

# Terraform

## Assignment – 1

### Task to be performed

1. Create EC2 service in the default subnet in the Ohio region

### Steps and Commands

- Update the machine by running the command **sudo apt update**
- Next create a file to install terraform by running the command **sudo nano (provide the name for creating file.sh)**
- Provide the necessary command to install terraform
  - a) wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
  - b) echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com \$(lsb\_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
  - c) sudo apt update && sudo apt install terraform -y
- save and exit from the created file
- To run the commands which was provided in file by running the command **sudo bash (provide the name for created file.sh)**
- To see the terraform installed in the machine by running the command **terraform –version**
- Next create a file for providing script for creating resource by running the command **sudo nano (provide the name for creating file.tf)**
- Provide the necessary script in created file to create EC2 instance in Ohio region
- save and exit from the created file
- To see the created file by running the command **ls**
- Next to initialize terraform by running the command **terraform init**
- And next run the command **terraform plan**
- To create the necessary resource by running the command **terraform apply**
- And also provide **yes** for confirmation.
- Now the new EC2 instance will be created in Ohio region by the terraform.

```
Swap usage:  0%
Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-15-128:~$ sudo apt update
```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-15-128:~$ sudo nano install.sh
```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
GNU nano 4.8                               install.sh                         Modified
wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
sudo apt update && sudo apt install terraform
```

[Get Help](#) [Write Out](#) [Where Is](#) [Cut Text](#) [Justify](#) [Cur Pos](#) [Undo](#) [Mark Text](#) [To Bracket](#)  
[Exit](#) [Read File](#) [Replace](#) [Paste Text](#) [To Spell](#) [Go To Line](#) [Redo](#) [Copy Text](#) [Where Was](#)

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-15-128:~$ sudo nano install.sh  
ubuntu@ip-172-31-15-128:~$ sudo bash install.sh
```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
Get:6 https://apt.releases.hashicorp.com focal/main amd64 Packages [128 kB]  
Fetched 258 kB in 1s (272 kB/s)  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
41 packages can be upgraded. Run 'apt list --upgradable' to see them.  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following NEW packages will be installed:  
  terraform  
0 upgraded, 1 newly installed, 0 to remove and 41 not upgraded.  
Need to get 27.7 MB of archives.  
After this operation, 88.2 MB of additional disk space will be used.  
Get:1 https://apt.releases.hashicorp.com focal/main amd64 terraform amd64 1.8.2-1 [27.7 MB]  
Fetched 27.7 MB in 1s (52.6 MB/s)  
Selecting previously unselected package terraform.  
(Reading database ... 62027 files and directories currently installed.)  
Preparing to unpack .../terraform_1.8.2-1_amd64.deb ...  
Unpacking terraform (1.8.2-1) ...  
Setting up terraform (1.8.2-1) ...  
ubuntu@ip-172-31-15-128:~$ terraform --version  
Terraform v1.8.2  
on linux_amd64  
ubuntu@ip-172-31-15-128:~$
```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
ubuntu@ip-172-31-15-128:~$ sudo nano assign1.tf
```

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
GNU nano 4.8                                         assign1.tf                                         Modified
provider "aws"{
  region = "us-east-2"
  access_key = "AKIA47CRV22SMCWBLXVL"
  secret_key = "1SQhC2+/3yj5KrrdUEE+NLjiuIjKi+jv/tOFmT2"
}

resource "aws_instance" "assignment-1" {
  ami = "ami-0b4750268a88e78e0"
  instance_type = "t2.micro"
  key_name = "keypair-ohio-terraform"
  tags = {
    Name = "assignment-1"
  }
}

^G Get Help   ^O Write Out   ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo   M-R Mark Text   M-] To Bracket
^X Exit      ^R Read File   ^V Replace   ^P Paste Text   ^I To Spell   ^L Go To Line   M-E Redo   M-G Copy Text   ^C Where Was

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

```
ubuntu@ip-172-31-15-128:~$ sudo nano assign1.tf
ubuntu@ip-172-31-15-128:~$ terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.46.0...
- Installed hashicorp/aws v5.46.0 (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.
ubuntu@ip-172-31-15-128:~$
```

```
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.
ubuntu@ip-172-31-15-128:~$ terraform plan

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.assignment-1 will be created
+ resource "aws_instance" "assignment-1" {
  + ami                               = "ami-0b4750268a88e78e0"
  + 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)
```

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

```

+ secondary_private_ips          = (known after apply)
+ security_groups                = (known after apply)
+ source_dest_check              = true
+ spot_instance_request_id       = (known after apply)
+ subnet_id                      = (known after apply)
+ tags                           = {
    + "Name" = "assignment-1"
}
+ tags_all                       = {
    + "Name" = "assignment-1"
}
+ 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: 1 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
ubuntu@ip-172-31-15-128:~$ terraform apply

```

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

