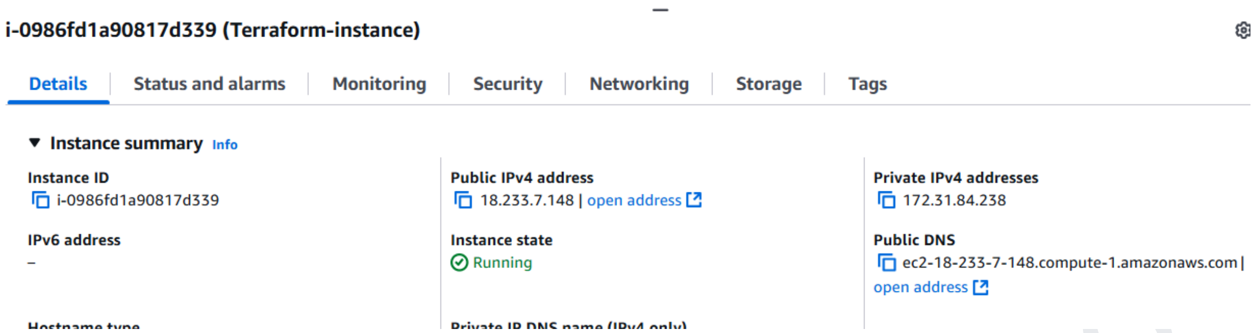
Project name: Infrastructure as a Code using AWS by Ms Minal jadhav

Creating an instance and installing Terraform in it



i-0986fd1a90817d339 (Terraform-instance)

▼ Instance summary

Instance ID: i-0986fd1a90817d339

Public IPv4 address: 18.233.7.148 | open address

Private IPv4 addresses: 172.31.84.238

IPv6 address: –

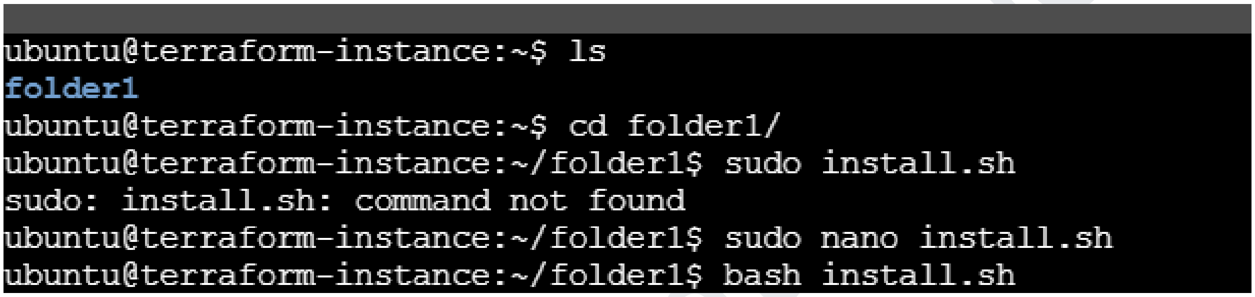
Instance state: Running

Public DNS: ec2-18-233-7-148.compute-1.amazonaws.com | open address

Hostname type

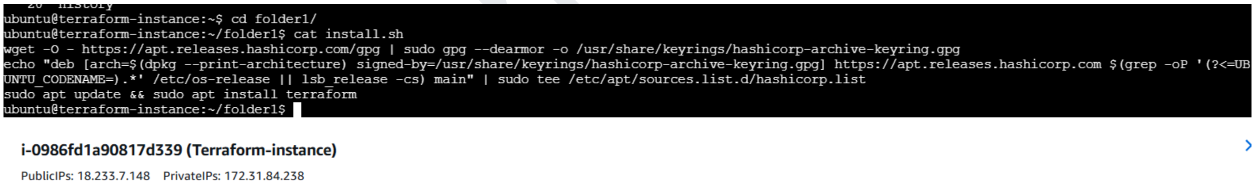
Private IP DNS name (IPv4 only)

Creating a directory named folder1 and the creating a file install.sh



```
ubuntu@terraform-instance:~$ ls
folder1
ubuntu@terraform-instance:~$ cd folder1/
ubuntu@terraform-instance:~/folder1$ sudo install.sh
sudo: install.sh: command not found
ubuntu@terraform-instance:~/folder1$ sudo nano install.sh
ubuntu@terraform-instance:~/folder1$ bash install.sh
```

Content in install.sh
https://developer.hashicorp.com/terraform/install#linux
Commands to install terraform was copied from above link

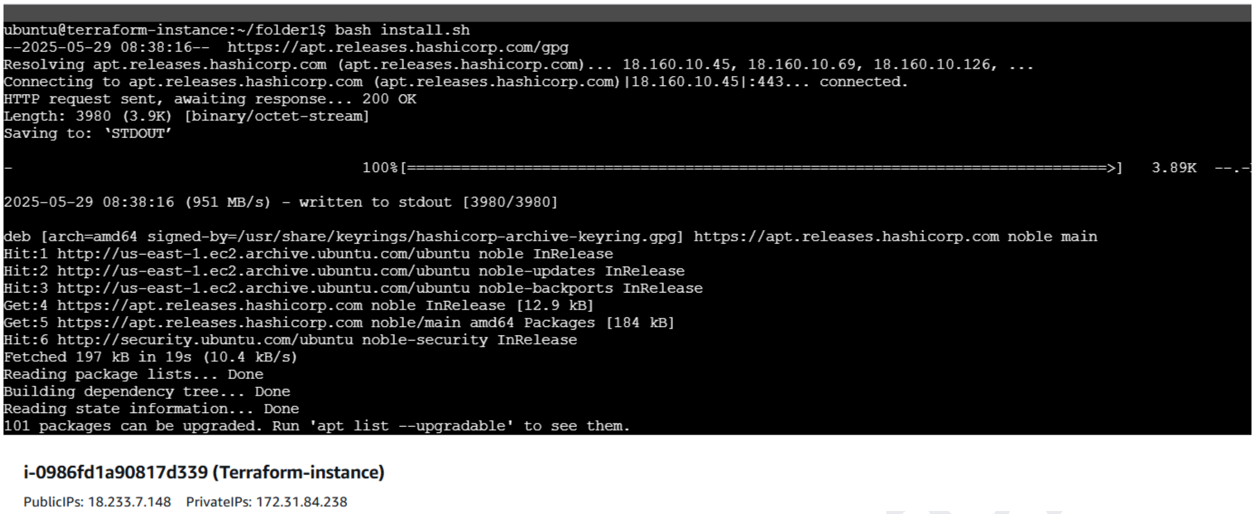


```
  20  history
ubuntu@terraform-instance:~$ cd folder1/
ubuntu@terraform-instance:~/folder1$ cat install.sh
wget -O - https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(grep -oP '(?<=UB
UNTU_CODENAME=).*' /etc/os-release || lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
sudo apt update && sudo apt install terraform
ubuntu@terraform-instance:~/folder1$
```

i-0986fd1a90817d339 (Terraform-instance)

