd64>terraform show

# aws\_instance.app\_server:

resource "aws\_instance" "app\_server" {

ami = "ami-079b5e5b3971bd10d"

arn = "arn:aws:ec2:ap-south-1:589711277245:instance/i-0a051369438de3453"

associate\_public\_ip\_address = true

availability\_zone = "ap-south-1b"

cpu\_core\_count = 1

cpu\_threads\_per\_core = 1

disable\_api\_termination = false

ebs\_optimized = false

get\_password\_data = false

hibernation = false

id = "i-0a051369438de3453"

instance\_initiated\_shutdown\_behavior = "stop"

instance\_state = "running"

instance\_type = "t2.micro"

ipv6\_address\_count = 0

ipv6\_addresses = []

monitoring = false

primary\_network\_interface\_id = "eni-00cd577eab7474805"

private\_dns = "ip-172-31-13-40.ap-south-1.compute.internal"

private\_ip = "172.31.13.40"

public\_dns = "ec2-13-233-27-115.ap-south-1.compute.amazonaws.com"

public\_ip = "13.233.27.115"

secondary\_private\_ips = []

security\_groups = [

"default",

]

source\_dest\_check = true

subnet\_id = "subnet-08d87886393b54ee4"

tags = {

"Name" = "aws\_instance"

}

tags\_all = {

"Name" = "aws\_instance"

}

tenancy = "default"

user\_data\_replace\_on\_change = false

vpc\_security\_group\_ids = [

"sg-0e768ef13a6895c54",

]

capacity\_reservation\_specification {

capacity\_reservation\_preference = "open"

}

credit\_specification {

cpu\_credits = "standard"

}

enclave\_options {

enabled = false

}

maintenance\_options {

auto\_recovery = "default"

}

metadata\_options {

http\_endpoint = "enabled"

http\_put\_response\_hop\_limit = 1

http\_tokens = "optional"

instance\_metadata\_tags = "disabled"

}

root\_block\_device {

delete\_on\_termination = true

device\_name = "/dev/xvda"

encrypted = false

iops = 100

tags = {}

throughput = 0

volume\_id = "vol-03ea437b7c06c6a1f"

volume\_size = 8

volume\_type = "gp2"

}

}

C:\Users\Administrator\Downloads\terraform\_1.1.9\_windows\_amd64>