```

+ "Name" = "assignment-1"
}
+ 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: 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.assignment-1: Creating...
aws_instance.assignment-1: Still creating... [10s elapsed]
aws_instance.assignment-1: Still creating... [20s elapsed]
aws_instance.assignment-1: Still creating... [30s elapsed]
aws_instance.assignment-1: Creation complete after 31s [id=i-0f122a20bf26f30de]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-15-128:~$ 

```

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

AWS Services Search [Alt+S] Ohio ▾

**Instances (1) Info**

Find Instance by attribute or tag (case-sensitive)

Instance ID = i-0f122a20bf26f30de

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
assignment-1	i-0f122a20bf26f30de	Running	t2.micro	Initializing	View alarms

Select an instance

EC2 Dashboard EC2 Global View Events Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Capacity Reservations New Images AMIs

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookies

## Assignment – 2

### Task to be performed

1. Destroy the previous deployment
2. Create a new EC2 instance with an Elastic IP

### Steps and Commands

- To destroy the previous deployment resource which was created in assignment 1 by running the command **terraform destroy**
- And also provide **yes** for confirmation.
- Next open the file which was created in the assignment 1 for providing script for creating resource by running the command **sudo nano (provide the name of created file.tf)**
- Edit the necessary script in file to create EC2 instance with Elastic IP
- save and exit from the file
- To see the created file by running the command **ls**
- Next to initialize terraform by running the command **terraform init**
- And next run the command **terraform plan**
- To create the necessary resource by running the command **terraform apply**
- And also provide **yes** for confirmation.
- Now the new EC2 instance will be created with Elastic IP by the terraform.

```
+ "Name" = "assignment-1"
}
+ tenancy
+ user_data
+ user_data_base64
+ user_data_replace_on_change
+ vpc_security_group_ids
}

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.assignment-1: Creating...
aws_instance.assignment-1: Still creating... [10s elapsed]
aws_instance.assignment-1: Still creating... [20s elapsed]
aws_instance.assignment-1: Still creating... [30s elapsed]
aws_instance.assignment-1: Creation complete after 31s [id=i-0f122a20bf26f30de]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-15-128:~$ terraform destroy
```

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

```

    - volume_id          = "vol-035df56d023db98f2" -> null
    - volume_size        = 8 -> null
    - volume_type        = "gp2" -> null
    # (1 unchanged attribute hidden)
}

Plan: 0 to add, 0 to change, 1 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_instance.assignment-1: Destroying... [id=i-0f122a20bf26f30de]
aws_instance.assignment-1: Still destroying... [id=i-0f122a20bf26f30de, 10s elapsed]
aws_instance.assignment-1: Still destroying... [id=i-0f122a20bf26f30de, 20s elapsed]
aws_instance.assignment-1: Still destroying... [id=i-0f122a20bf26f30de, 30s elapsed]
aws_instance.assignment-1: Still destroying... [id=i-0f122a20bf26f30de, 40s elapsed]
aws_instance.assignment-1: Still destroying... [id=i-0f122a20bf26f30de, 50s elapsed]
aws_instance.assignment-1: Destruction complete after 50s

Destroy complete! Resources: 1 destroyed.
ubuntu@ip-172-31-15-128:~$ 

```

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

The screenshot shows the AWS CloudShell interface. At the top, there's a navigation bar with the AWS logo, a search bar, and various icons. Below it is a sidebar with links like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations. The main content area displays the Terraform output from the previous step, showing the destruction of an instance named 'assignment-1'. It includes log entries for the instance being destroyed, followed by a confirmation message: 'Destroy complete! Resources: 1 destroyed.' At the bottom of the main area, there's a prompt: 'ubuntu@ip-172-31-15-128:~\$ sudo nano assign1.tf'.

```

ubuntu@ip-172-31-15-128:~$ sudo nano assign1.tf

```

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
GNU nano 4.8                                     assign1.tf                                         Modified
}
secret_key = "1SQhC2+/3yj5KtrrdUEE+NLjiuIjKi+Jv/tOFmT2"
}

resource "aws_instance" "assignment-2" {
  ami = "ami-0b4750268a88e78e0"
  instance_type = "t2.micro"
  key_name = "keypair-ohio-terraform"
  tags = {
    Name = "assignment-2"
  }
}

resource "aws_eip" "eip" {
  vpc = true
}

resource "aws_eip_association" "eip_assoc" {
  instance_id = aws_instance.assignment-2.id
  allocation_id = aws_eip.eip.id
}

^G Get Help   ^O Write Out   ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo   M-A Mark Text   M-] To Bracket
^X Exit      ^R Read File   ^\ Replace   ^U Paste Text  ^I To Spell   ^_ Go To Line  M-B Redo   M-C Copy Text   ^O Where Was

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

```
ubuntu@ip-172-31-15-128:~$ sudo nano assign1.tf
ubuntu@ip-172-31-15-128:~$ terraform init
```

```
Initializing the backend...
Initializing provider plugins...
  - Reusing previous version of hashicorp/aws from the dependency lock file
  - Using previously-installed hashicorp/aws v5.46.0

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.
ubuntu@ip-172-31-15-128:~$ terraform plan
```

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

```

+ 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: 3 to add, 0 to change, 0 to destroy.