PublicIPs: 18.233.7.148   PrivateIPs: 172.31.84.238

Installing terraform in instance

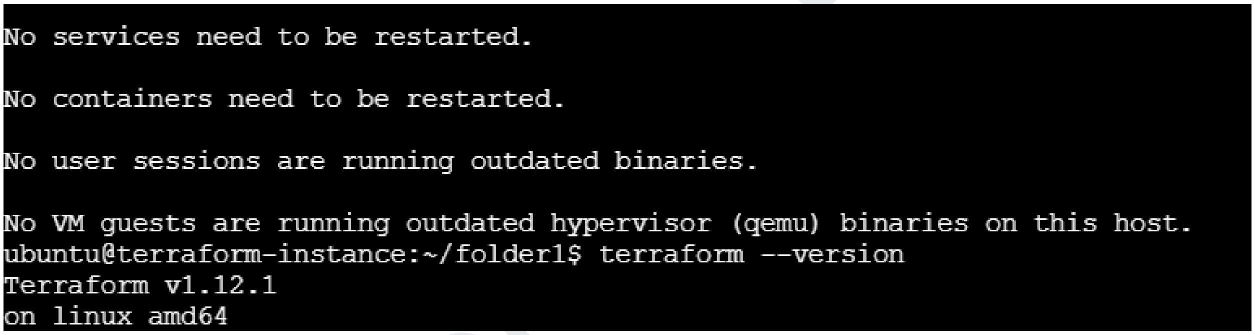


```
ubuntu@terraform-instance:~/folder1$ bash install.sh
--2025-05-29 08:38:16--  https://apt.releases.hashicorp.com/gpg
Resolving apt.releases.hashicorp.com (apt.releases.hashicorp.com)... 18.160.10.45, 18.160.10.69, 18.160.10.126, ...
Connecting to apt.releases.hashicorp.com (apt.releases.hashicorp.com)|18.160.10.45|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3980 (3.9K) [binary/octet-stream]
Saving to: 'STDOUT'

-                                        100%[===============================================================================>]   3.89K  --.-

2025-05-29 08:38:16 (951 MB/s) - written to stdout [3980/3980]

deb [arch=amd64 signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com noble main
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:4 https://apt.releases.hashicorp.com noble InRelease [12.9 kB]
Get:5 https://apt.releases.hashicorp.com noble/main amd64 Packages [184 kB]
Hit:6 http://security.ubuntu.com/ubuntu noble-security InRelease
Fetched 197 kB in 19s (10.4 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
101 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

i-0986fd1a90817d339 (Terraform-instance)

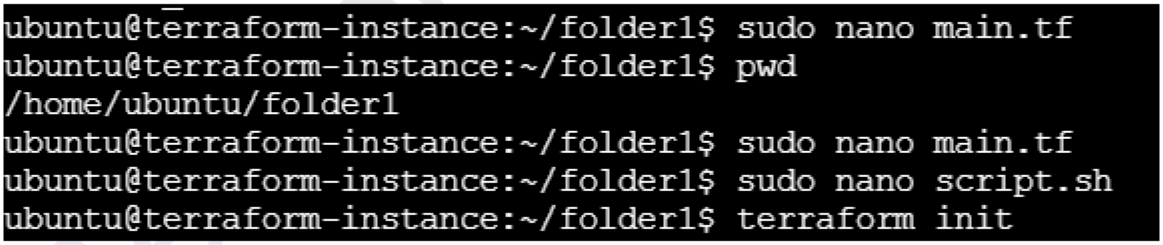
PublicIPs: 18.233.7.148   PrivateIPs: 172.31.84.238

Confirming terraform installation



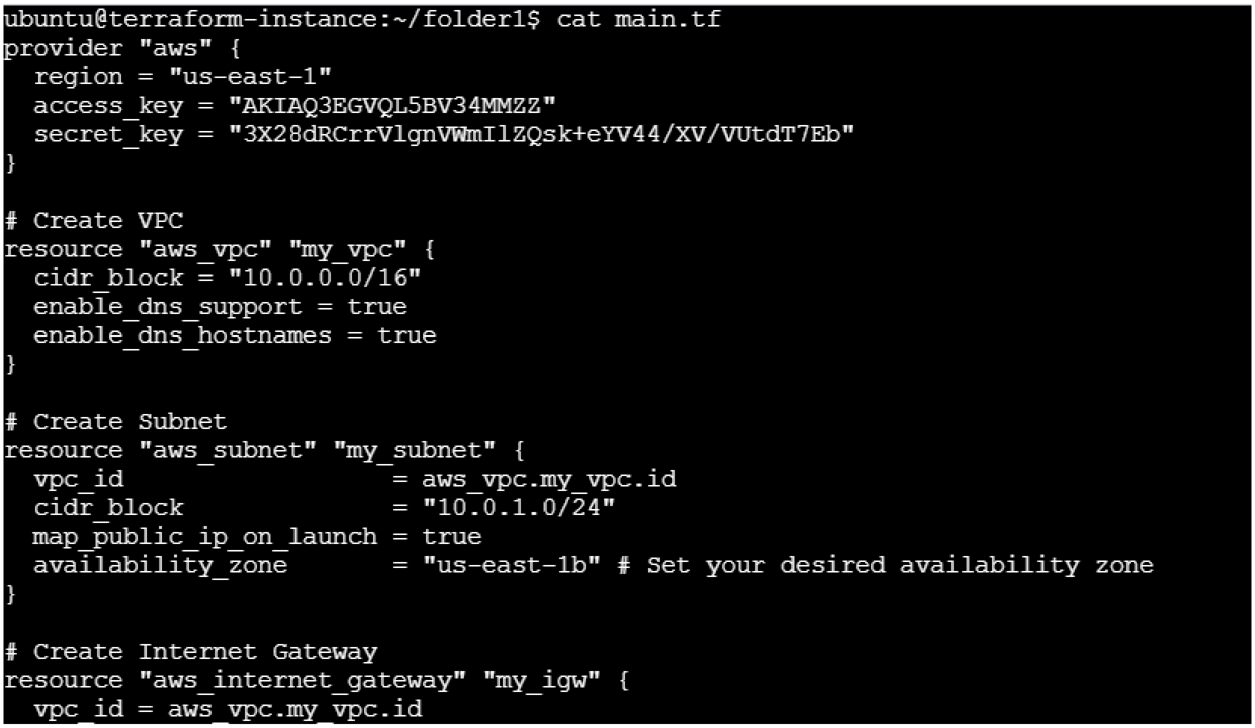
```
No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@terraform-instance:~/folder1$ terraform --version
Terraform v1.12.1
on linux amd64
```

Creating main.tf file to write code

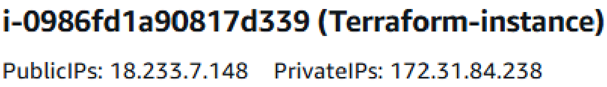


