

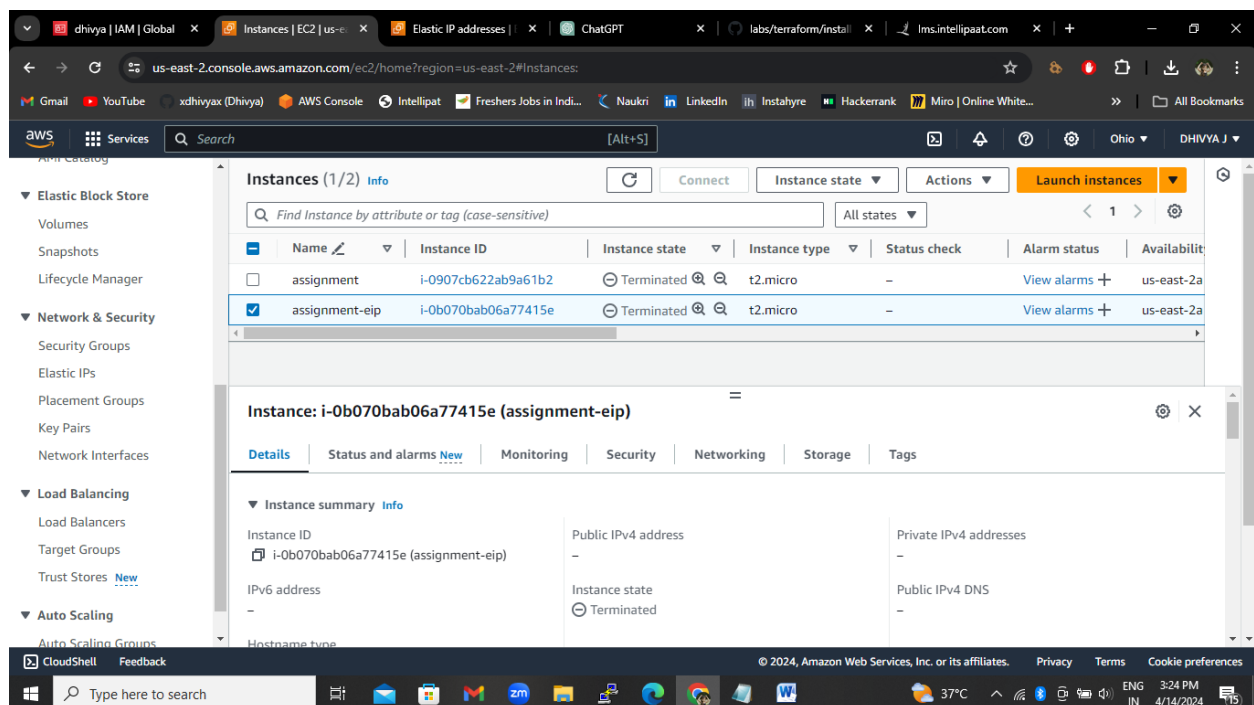
## Terraform Assignment - 3