Warning: Argument is deprecated

with aws_eip.eip,
on assign1.tf line 17, in resource "aws_eip" "eip":
17:   vpc = true

use domain attribute instead

(and one more similar warning elsewhere)

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
ubuntu@ip-172-31-15-128:~$ terraform apply

```

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

```

Only 'yes' will be accepted to approve.

Enter a value: yes

aws_instance.assignment-2: Creating...
aws_eip.eip: Creating...
aws_eip.eip: Creation complete after 1s [id=eipalloc-0e37e685f4891c4e7]
aws_instance.assignment-2: Still creating... [10s elapsed]
aws_instance.assignment-2: Still creating... [20s elapsed]
aws_instance.assignment-2: Still creating... [30s elapsed]
aws_instance.assignment-2: Creation complete after 32s [id=i-0f806abe13c1a0086]
aws_eip_association.eip_assoc: Creating...
aws_eip_association.eip_assoc: Creation complete after 1s [id=eipassoc-0b1a7a1b4a1c23ad]

Warning: Argument is deprecated

with aws_eip.eip,
on assign1.tf line 17, in resource "aws_eip" "eip":
17:   vpc = true

use domain attribute instead

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-15-128:~$ 

```

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

The screenshot shows the AWS CloudShell interface with the following content:

```

aws Services Search [Alt+S] Instances (1) Info Connect Instance state Actions Launch instances
Find Instance by attribute or tag (case-sensitive) All states
Instance ID = i-0f806abe13c1a0086 Clear filters
Name Instance ID Instance state Instance type Status check Alarm status
assignment-2 i-0f806abe13c1a0086 Running t2.micro Initializing View alarms
Select an instance

```

The left sidebar shows the navigation menu for EC2 services, including EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, Reservations (New), Images (selected), and AMIs.

EC2 Dashboard			
EC2 Global View	X	i-0f806abe13c1a0086 (assignment-2)	3.134.150.134   open address <a href="#">🔗</a>
Events		IPv6 address	172.31.13.197
Instances		Instance state	Public IPv4 DNS
Instances		Running	<a href="#">🔗</a>
Instance Types		Hostname type	ec2-3-134-150-134.us-east-2.compute.amazonaws.com   open address <a href="#">🔗</a>
Launch Templates		IP name: ip-172-31-13-197.us-east-2.compute.internal	
Spot Requests		Private IP DNS name (IPv4 only)	
Savings Plans		<a href="#">🔗</a>	
Reserved Instances		ip-172-31-13-197.us-east-2.compute.internal	
Dedicated Hosts		Answer private resource DNS name	Elastic IP addresses
Capacity		-	<a href="#">🔗</a>
Reservations <a href="#">New</a>		Instance type	3.134.150.134 [Public IP]
Images		t2.micro	
AMIs		Auto-assigned IP address	AWS Compute Optimizer finding
IAM Role		-	<a href="#">🔗</a> Opt-in to AWS Compute Optimizer for recommendations.
		VPC ID	Learn more <a href="#">🔗</a>
		<a href="#">🔗</a>	
		vpc-0cf7f5be256d8dbd6 <a href="#">🔗</a>	
		Subnet ID	Auto Scaling Group name
		-	

[CloudShell](#) [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookies](#)

# Assignment – 3

## Task to be performed

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

## Steps and Commands

- To destroy the previous deployment resource which was created in assignment 2 by running the command **terraform destroy**
- And also provide **yes** for confirmation.
- Next open the file which was created in the assignment 1 for providing script for creating resource by running the command **sudo nano (provide the name of created file.tf)**
- Edit the necessary script in file to create 2 EC2 instance in Ohio and N.Virginia region
- save and exit from the file
- To see the file by running the command **ls**
- Next to initialize terraform by running the command **terraform init**
- And next run the command **terraform plan**
- To create the necessary resource by running the command **terraform apply**
- And also provide **yes** for confirmation.
- Now the new 2 EC2 instance will be created in Ohio and N.Virginia region in the name of Ohio's instance as 'hello-ohio' and N.Virginia's instance as 'hello-virginia' by the terraform.