```
ubuntu@terraform-instance:~/folder1$ sudo nano main.tf
ubuntu@terraform-instance:~/folder1$ pwd
/home/ubuntu/folder1
ubuntu@terraform-instance:~/folder1$ sudo nano main.tf
ubuntu@terraform-instance:~/folder1$ sudo nano script.sh
ubuntu@terraform-instance:~/folder1$ terraform init
```

Content in main.tf (entire code is attached in github repo)



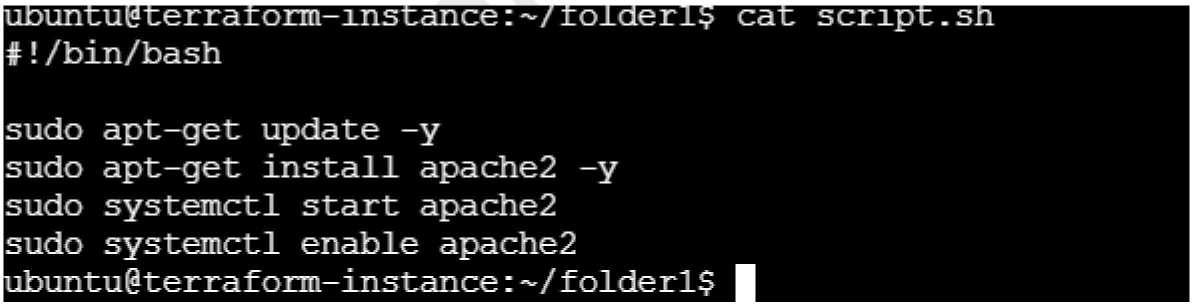
```
ubuntu@terraform-instance:~/folder1$ cat main.tf
provider "aws" {
  region = "us-east-1"
  access_key = "AKIAQ3EGVQL5BV34MMZZ"
  secret_key = "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"
}

# Create VPC
resource "aws_vpc" "my_vpc" {
  cidr_block = "10.0.0.0/16"
  enable_dns_support = true
  enable_dns_hostnames = true
}

# Create Subnet
resource "aws_subnet" "my_subnet" {
  vpc_id                  = aws_vpc.my_vpc.id
  cidr_block              = "10.0.1.0/24"
  map_public_ip_on_launch = true
  availability_zone       = "us-east-1b" # Set your desired availability zone
}

# Create Internet Gateway
resource "aws_internet_gateway" "my_igw" {
  vpc_id = aws_vpc.my_vpc.id
```



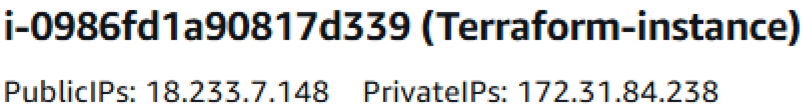
i-0986fd1a90817d339 (Terraform-instance)

PublicIPs: 18.233.7.148 PrivateIPs: 172.31.84.238

Code in script.sh



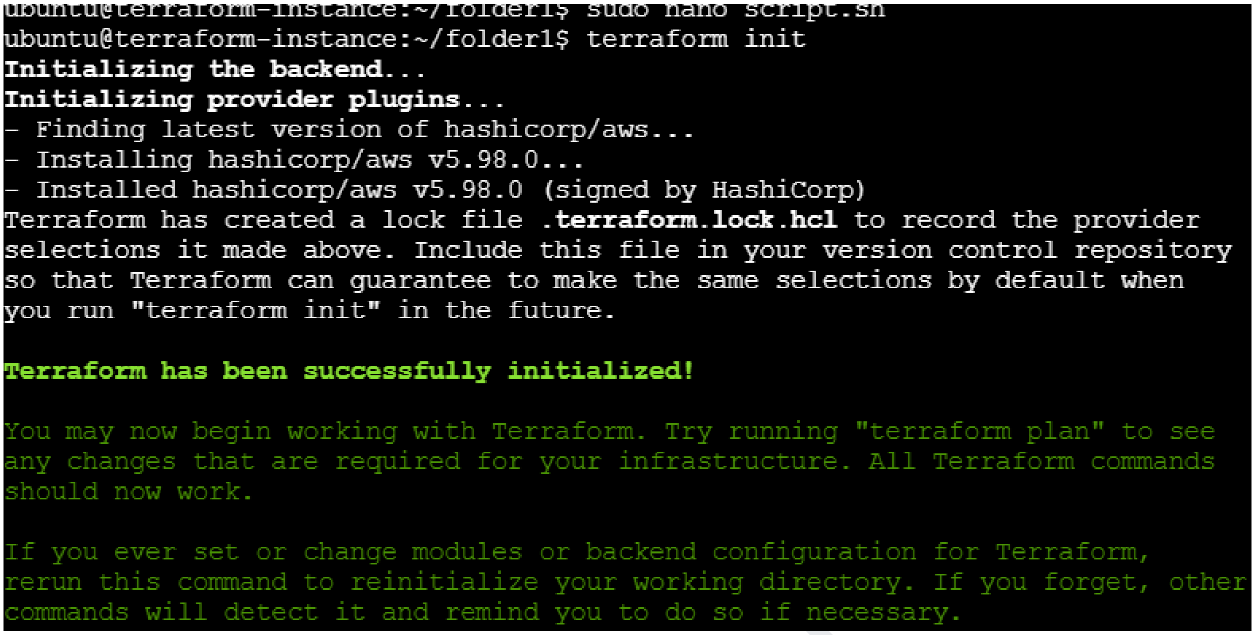
```
ubuntu@terraform-instance:~/folder1$ cat script.sh
#!/bin/bash

