

Tasks To Be Performed:

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

1. Destroy the previous deployments

```
apply complete! Resources: 2 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-28-242:~$ terraform destroy
aws_instance.NV-instance: Refreshing state... [id=i-050ab0975b40bd8d9]
aws_instance.oregon-instance: Refreshing state... [id=i-0f36a0c0145403dca]

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_instance.NV-instance will be destroyed
- resource "aws_instance" "NV-instance" {
    - ami = "ami-0cae6d6fe6048ca2c" => null
    - arn = "arn:aws:ec2::us-east-1:251985476962:instance/i-050ab0975b40bd8d9" => null
    - associate_public_ip_address = true => null
    - availability_zone = "us-east-1b" => null
    - disable_api_stop = false => null
    - disable_api_termination = false => null
    - ebs_optimized = false => null
    - force_destroy = false => null
    - get_password_data = false => null
    - hibernation = false => null
    - id = "i-050ab0975b40bd8d9" => null
    - instance_initiated_shutdown_behavior = "stop" => null
    - instance_state = "running" => null
    - instance_type = "t2.micro" => null
    - ipv6_address_count = 0 => null
    - ipv6_addresses = [] => null
    - key_name = "terra-kp" => null
    - monitoring = false => null
    - placement_partition_number = 0 => null
    - primary_network_interface_id = "eni-01d7e8fbf5ea210c" => null
    - private_dns = "ip-172-31-23-5.ec2.internal" => null
    - private_ip = "172.31.23.5" => null
}

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.NV-instance: Destroying... [id=i-050ab0975b40bd8d9]
aws_instance.oregon-instance: Destroying... [id=i-0f36a0c0145403dca]
aws_instance.NV-instance: Still destroying... [id=i-050ab0975b40bd8d9, 00m10s elapsed]
aws_instance.oregon-instance: Still destroying... [id=i-0f36a0c0145403dca, 00m10s elapsed]
aws_instance.NV-instance: Still destroying... [id=i-050ab0975b40bd8d9, 00m20s elapsed]
aws_instance.oregon-instance: Still destroying... [id=i-0f36a0c0145403dca, 00m20s elapsed]
aws_instance.NV-instance: Still destroying... [id=i-050ab0975b40bd8d9, 00m30s elapsed]
aws_instance.oregon-instance: Still destroying... [id=i-0f36a0c0145403dca, 00m30s elapsed]
aws_instance.oregon-instance: Destruction complete after 30s
aws_instance.NV-instance: Still destroying... [id=i-050ab0975b40bd8d9, 00m40s elapsed]
aws_instance.NV-instance: Still destroying... [id=i-050ab0975b40bd8d9, 00m50s elapsed]
aws_instance.NV-instance: Destruction complete after 1m0s

Destroy complete! Resources: 2 destroyed.
ubuntu@ip-172-31-28-242:~$
```