```
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_instance.assignment-2: Creating...
aws_eip.eip: Creating...
aws_eip.eip: Creation complete after 1s [id=eipalloc-0e37e685f4891c4e7]
aws_instance.assignment-2: Still creating... [10s elapsed]
aws_instance.assignment-2: Still creating... [20s elapsed]
aws_instance.assignment-2: Still creating... [30s elapsed]
aws_instance.assignment-2: Creation complete after 32s [id=i-0f806abel3c1a0086]
aws_eip_association.eip_assoc: Creating...
aws_eip_association.eip_assoc: Creation complete after 1s [id=eipassoc-0b1a7a1b4alc232ad]

Warning: Argument is deprecated
          with aws_eip.eip,
          on assignn1.tf line 17, in resource "aws_eip" "eip":
          17:   vpc = true

use domain attribute instead

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-15-128:~$ terraform destroy
```

```

on assign1.tf line 17, in resource "aws_eip" "eip":
17:   vpc = true

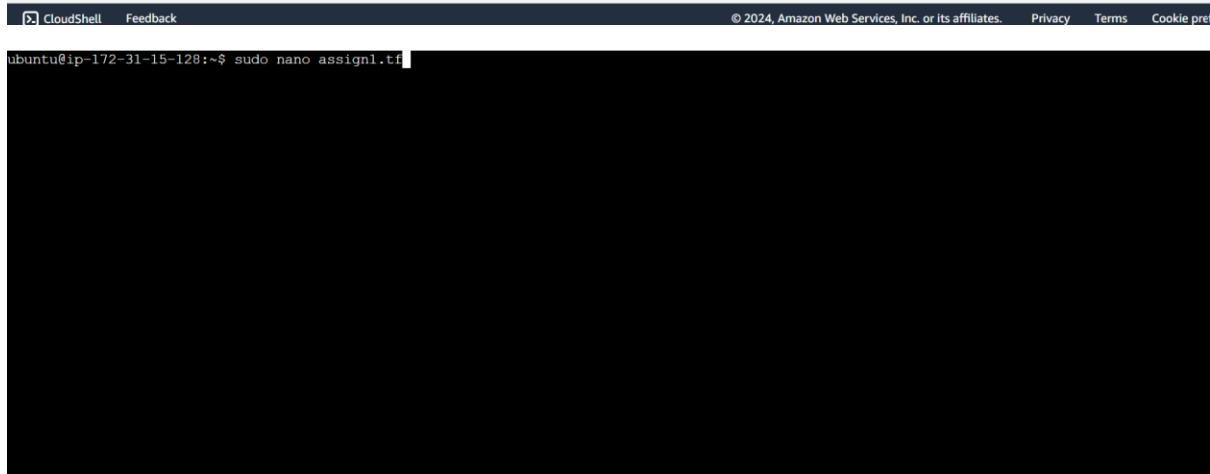
use domain attribute instead

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_association.eip_assoc: Destroying... [id=eipassoc-0b1a7a1b4a1c232ad]
aws eip_association.eip_assoc: Destruction complete after 1s
aws_instance.assignment-2: Destroying... [id=i-0f806abe13c1a0086]
aws_eip.eip: Destroying... [id=eipalloc-0e37e685f4891c4e7]
aws_eip.eip: Destruction complete after 1s
aws_instance.assignment-2: Still destroying... [id=i-0f806abe13c1a0086, 10s elapsed]
aws_instance.assignment-2: Still destroying... [id=i-0f806abe13c1a0086, 20s elapsed]
aws_instance.assignment-2: Still destroying... [id=i-0f806abe13c1a0086, 30s elapsed]
aws_instance.assignment-2: Still destroying... [id=i-0f806abe13c1a0086, 40s elapsed]
aws_instance.assignment-2: Destruction complete after 41s

Destroy complete! Resources: 3 destroyed.
ubuntu@ip-172-31-15-128:~$ 
```



```

GNU nano 4.8                                         assign1.tf                                         Modified
provider "aws"{
  alias = "Ohio"
  region = "us-east-2"
  access_key = "AKIA47CRV22SMCWBLXVL"
  secret_key = "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"
}

provider "aws"{
  alias = "NV"
  region = "us-east-1"
  access_key = "AKIA47CRV22SMCWBLXVL"
  secret_key = "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"
}

resource "aws_instance" "assignment-3-1" {
  provider = aws.Ohio
  ami = "ami-0b4750268a88e78e0"
  instance_type = "t2.micro"
  key_name = "keypair-ohio-terraform"
  tags = {
    Name = "hello-ohio"
  }
}

^G Get Help  ^O Write Out  ^W Where Is  ^R Cut Text  ^J Justify  ^C Cur Pos  M-U Undo  M-A Mark Text  M-] To Bracket
^X Exit      ^R Read File  ^\ Replace   ^U Paste Text  ^T To Spell   ^_ Go To Line  M-E Redo   M-C Copy Text  ^Q Where Was 
```

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

```
GNU nano 4.8                               assign1.tf                         Modified
resource "aws_instance" "assignment-3-1" {
  provider = aws.Ohio
  ami      = "ami-0b4750260a88e78e0"
  instance_type = "t2.micro"
  key_name = "keypair-ohio-terraform"
  tags = {
    Name = "hello-ohio"
  }
}

resource "aws_instance" "assignment-3-2" {
  provider = aws.NV
  ami      = "ami-080e1f13689e07408"
  instance_type = "t2.micro"
  key_name = "new"
  tags = {
    Name = "hello-virginia"
  }
}