sudo apt-get update -y
sudo apt-get install apache2 -y
sudo systemctl start apache2
sudo systemctl enable apache2
ubuntu@terraform-instance:~/folder1$
```



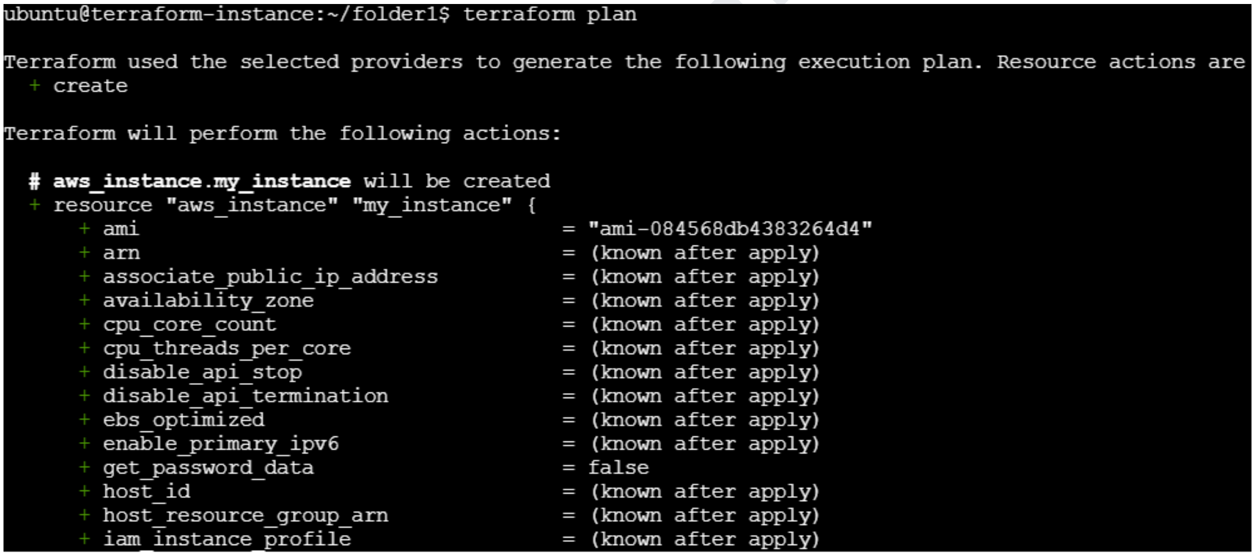
i-0986fd1a90817d339 (Terraform-instance)

PublicIPs: 18.233.7.148 PrivateIPs: 172.31.84.238

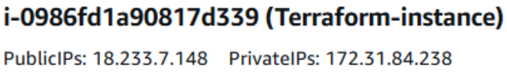
Initiated terraform, then terraform plan and apply gives following result



```
ubuntu@terraform-instance:~/folder1$ sudo nano script.sh
ubuntu@terraform-instance:~/folder1$ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.98.0...
- Installed hashicorp/aws v5.98.0 (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.
```

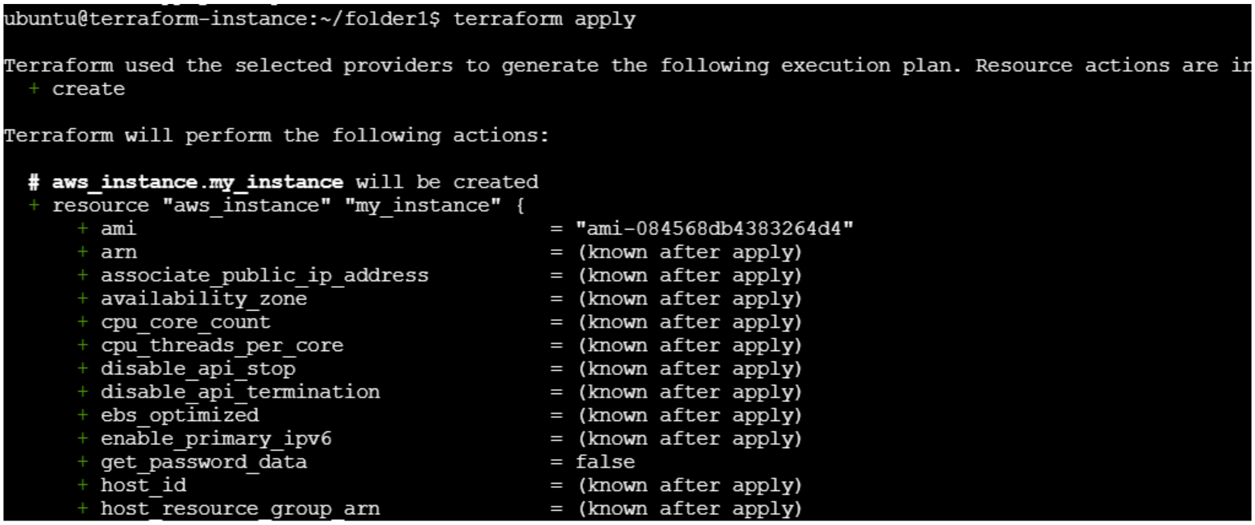


```
ubuntu@terraform-instance:~/folder1$ terraform plan

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

Terraform will perform the following actions:

  # aws_instance.my_instance will be created
  + resource "aws_instance" "my_instance" {
      + ami                                  = "ami-084568db4383264d4"
      + 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)
      + enable_primary_ipv6                  = (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)
```

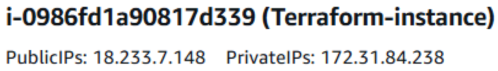


i-0986fd1a90817d339 (Terraform-instance)

PublicIPs: 18.233.7.148   PrivateIPs: 172.31.84.238

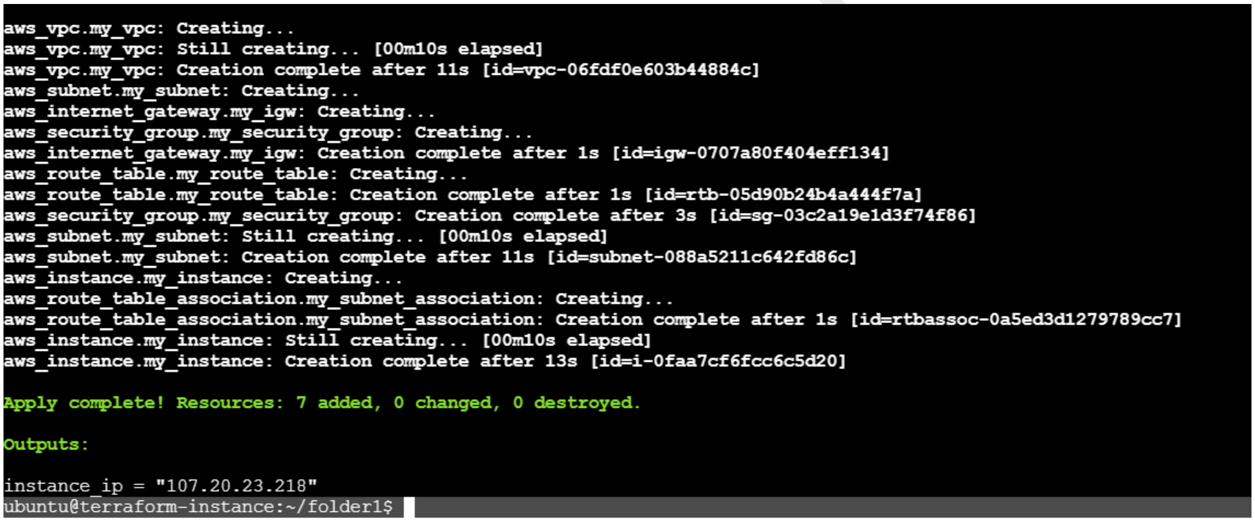


```
ubuntu@terraform-instance:~/folder1$ terraform apply

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

Terraform will perform the following actions:

  # aws_instance.my_instance will be created
  + resource "aws_instance" "my_instance" {
      + ami                                  = "ami-084568db4383264d4"
      + 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)
      + enable_primary_ipv6                  = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
```

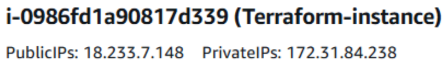


i-0986fd1a90817d339 (Terraform-instance)

PublicIPs: 18.233.7.148   PrivateIPs: 172.31.84.238



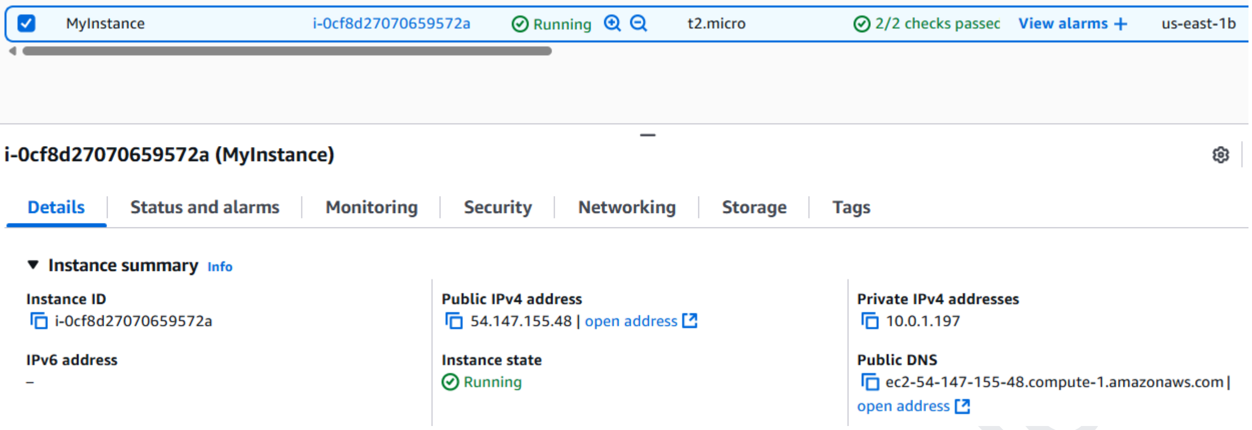
```
aws_vpc.my_vpc: Creating...
aws_vpc.my_vpc: Still creating... [00m10s elapsed]
aws_vpc.my_vpc: Creation complete after 11s [id=vpc-06fdf0e603b44884c]
aws_subnet.my_subnet: Creating...
aws_internet_gateway.my_igw: Creating...
aws_security_group.my_security_group: Creating...
aws_internet_gateway.my_igw: Creation complete after 1s [id=igw-0707a80f404eff134]
aws_route_table.my_route_table: Creating...
aws_route_table.my_route_table: Creation complete after 1s [id=rtb-05d90b24b4a444f7a]
aws_security_group.my_security_group: Creation complete after 3s [id=sg-03c2a19e1d3f74f86]
aws_subnet.my_subnet: Still creating... [00m10s elapsed]
aws_subnet.my_subnet: Creation complete after 11s [id=subnet-088a5211c642fd86c]
aws_instance.my_instance: Creating...
aws_route_table_association.my_subnet_association: Creating...
aws_route_table_association.my_subnet_association: Creation complete after 1s [id=rtbassoc-0a5ed3d1279789cc7]
aws_instance.my_instance: Still creating... [00m10s elapsed]
aws_instance.my_instance: Creation complete after 13s [id=i-0faa7cf6fcc6c5d20]

Apply complete! Resources: 7 added, 0 changed, 0 destroyed.

Outputs:

instance_ip = "107.20.23.218"
ubuntu@terraform-instance:~/folder1$
```



i-0986fd1a90817d339 (Terraform-instance)

PublicIPs: 18.233.7.148   PrivateIPs: 172.31.84.238

New instance launched named MyInstance



☑ MyInstance | i-0cf8d27070659572a | Running | t2.micro | 2/2 checks passec | View alarms + | us-east-1b

i-0cf8d27070659572a (MyInstance)

▼ Instance summary Info

Instance ID: i-0cf8d27070659572a

IPv6 address: –

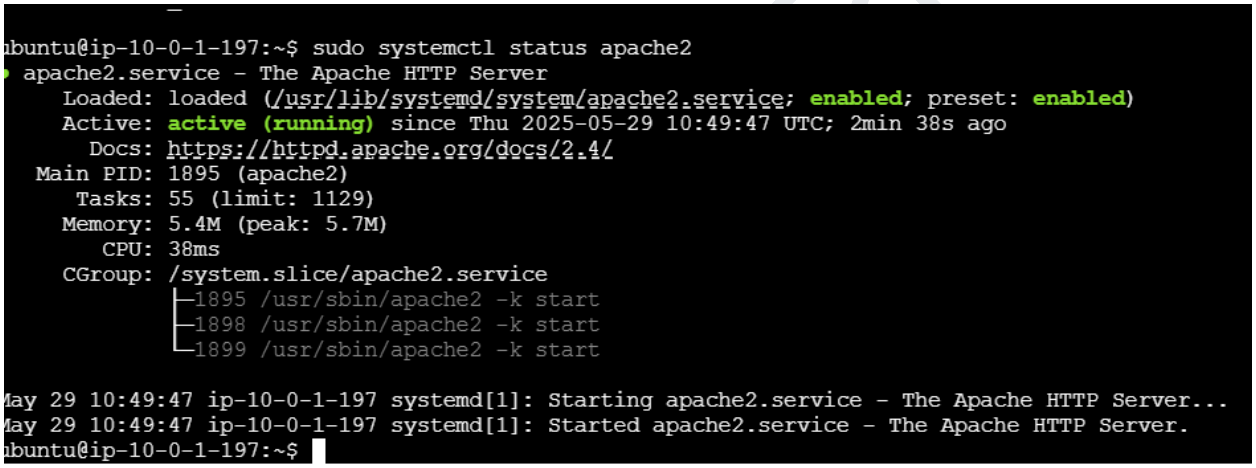
Public IPv4 address: 54.147.155.48 | open address

Instance state: Running

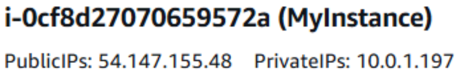
Private IPv4 addresses: 10.0.1.197

Public DNS: ec2-54-147-155-48.compute-1.amazonaws.com | open address

Checking apache2 server at new instance ip

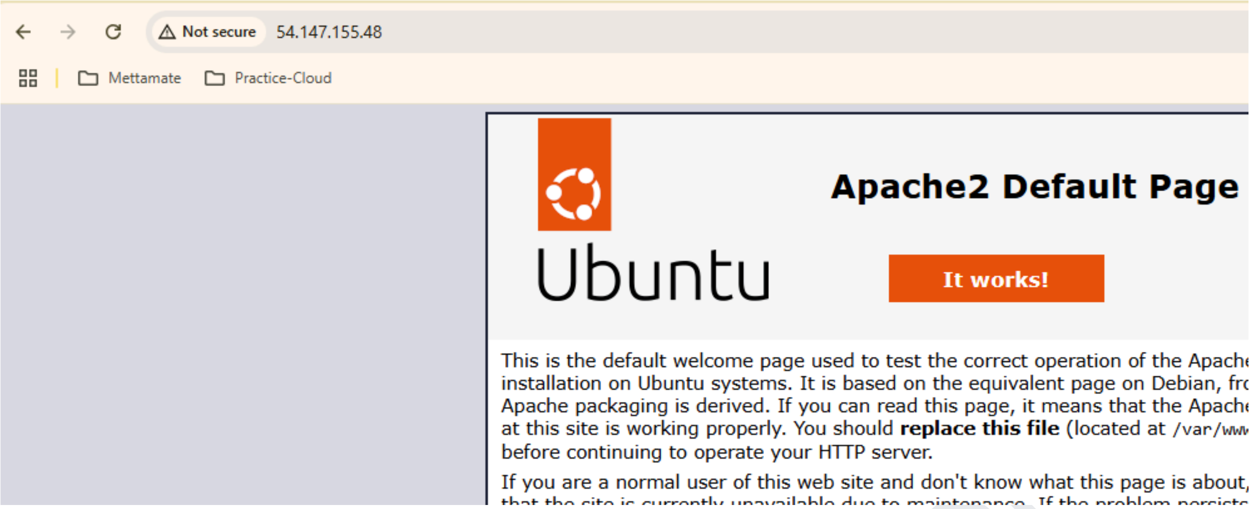


```
ubuntu@ip-10-0-1-197:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Thu 2025-05-29 10:49:47 UTC; 2min 38s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 1895 (apache2)
      Tasks: 55 (limit: 1129)
     Memory: 5.4M (peak: 5.7M)
        CPU: 38ms
     CGroup: /system.slice/apache2.service
             ├─1895 /usr/sbin/apache2 -k start
             ├─1898 /usr/sbin/apache2 -k start
             └─1899 /usr/sbin/apache2 -k start