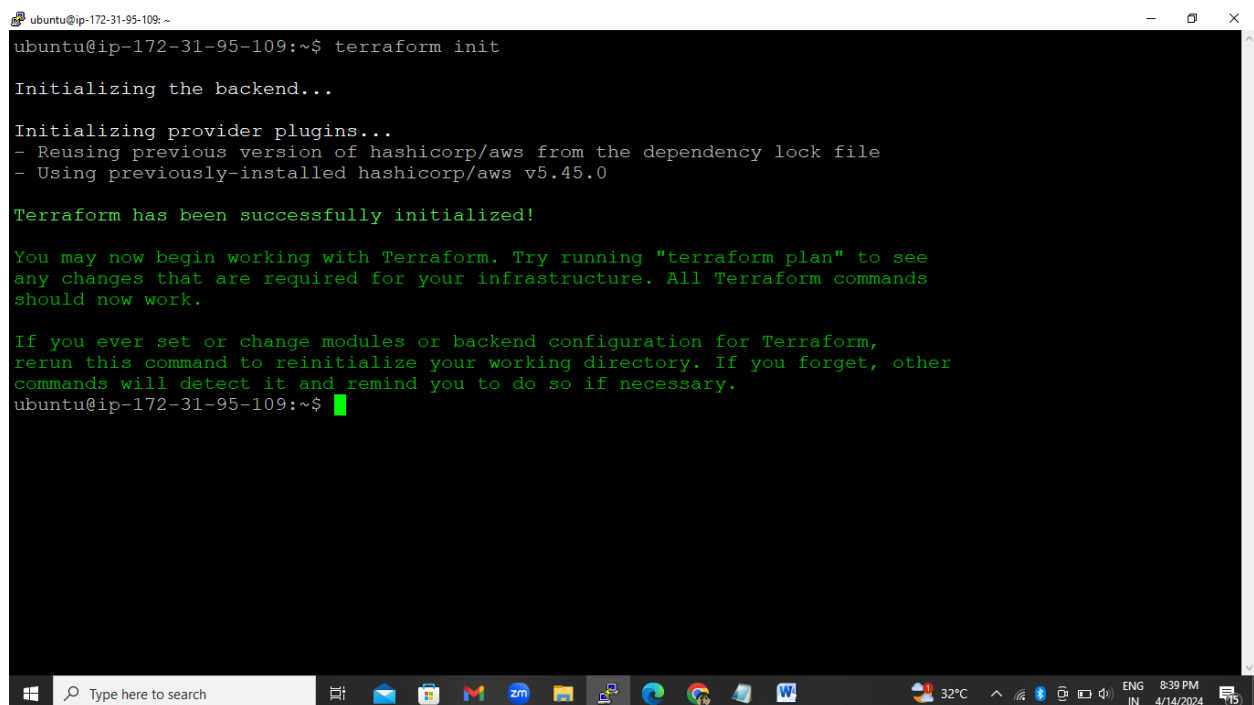
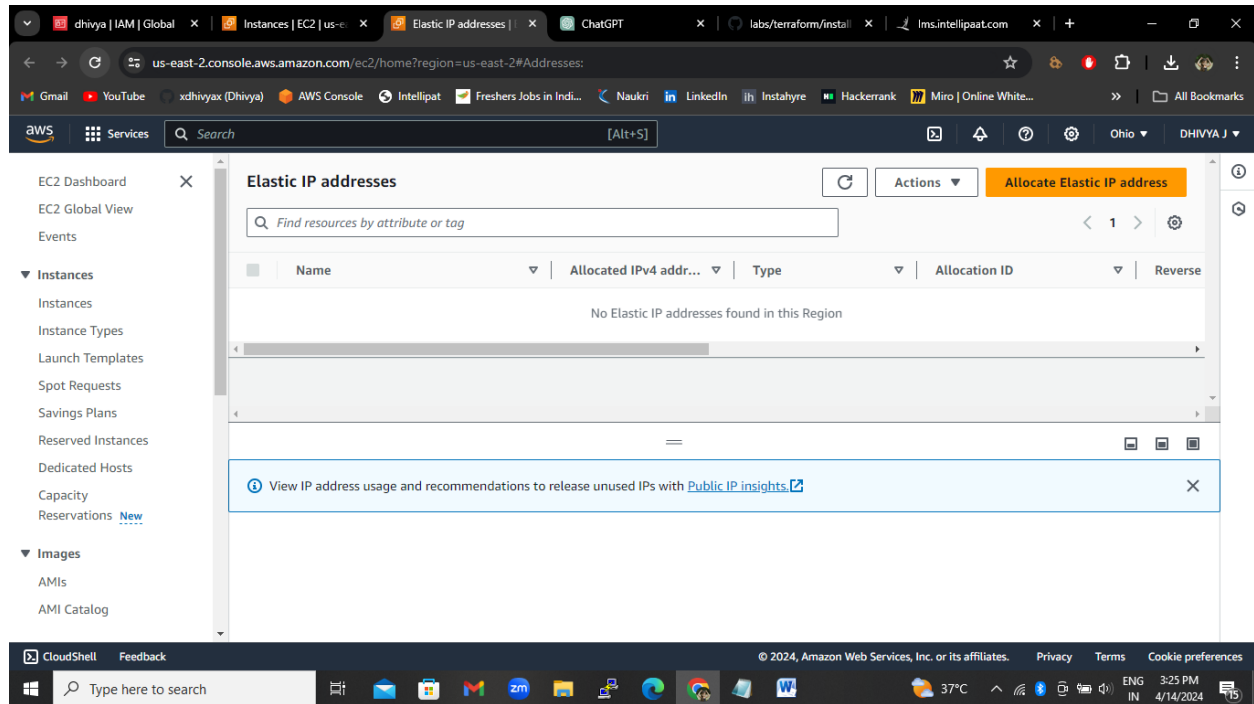
You have been asked to:

- Destroy the previous deployment
- Create 2 EC2 instances in Ohio and N.Virginia respectively
- Rename Ohio's instance to 'hello-ohio' and Virginia's instance to 'hello-virginia'

```
ubuntu@ip-172-31-95-109: ~  
ubuntu@ip-172-31-95-109:~$ terraform destroy  
aws_instance.assignment: Refreshing state... [id=i-0b070bab06a77415e]  
aws_eip.example: Refreshing state... [id=eipalloc-0c83e9d96875cbbb5]  
  
Terraform used the selected providers to generate the following execution plan. Resource actions are  
indicated with the following symbols:  
- destroy  
  
Terraform will perform the following actions:  
  
# aws_eip.example will be destroyed  
- resource "aws_eip" "example" {  
  - allocation_id      = "eipalloc-0c83e9d96875cbbb5" -> null  
  - association_id     = "eipassoc-032c09123f8e5e54b" -> null  
  - domain            = "vpc" -> null  
  - id                = "eipalloc-0c83e9d96875cbbb5" -> null  
  - instance          = "i-0b070bab06a77415e" -> null  
  - network_border_group = "us-east-2" -> null  
  - network_interface  = "eni-03004515359a77e13" -> null  
  - private_dns        = "ip-172-31-6-114.us-east-2.compute.internal" -> null  
  - private_ip         = "172.31.6.114" -> null  
  - public_dns         = "ec2-3-128-68-76.us-east-2.compute.amazonaws.com" -> null  
  - public_ip          = "3.128.68.76" -> null  
  - public_ipv4_pool    = "amazon" -> null  
  - tags               = {  
    - "Name" = "example-eip"  
  } -> null  
  - tags_all           = {  
    - "Name" = "example-eip"  
  } -> null
```

```
ubuntu@ip-172-31-95-109: ~  
- iops = 100 -> null  
- tags = {} -> null  
- tags_all = {} -> null  
- throughput = 0 -> null  
- volume_id = "vol-09501db867083b228" -> null  
- volume_size = 8 -> null  
- volume_type = "gp2" -> null  
# (1 unchanged attribute hidden)  
}  
}  
  
Plan: 0 to add, 0 to change, 2 to destroy.  
  
Do you really want to destroy all resources?  
Terraform will destroy all your managed infrastructure, as shown above.  
There is no undo. Only 'yes' will be accepted to confirm.  
  
Enter a value: yes  
  
aws_eip.example: Destroying... [id=eipalloc-0c83e9d96875cbbb5]  
aws_eip.example: Destruction complete after 2s  
aws_instance.assignment: Destroying... [id=i-0b070bab06a77415e]  
aws_instance.assignment: Still destroying... [id=i-0b070bab06a77415e, 10s elapsed]  
aws_instance.assignment: Still destroying... [id=i-0b070bab06a77415e, 20s elapsed]  
aws_instance.assignment: Still destroying... [id=i-0b070bab06a77415e, 30s elapsed]  
aws_instance.assignment: Still destroying... [id=i-0b070bab06a77415e, 40s elapsed]  
aws_instance.assignment: Destruction complete after 40s  
  
Destroy complete! Resources: 2 destroyed.  
ubuntu@ip-172-31-95-109:~$
```





```
ubuntu@ip-172-31-95-109: ~  
ubuntu@ip-172-31-95-109:~$ 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.ohio-instance will be created  
+ resource "aws_instance" "ohio-instance" {  
+ ami                        = "ami-0b8b44ec9a8f90422"  
+ 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)  
+ 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)  
}
```

```
}  
+ 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)  
}  
  
Plan: 2 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.virginia-instance: Creating...  
aws_instance.ohio-instance: Creating...  
aws_instance.virginia-instance: Still creating... [10s elapsed]  
aws_instance.ohio-instance: Still creating... [10s elapsed]  
aws_instance.virginia-instance: Still creating... [20s elapsed]  
aws_instance.ohio-instance: Still creating... [20s elapsed]  
aws_instance.virginia-instance: Still creating... [30s elapsed]  
aws_instance.ohio-instance: Still creating... [30s elapsed]  
aws_instance.virginia-instance: Creation complete after 32s [id=i-05efdd3977fclc445]  
aws_instance.ohio-instance: Still creating... [40s elapsed]  
aws_instance.ohio-instance: Creation complete after 42s [id=i-0df23ace6da5be243]  
  
Apply complete! Resources: 2 added, 0 changed, 0 destroyed.  
ubuntu@ip-172-31-95-109:~$
```