^G Get Help   ^O Write Out   ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo   M-A Mark Text   M-[ To Bracket
^X Exit      ^R Read File   ^\ Replace    ^U Paste Text  ^I To Spell   ^_ Go To Line  M-B Redo   M-C Copy Text  ^O Where Was


```

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

```
ubuntu@ip-172-31-15-128:~$ sudo nano assign1.tf
ubuntu@ip-172-31-15-128:~$ terraform init

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.46.0

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.
ubuntu@ip-172-31-15-128:~$ terraform plan
```

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

```
+ secondary_private_ips          = (known after apply)
+ security_groups                = (known after apply)
+ source_dest_check              = true
+ spot_instance_request_id       = (known after apply)
+ subnet_id                      = (known after apply)
+ tags                           = {
    + "Name" = "hello-virginia"
  }
+ tags_all                       = {
    + "Name" = "hello-virginia"
  }
+ 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.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
ubuntu@ip-172-31-15-128:~$ terraform apply
```

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

```

+ 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.assignment-3-1: Creating...
aws_instance.assignment-3-2: Creating...
aws_instance.assignment-3-1: Still creating... [10s elapsed]
aws_instance.assignment-3-2: Still creating... [10s elapsed]
aws_instance.assignment-3-1: Still creating... [20s elapsed]
aws_instance.assignment-3-2: Still creating... [20s elapsed]
aws_instance.assignment-3-1: Still creating... [30s elapsed]
aws_instance.assignment-3-2: Still creating... [30s elapsed]
aws_instance.assignment-3-1: Creation complete after 32s [id=i-085b8af8ac87f479f]
aws_instance.assignment-3-2: Creation complete after 32s [id=i-00c08c5bf27ad9aa2]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-15-128:~$ 

```

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

AWS Services Search [Alt+S]

Instances (1) Info Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive)

Instance ID = i-085b8af8ac87f479f

Name Instance ID Instance state Instance type Status check Alarm status

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
hello-ohio	i-085b8af8ac87f479f	Running	t2.micro	Initializing	<a href="#">View alarm</a>

Select an instance

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookies

AWS Services Search [Alt+S]

Instances (1) Info Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive)

All states

Name Instance ID Instance state Instance type Status check Alarm status

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
hello-virginia	i-00c08c5bf27ad9aa2	Running	t2.micro	Initializing	<a href="#">View alarms</a>

Select an instance

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookies

## Assignment – 4

### Task to be performed

1. Destroy the previous deployment
2. Create a VPC with the required components using Terraform
3. Deploy an EC2 instance inside the VPC

### Steps and Commands

- To destroy the previous deployment resource which was created in assignment 3 by running the command **terraform destroy**
- And also provide **yes** for confirmation.
- Next open the file which was created in the assignment 1 for providing script for creating resource by running the command **sudo nano (provide the name of created file.tf)**
- Edit the necessary script in file to create a EC2 instance inside the VPC
- save and exit from the file
- To see the file by running the command **ls**
- Next to initialize terraform by running the command **terraform init**
- And next run the command **terraform plan**
- To create the necessary resource by running the command **terraform apply**
- And also provide **yes** for confirmation.
- Now the EC2 instance will be created inside the VPC by the terraform.

```
+ 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.assignment-3-1: Creating...
aws_instance.assignment-3-2: Creating...
aws_instance.assignment-3-1: Still creating... [10s elapsed]
aws_instance.assignment-3-2: Still creating... [10s elapsed]
aws_instance.assignment-3-1: Still creating... [20s elapsed]
aws_instance.assignment-3-2: Still creating... [20s elapsed]
aws_instance.assignment-3-1: Still creating... [30s elapsed]
aws_instance.assignment-3-2: Still creating... [30s elapsed]
aws_instance.assignment-3-1: Creation complete after 32s [id=i-085b8af8ac87f479f]
aws_instance.assignment-3-2: Creation complete after 32s [id=i-00c08c5bf27ad9aa2]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-15-128:~$ terraform destroy
```

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