2. Create a VPC with the required components using Terraform

```
GNU nano 6.2
provider "aws"{
    region = "us-east-1"
    access_key = "AKIATVK4BHVRIRE5B5OU"
    secret_key = "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"
}
resource "aws_instance" "task-4"{
    ami = "ami-0cae6d6fe6048ca2c"
    instance_type = "t2.micro"
    associate_public_ip_address = true
    key_name = "terra-kp"
    tags = {
        Name = "work-1"
    }
}

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

resource "aws_internet_gateway" "task-4-igw" {
    vpc_id = aws_vpc.task-4-vpc.id
}

resource "aws_route_table" "task-4-rt" {
    vpc_id = aws_vpc.task-4-vpc.id

    route {
        cidr_block = "0.0.0.0/0"
        gateway_id = aws_internet_gateway.task-4-igw.id
    }

    tags = {
        Name = "main-rt"
    }
}

resource "aws_route_table_association" "rt"{
    subnet_id = aws_subnet.task-4-subnet.id
    route_table_id = aws_route_table.task-4-rt.id
}
```

```
ubuntu@ip-172-31-28-242:~$ 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.task-4 will be created
+ resource "aws_instance" "task-4" {
  ami                               = "ami-0cae6d6fe6048ca2c"
  arn                             = (known after apply)
  associate_public_ip_address      = true
  availability_zone                = (known after apply)
  disable_api_stop                 = (known after apply)
  disable_api_termination          = (known after apply)
  ebs_optimized                    = (known after apply)
  enable_primary_ipv6              = (known after apply)
  force_destroy                    = false
  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                         = "terra-kp"
  monitoring                       = (known after apply)
  outpost_arn                      = (known after apply)
  password_data                   = (known after apply)
  + placement_group
  - placement_group
```

```
# aws_vpc.task-4-vpc will be created
+ resource "aws_vpc" "task-4-vpc" {
    + arn                               = (known after apply)
    + cidr_block                        = "10.0.0.0/16"
    + default_network_acl_id           = (known after apply)
    + default_route_table_id           = (known after apply)
    + default_security_group_id        = (known after apply)
    + dhcp_options_id                  = (known after apply)
    + 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)
    + region                            = "us-east-1"
    + tags
        + "Name" = "main-vpc"
    }
    + tags_all                         = {
        + "Name" = "main-vpc"
    }
}
```

Plan: 6 to add, 0 to change, 0 to destroy.

```
ubuntu@ip-172-31-28-242:~$ terraform apply

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

Terraform will perform the following actions:

# aws_instance.task-4 will be created
+ resource "aws_instance" "task-4" {
    + ami                               = "ami-0cae6d6fe6048ca2c"
    + arn                               = (known after apply)
    + associate_public_ip_address       = true
    + availability_zone                 = (known after apply)
    + disable_api_stop                  = (known after apply)
    + disable_api_termination           = (known after apply)
    + ebs_optimized                     = (known after apply)
    + enable_primary_ipv6               = (known after apply)
    + force_destroy                     = false
    + 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                          = "terra-kp"
    + monitoring                        = (known after apply)
    + outpost_arn                       = (known after apply)
    + password_data                     = (known after apply)
    + placement_group                   = (known after apply)
    + placement_group_id                = (known after apply)
```

```

Plan: 6 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.task-4: Creating...
aws_vpc.task-4-vpc: Creating...
aws_vpc.task-4-vpc: Creation complete after 2s [id=vpc-04e42c0416c9e4026]
aws_subnet.task-4-subnet: Creating...
aws_internet_gateway.task-4-igw: Creating...
aws_internet_gateway.task-4-igw: Creation complete after 0s [id=igw-0d5110650d3ea3c20]
aws_route_table.task-4-rt: Creating...
aws_subnet.task-4-subnet: Creation complete after 1s [id=subnet-05cac536d8614db94]
aws_route_table.task-4-rt: Creation complete after 1s [id=rtb-0ae3000091662bbae]
aws_route_table_association.rt: Creating...
aws_route_table_association.rt: Creation complete after 0s [id=rtbassoc-03e91d75df1d72ce3]
aws_instance.task-4: Still creating... [00m10s elapsed]
aws_instance.task-4: Creation complete after 13s [id=i-01a1fa97a666c74d9]

Apply complete! Resources: 6 added, 0 changed, 0 destroyed.

```

Screenshot of the AWS CloudFormation console showing the successful creation of a VPC and EC2 instance.

CloudFormation Stack Details:

- Stack Name:** work-1
- Stack ID:** i-01a1fa97a666c74d9
- Status:** Running
- Region:** t2.micro
- Creation Time:** Initializing
- View alarms +**

EC2 Instance Details:

- Instance ID:** i-01a1fa97a666c74d9
- Public IPv4 address:** 34.227.206.98
- Private IPv4 addresses:** 172.31.22.252
- Public DNS:** ec2-34-227-206-98.compute-1.amazonaws.com
- Instance state:** Running

Your VPCs (1/2) Info:

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
main-vpc	vpc-04e42c0416c9e4026	Available	Off	10.0.0.0/16	-
-	vpc-02196942317add486	Available	Off	172.31.0.0/16	-