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#instances:instanceId=i-05efdd3977fc1c445v=3:\$case=tag:true%5Cclient:false:\$r...

Instances (1) Info

Find Instance by attribute or tag (case-sensitive)

Instance ID = i-05efdd3977fc1c445

Clear filters

All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
<input type="checkbox"/>	hello-virginia	i-05efdd3977fc1c445	Running	t2.micro	Initializing	View alarms	us-east-1a

Select an instance

CloudShell Feedback

Type here to search

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

32°C 8:41 PM 4/14/2024

us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#instances:

Instances (1) Info

Find Instance by attribute or tag (case-sensitive)

All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
<input type="checkbox"/>	hello-ohio	i-0df23ace6da5be243	Running	t2.micro	Initializing	View alarms	us-east-2a

Instance: i-0b070bab06a77415e (assignment-eip)

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary Info

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0b070bab06a77415e (assignment-eip)	-	-
IPv6 address	Instance state	Public IPv4 DNS
-	Terminated	-
Hostname type		

CloudShell Feedback

Type here to search

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

32°C 8:42 PM 4/14/2024

```
ubuntu@ip-172-31-95-109: ~  
ubuntu@ip-172-31-95-109:~$ cat ec2-diff-region.tf  
provider "aws"{  
  access_key = "AKIA5LMDQBKCCJJE4YA"  
  secret_key = "V08DsUKssqdm9QpVF9tZtUmvOawJZGEGZgLBVs2/"  
  region = "us-east-2"  
}  
resource "aws_instance" "ohio-instance" {  
  ami           = "ami-0b8b44ec9a8f90422"  
  instance_type = "t2.micro"  
  security_groups = ["default"]  
  
  tags = {  
    Name = "hello-ohio"  
  }  
}  
provider "aws"{  
  alias = "us-east-1"  
  access_key = "AKIA5LMDQBKCCJJE4YA"  
  secret_key = "V08DsUKssqdm9QpVF9tZtUmvOawJZGEGZgLBVs2/"  
  region = "us-east-1"  
}  
  
resource "aws_instance" "virginia-instance" {  
  provider      = aws.us-east-1  
  ami           = "ami-080elf13689e07408"  
  instance_type = "t2.micro"  
  security_groups = ["default"]  
  
  tags = {  
    Name = "hello-virginia"  
  }  
}
```

```
ubuntu@ip-172-31-95-109: ~  
    region = "us-east-2"  
}  
resource "aws_instance" "ohio-instance" {  
  ami           = "ami-0b8b44ec9a8f90422"  
  instance_type = "t2.micro"  
  security_groups = ["default"]  
  
  tags = {  
    Name = "hello-ohio"  
  }  
}  
provider "aws"{  
  alias = "us-east-1"  
  access_key = "AKIA5LMDQBKCCJJE4YA"  
  secret_key = "V08DsUKssqdm9QpVF9tZtUmvOawJZGEGZgLBVs2/"  
  region = "us-east-1"  
}  
  
resource "aws_instance" "virginia-instance" {  
  provider      = aws.us-east-1  
  ami           = "ami-080elf13689e07408"  
  instance_type = "t2.micro"  
  security_groups = ["default"]  
  
  tags = {  
    Name = "hello-virginia"  
  }  
}  
}  
ubuntu@ip-172-31-95-109:~$
```