```
}
```

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_instance.assignment-3-1: Destroying... [id=i-085b8af8ac87f479f]
aws_instance.assignment-3-2: Destroying... [id=i-00c08c5bf27ad9aa2]
aws_instance.assignment-3-1: Still destroying... [id=i-085b8af8ac87f479f, 10s elapsed]
aws_instance.assignment-3-2: Still destroying... [id=i-00c08c5bf27ad9aa2, 10s elapsed]
aws_instance.assignment-3-1: Still destroying... [id=i-085b8af8ac87f479f, 20s elapsed]
aws_instance.assignment-3-2: Still destroying... [id=i-00c08c5bf27ad9aa2, 20s elapsed]
aws_instance.assignment-3-1: Still destroying... [id=i-085b8af8ac87f479f, 30s elapsed]
aws_instance.assignment-3-2: Still destroying... [id=i-00c08c5bf27ad9aa2, 30s elapsed]
aws_instance.assignment-3-1: Still destroying... [id=i-085b8af8ac87f479f, 40s elapsed]
aws_instance.assignment-3-2: Destruction complete after 40s
aws_instance.assignment-3-1: Still destroying... [id=i-00c08c5bf27ad9aa2, 40s elapsed]
aws_instance.assignment-3-2: Destruction complete after 40s

Destroy complete! Resources: 2 destroyed.
```

ubuntu@ip-172-31-15-128:~\$

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

```
ubuntu@ip-172-31-15-128:~$ sudo nano assign1.tf
```

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

```
provider "aws" {
    region = "us-east-2"
    access_key = "AKIA47CRV22SMCWBLXVL"
    secret_key = "lsQhcC2+/3yj5KtrrdUEE+NLjiuIjKi+Jv/tOFmT2"
}

resource "aws_instance" "assignment-4" {
    ami = "ami-0b4750268a88e78e0"
    instance_type = "t2.micro"
    subnet_id = aws_subnet.assignment-4-subnet.id
    key_name = "keypair-ohio-terraform"
    tags = {
        Name = "assignment-4"
    }
}

resource "aws_vpc" "assignment-4-vpc" {
    cidr_block = "10.0.0.0/16"
    tags = {
        Name = "assignment-4-vpc"
    }
}
```

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo M-A Mark Text M-J To Bracket  
^X Exit ^R Read File ^V Replace ^U Paste Text ^T To Spell ^\_ Go To Line M-E Redo M-C Copy Text ^Q Where Was

```
GNU nano 4.8                                     assign1.tf                                         Modified
tags = {
  Name = "assignment-4"
}

resource "aws_vpc" "assignment-4-vpc" {
  cidr_block = "10.0.0.0/16"
  tags = {
    Name = "assignment-4-vpc"
  }
}

resource "aws_subnet" "assignment-4-subnet" {
  vpc_id = aws_vpc.assignment-4-vpc.id
  cidr_block = "10.0.128.0/17"
  availability_zone = "us-east-2a"
  tags = {
    Name = "assignment-4-subnet"
  }
}

^G Get Help  ^O Write Out  ^W Where Is  ^F Cut Text  ^J Justify  ^C Cur Pos  M-U Undo  M-A Mark Text  M-J To Bracket
^X Exit     ^R Read File  ^V Replace   ^U Paste Text  ^I To Spell   ^L Go To Line  M-E Redo  M-G Copy Text  ^Q Where Was
```

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

```
ubuntu@ip-172-31-15-128:~$ sudo nano assign1.tf
ubuntu@ip-172-31-15-128:~$ terraform init

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.46.0

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.
ubuntu@ip-172-31-15-128:~$ terraform plan
```

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

```
+ enable_dns_hostnames          = (known after apply)
+ enable_dns_support            = true
+ enable_network_address_usage_metrics = (known after apply)
+ id                            = (known after apply)
+ instance_tenancy              = "default"
+ ipv6_association_id          = (known after apply)
+ ipv6_cidr_block               = (known after apply)
+ ipv6_cidr_block_network_border_group = (known after apply)
+ main_route_table_id           = (known after apply)
+ owner_id                      = (known after apply)
+ tags                          = {
    + "Name" = "assignment-4-vpc"
  }
+ tags_all                      = {
    + "Name" = "assignment-4-vpc"
  }
}

Plan: 3 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
ubuntu@ip-172-31-15-128:~$ terraform apply
```

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

```

+ tags_all = {
+   "Name" = "assignment-4-vpc"
}
}

Plan: 3 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_vpc.assignment-4-vpc: Creating...
aws_vpc.assignment-4-vpc: Creation complete after 1s [id=vpc-075db2c8814a7cc1c]
aws_subnet.assignment-4-subnet: Creating...
aws_subnet.assignment-4-subnet: Creation complete after 1s [id=subnet-0694b4be8cd219dd3]
aws_instance.assignment-4: Creating...
aws_instance.assignment-4: Still creating... [10s elapsed]
aws_instance.assignment-4: Still creating... [20s elapsed]
aws_instance.assignment-4: Still creating... [30s elapsed]
aws_instance.assignment-4: Creation complete after 31s [id=i-035de281501352ce9]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-15-128:~$ 

```

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

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations. Below that is another section for Images and AMIs. The main content area has a heading 'Instances (1) Info' with a search bar and filters for Instance ID, Name, Instance state, Instance type, Status check, and Alarm status. A table lists one instance: 'assignment-4' with Instance ID 'i-035de281501352ce9', which is 'Running'. It also shows the instance type as 't2.micro', status as 'Initializing', and alarm status as 'View alarms'. At the bottom, there's a 'Select an instance' button.