May 29 10:49:47 ip-10-0-1-197 systemd[1]: Starting apache2.service - The Apache HTTP Server...
May 29 10:49:47 ip-10-0-1-197 systemd[1]: Started apache2.service - The Apache HTTP Server.
ubuntu@ip-10-0-1-197:~$
```



i-0cf8d27070659572a (MyInstance)

PublicIPs: 54.147.155.48   PrivateIPs: 10.0.1.197

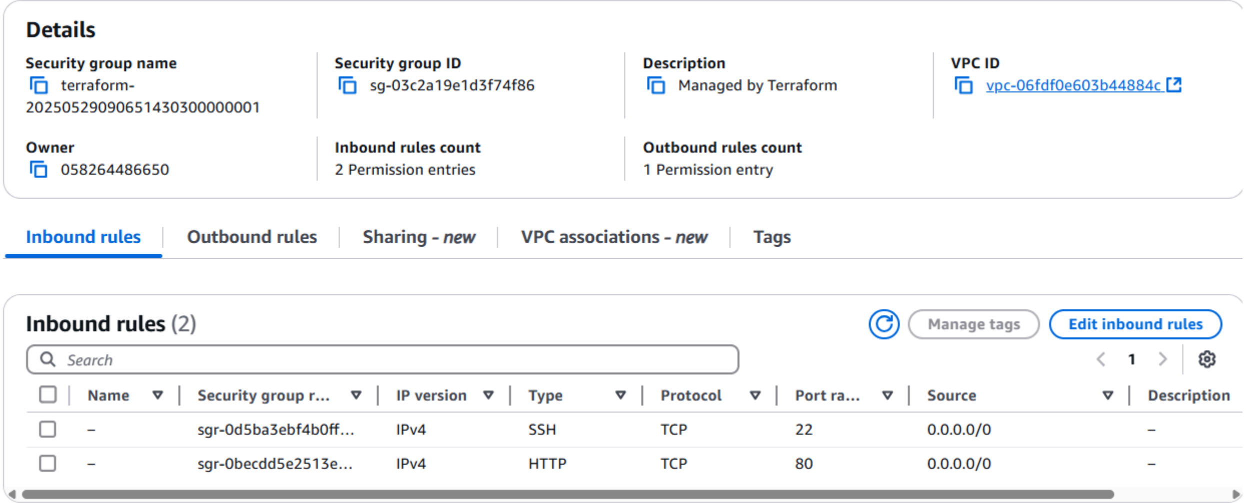


Apache2 Default Page

This is the default welcome page used to test the correct operation of the Apache
installation on Ubuntu systems. It is based on the equivalent page on Debian, fro
Apache packaging is derived. If you can read this page, it means that the Apache
at this site is working properly. You should **replace this file** (located at `/var/www`
before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about,

Security group got created



Details

Security group name
terraform-20250529090651430300000001

Security group ID
sg-03c2a19e1d3f74f86

Description
Managed by Terraform

VPC ID
vpc-06fdf0e603b44884c

Owner
058264486650

Inbound rules count
2 Permission entries

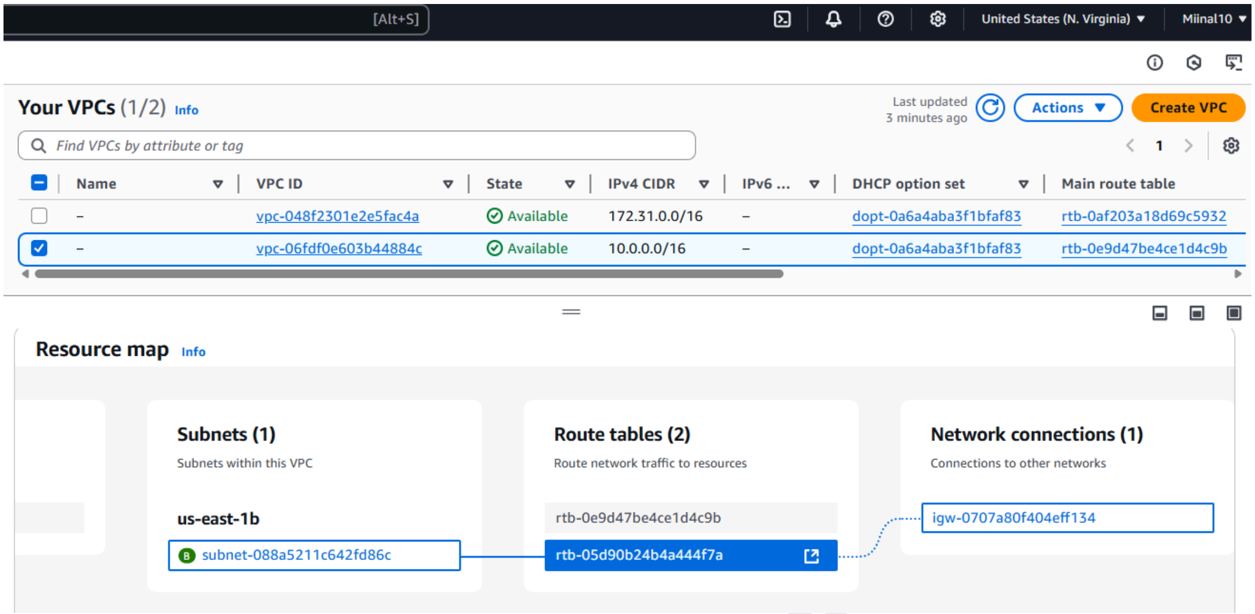
Outbound rules count
1 Permission entry

Inbound rules (2)

□	Name	Security group r...	IP version	Type	Protocol	Port ra...	Source	Description
□	–	sgr-0d5ba3ebf4b0ff...	IPv4	SSH	TCP	22	0.0.0.0/0	–
□	–	sgr-0becdd5e2513e...	IPv4	HTTP	TCP	80	0.0.0.0/0	–

Project name: Infrastructure as a Code using AWS by Ms Minal jadhav