VPC Details:

- VPC ID:** vpc-04e42c0416c9e4026 / main-vpc
- Details:**
 - VPC ID:** vpc-04e42c0416c9e4026
 - DNS resolution:** Enabled
 - Main network ACL:** acl-005027cc004d00-00d
 - State:** Available
 - Tenancy:** default
 - Default VPC:** -
- Block Public Access:** Off
- DHCP option set:** dopt-0a70ccc0ef04464d3
- IPv4 CIDR:** 10.0.0.0/16
- DNS hostnames:** Disabled
- Main route table:** rtb-083ba6fdcfdd5011e
- IPv6 pool:** -

Subnets (1/7) [Info](#)

Find subnets by attribute or tag

Name	Subnet ID	State
-	subnet-0c3e601c1bdf99c4d	Available
<input checked="" type="checkbox"/> Main-subnet	subnet-05cac536d8614db94	Available
-	subnet-0dffac280838582fa	Available
-	subnet-08d2972ac79bdd96d	Available
-	subnet-0a9ed3914328bcd7c	Available
-	subnet-0f611e218bcc81f3b	Available

subnet-05cac536d8614db94 / Main-subnet

[Details](#) [Flow logs](#) [Route table](#) [Network ACL](#) [CIDR reservations](#) [Sharing](#)

igw-0d5110650d3ea3c20 / igw

[Details](#) [Tags](#)

Details

Internet gateway ID igw-0d5110650d3ea3c20	State Attached	VPC ID vpc-04e42c0416c9e4026 main-vpc	Owner 251985476962
--	-------------------	--	---------------------------------------

Route tables (1/3) [Info](#)

Last updated 2 minutes ago

Find route tables by attribute or tag

Name	Route table ID	Explicit subnet associations	Edge associations
-	rtb-083ba6fdcfdd5011e	-	-
<input checked="" type="checkbox"/> main-rt	rtb-0ae3000091662bbae	subnet-05cac536d8614db94 / Main-subnet	-
-	rtb-0f611d2f4c036ab1c	subnet-012d590a3c6d19...	-

rtb-0ae3000091662bbae / main-rt

[Details](#) [Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

Details

Route table ID rtb-0ae3000091662bbae	Main <input type="checkbox"/> No	Explicit subnet associations subnet-05cac536d8614db94 / Main-subnet
VPC vpc-04e42c0416c9e4026 main-vpc	Owner ID 251985476962	

tb-0ae3000091662bbae / main-rt

Details | Routes | **Subnet associations** | Edge associations | Route propagation | Tags

Explicit subnet associations (1)

Name	Subnet ID	IPv4 CIDR
Main-subnet	subnet-05cac536d8614db94	10.0.1.0/24

Subnets without explicit associations (0)

3. Deploy an EC2 instance inside the VPC

```
GNU nano 6.2
provider "aws"{
    region = "us-east-1"
    access_key = "AKIATVK4BHVRIRE5B5OU"
    secret_key = "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"
}
resource "aws_instance" "task-4"{
    ami = "ami-0cae6d6fe6048ca2c"
    instance_type = "t2.micro"
    associate_public_ip_address = true
    key_name = "terra-kp"
    tags = {
        Name = "work-1"
    }
}

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

resource "aws_internet_gateway" "task-4-igw" {
    vpc_id = aws_vpc.task-4-vpc.id
}
```

Instances (1/2) [Info](#)

Last updated [\(C\)](#) ([Connect](#)) ([Instance state ▾](#)) ([Actions ▾](#)) ([Launch instances](#))

less than a minute ago

Find Instance by attribute or tag (case-sensitive)

All states ▾

Instance state = running [X](#) [Clear filters](#)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
terraform	i-070c9c1966c5aa13e	Running Q Q	t2.micro	2/2 checks passed	View alarms +	us-east-1b
work-1	i-01a1fa97a666c74d9	Running Q Q	t2.micro	2/2 checks passed	View alarms +	us-east-1b

[Unselect instance: work-1](#)