The screenshot shows the AWS EC2 Instance Details page for the instance 'i-035de281501352ce9 (assignment-4)'. The sidebar is identical to the previous screenshot. The main content area displays various details for the instance, such as its IPv6 address (empty), instance state (Running), private IP DNS name (ip-10-0-252-90.us-east-2.compute.internal), instance type (t2.micro), VPC ID (vpc-075db2c8814a7cc1c), subnet ID (subnet-0694b4be8cd219dd3), and IAM Role (empty). It also includes sections for Auto-assigned IP address, Answer private resource DNS name, and IMDSv2 Optional. A note at the bottom right encourages opt-in to AWS Compute Optimizer.

AWS Services Search [Alt+S] Ohio

VPC dashboard EC2 Global View Filter by VPC: Select a VPC

Virtual private cloud Your VPCs (1/1) Info

vpc-075db2c8814a7cc1c X Clear filters

Name	VPC ID	State	IPv4 CIDR
assignment-4-vpc	vpc-075db2c8814a7cc1c	Available	10.0.0.0/16

vpc-075db2c8814a7cc1c / assignment-4-vpc

Details Resource map CIDRs Flow logs Tags Integrations

Details

VPC ID	State	DNS hostnames	DNS resolution
vpc-075db2c8814a7cc1c	Available	Disabled	Enabled

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookies CloudShell Feedback

AWS Services Search [Alt+S] Ohio

VPC dashboard EC2 Global View Filter by VPC: Select a VPC

Virtual private cloud Your VPCs Subnets

Subnets (1) Info

VPC : vpc-075db2c8814a7cc1c X Clear filters

Name	Subnet ID	State	VPC
assignment-4-subnet	subnet-0694b4be8cd219dd	Available	vpc-075db2c8814a7cc1c

Select a subnet

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookies CloudShell Feedback

## Assignment – 5

### Task to be performed

1. Destroy the previous deployment
2. Create a Script to install Apache2
3. Run this script on a newly created EC2 instance
4. Print the IP address of the instance in a file on the local once deployed

### Steps and Commands

- To destroy the previous deployment resource which was created in assignment 4 by running the command **terraform destroy**
- And also provide **yes** for confirmation.
- Next create a file to install Apache2 by running the command `sudo nano apache2.sh`
- Provide the necessary command to install Apache2
- Next save and exit from the Apache2 file
- Next open the file which was created in the assignment 1 for providing script for creating resource by running the command **sudo nano (provide the name of created file.tf)**
- Edit the necessary script in file to create a EC2 instance and mention the Apache2 file which was created and also to print the IP address of the instance
- save and exit from the file
- To see the file by running the command **ls**
- Next to initialize terraform by running the command **terraform init**
- And next run the command **terraform plan**
- To create the necessary resource by running the command **terraform apply**
- And also provide **yes** for confirmation.
- Now the EC2 instance will be created with Apache2 with installed. If we put the public IP address of instance in browser the custom page will be reflected and also the Public IP address of newly created instance will be printed and reflected in machine by the terraform.

```
+ tags_all = {
+   "Name" = "assignment-4-vpc"
}

Plan: 3 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_vpc.assignment-4-vpc: Creating...
aws_vpc.assignment-4-vpc: Creation complete after 1s [id=vpc-075db2c8814a7cc1c]
aws_subnet.assignment-4-subnet: Creating...
aws_subnet.assignment-4-subnet: Creation complete after 1s [id=subnet-0694b4be8cd219dd3]
aws_instance.assignment-4: Creating...
aws_instance.assignment-4: Still creating... [10s elapsed]
aws_instance.assignment-4: Still creating... [20s elapsed]
aws_instance.assignment-4: Still creating... [30s elapsed]
aws_instance.assignment-4: Creation complete after 31s [id=i-035de281501352ce9]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-15-128:~$ terraform destroy
```

CloudShell Feedback

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

```
"Name" = "assignment-4-vpc"
} -> null
# (4 unchanged attributes hidden)

Plan: 0 to add, 0 to change, 3 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_instance.assignment-4: Destroying... [id=i-035de281501352ce9]
aws_instance.assignment-4: Still destroying... [id=i-035de281501352ce9, 10s elapsed]
aws_instance.assignment-4: Still destroying... [id=i-035de281501352ce9, 20s elapsed]
aws_instance.assignment-4: Still destroying... [id=i-035de281501352ce9, 30s elapsed]
aws_instance.assignment-4: Destruction complete after 39s
aws_subnet.assignment-4-subnet: Destroying... [id=subnet-0694b4be8cd219dd3]
aws_subnet.assignment-4-subnet: Destruction complete after 1s
aws_vpc.assignment-4-vpc: Destroying... [id=vpc-075db2c8814a7cc1c]
aws_vpc.assignment-4-vpc: Destruction complete after 0s

Destroy complete! Resources: 3 destroyed.
ubuntu@ip-172-31-15-128:~$
```

CloudShell Feedback

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

```
ubuntu@ip-172-31-15-128:~$ sudo nano apache2.sh
```

CloudShell Feedback

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

```
GNU nano 4.8                                         apache2.sh                                         Modified
#!/bin/bash
sudo apt update -y
sudo apt install apache2 -y
sudo su
echo "Hello World" > /var/www/html/index.html

^G Get Help   ^O Write Out   ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo   M-A Mark Text   M-] To Bracket
^X Exit      ^R Read File   ^V Replace    ^U Paste Text  ^T To Spell   ^_ Go To Line  M-B Redo   M-G Copy Text  ^Q Where Was

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

```
ubuntu@ip-172-31-15-128:~$ sudo nano apache2.sh
ubuntu@ip-172-31-15-128:~$ sudo nano assign1.tf
```

```
GNU nano 4.8                                         assign1.tf                                         Modified
provider "aws"{
  region = "us-east-2"
  access_key = "AKIA47CRV22SMCWBLXVL"
  secret_key = "lSQhcC2+/3yj5KtrrdUEE+NLjiuIjKi+Jv/tOFmT2"
}

resource "aws_instance" "assignment-5" {
  ami = "ami-0b4750268a88e78e0"
  instance_type = "t2.micro"
  key_name = "keypair-ohio-terraform"
  user_data = "${file("apache2.sh")}"
  tags = {
    Name = "assignment-5"
  }
}

output "IPv4" {
  value = aws_instance.assignment-5.public_ip
}

^G Get Help   ^O Write Out   ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo   M-A Mark Text   M-] To Bracket
^X Exit      ^R Read File   ^V Replace    ^U Paste Text  ^T To Spell   ^_ Go To Line  M-B Redo   M-G Copy Text  ^Q Where Was

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

```

ubuntu@ip-172-31-15-128:~$ sudo nano apache2.sh
ubuntu@ip-172-31-15-128:~$ sudo nano assign1.tf
ubuntu@ip-172-31-15-128:~$ terraform init

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.46.0

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.
ubuntu@ip-172-31-15-128:~$ terraform plan

```

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```

+ spot_instance_request_id      = (known after apply)
+ subnet_id                     = (known after apply)
+ tags                          = {
    + "Name" = "assignment-5"
}
+ tags_all                      = {
    + "Name" = "assignment-5"
}
+ tenancy                       = (known after apply)
+ user_data                      = "408260431c6d918e5365f251103d1864064aa72d"
+ user_data_base64              = (known after apply)
+ user_data_replace_on_change   = false
+ vpc_security_group_ids        = (known after apply)
}

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

Changes to Outputs:
+ IPv4 = (known after apply)

```

Note: You didn't use the `-out` option to save this plan, so Terraform can't guarantee to take exactly these actions if you run `"terraform apply"` now.  
ubuntu@ip-172-31-15-128:~\$ terraform apply

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```

}

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

Changes to Outputs:
+ IPv4 = (known after apply)

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.assignment-5: Creating...
aws_instance.assignment-5: Still creating... [10s elapsed]
aws_instance.assignment-5: Still creating... [20s elapsed]
aws_instance.assignment-5: Still creating... [30s elapsed]
aws_instance.assignment-5: Creation complete after 32s [id=i-0102d5b269f79504b]

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

Outputs:

IPv4 = "18.225.57.142"
ubuntu@ip-172-31-15-128:~$ 
```

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

AWS Services Search [Alt+S] Ohio

EC2 Dashboard EC2 Global View Events Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Capacity Reservations New Images AMIs

Instances (1) Info Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive) All states

Instance ID = i-0102d5b269f79504b Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
assignment-5	i-0102d5b269f79504b	Running	t2.micro	2/2 checks passed	<a href="#">View alarms</a>

Select an instance

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookies

aws Services Search [Alt+S] Ohio

EC2 Dashboard EC2 Global View Events Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Capacity Reservations New Images AMIs

EC2 > Instances > i-0102d5b269f79504b

Instance summary for i-0102d5b269f79504b (assignment-5) Info

C Connect Instance state Actions

Updated less than a minute ago

Instance ID i-0102d5b269f79504b (assignment-5)	Public IPv4 address 18.225.57.142   open address	Private IPv4 addresses 172.31.3.190
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-18-225-57-142.us-east-2.compute.amazonaws.com   open address
Hostname type IP name: ip-172-31-3-190.us-east-2.compute.internal	Private IP DNS name (IPv4 only) ip-172-31-3-190.us-east-2.compute.internal	Elastic IP addresses -
Answer private resource DNS name -	Instance type t2.micro	

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookies

← → C △ Not secure 18.225.57.142

My Courses | Nave... Cloud Computing... Railway Recruitme... Notices | Staff Sele... Visual Subnet Calc... Microsoft Azure Po...

Incognito All Bookmarks

Hello World