VPC is created with subnets, route table, Internet gateway



Your VPCs (1/2)

☐	Name	VPC ID	State	IPv4 CIDR	IPv6 ...	DHCP option set	Main route table
□	–	vpc-048f2301e2e5fac4a	✓ Available	172.31.0.0/16	–	dopt-0a6a4aba3f1bfaf83	rtb-0af203a18d69c5932
☑	–	vpc-06fdf0e603b44884c	✓ Available	10.0.0.0/16	–	dopt-0a6a4aba3f1bfaf83	rtb-0e9d47be4ce1d4c9b

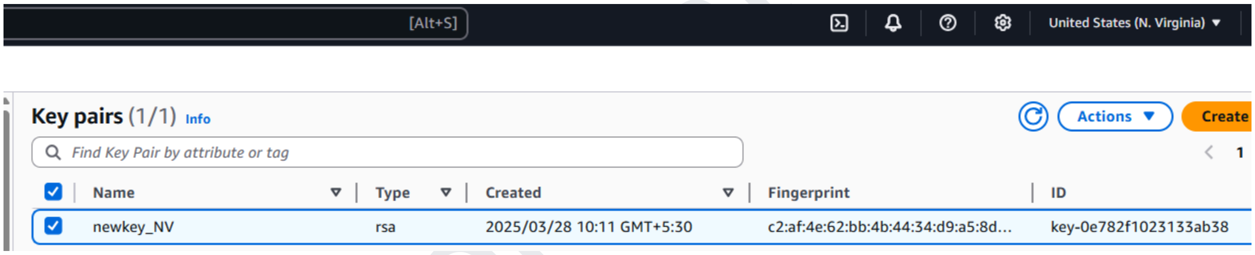
Resource map

Subnets (1)
Subnets within this VPC
us-east-1b
subnet-088a5211c642fd86c

Route tables (2)
Route network traffic to resources
rtb-0e9d47be4ce1d4c9b
rtb-05d90b24b4a444f7a

Network connections (1)
Connections to other networks
igw-0707a80f404eff134

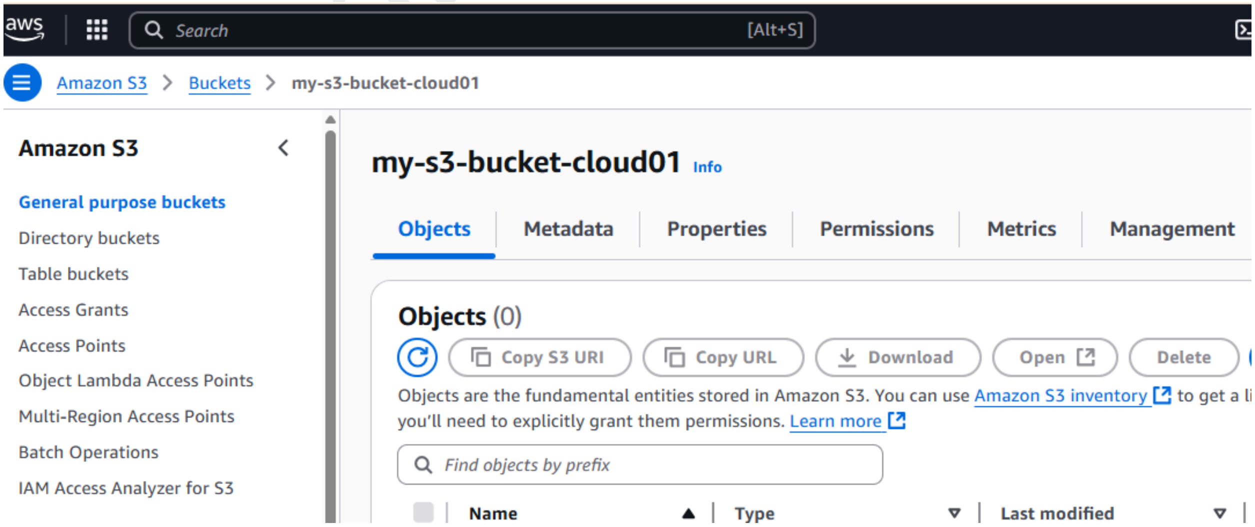
Keypair


[Alt+S]
United States (N. Virginia) ▼
Info
Actions ▼
Create
Find Key Pair by attribute or tag
1

Key pairs (1/1)

☑	Name	Type	Created	Fingerprint	ID
☑	newkey_NV	rsa	2025/03/28 10:11 GMT+5:30	c2:af:4e:62:bb:4b:44:34:d9:a5:8d...	key-0e782f1023133ab38

S3 bucket is created



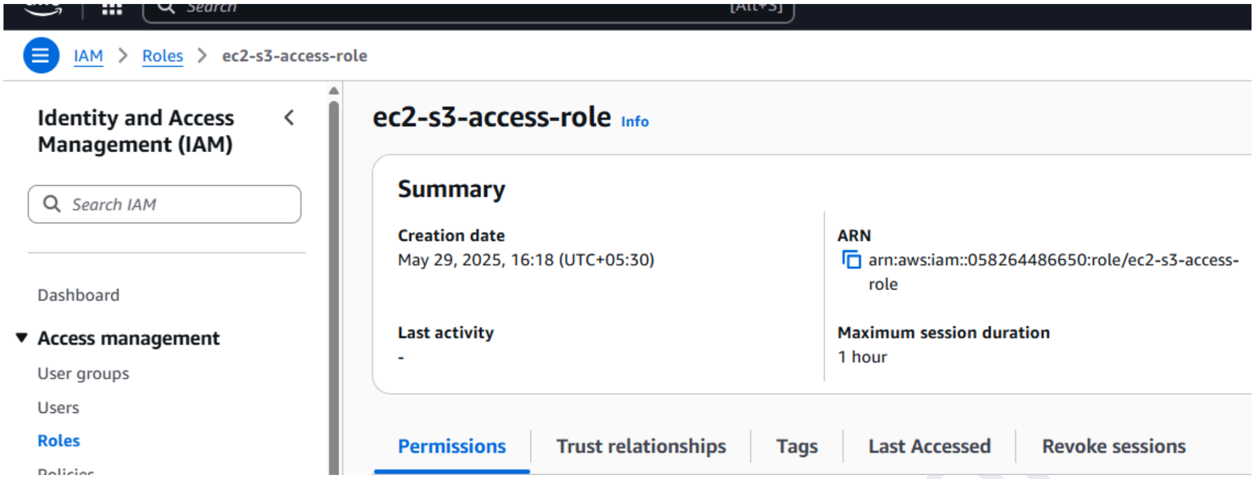
my-s3-bucket-cloud01

Objects (0)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a li
you'll need to explicitly grant them permissions. Learn more

Project name: Infrastructure as a Code using AWS by Ms Minal jadhav

IAM role with policy attached to it



IAM > Roles > ec2-s3-access-role

ec2-s3-access-role Info

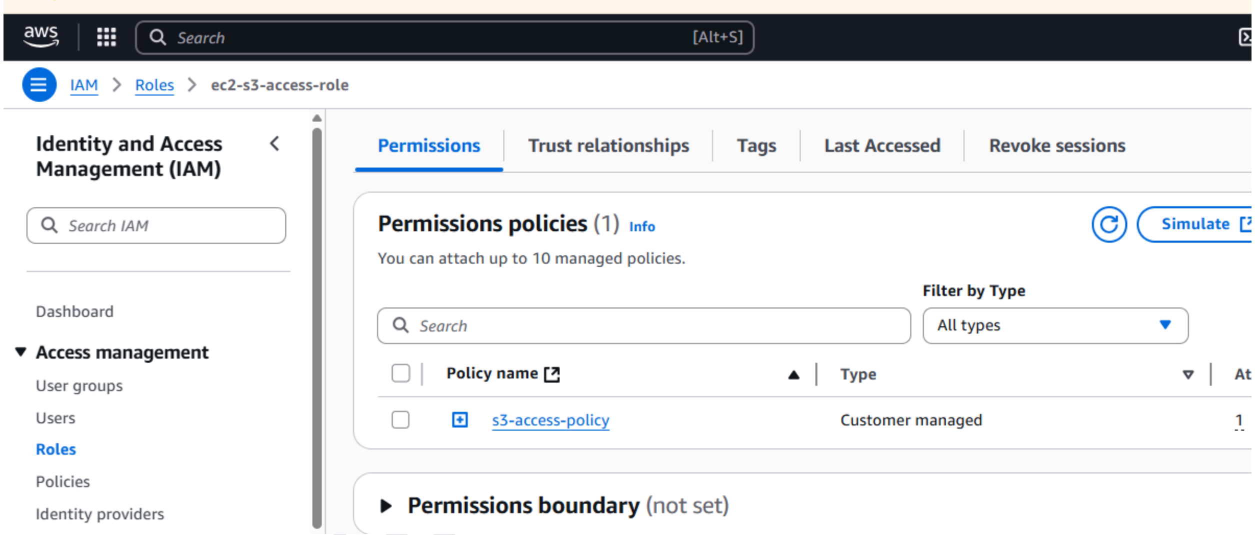
Summary

Creation date
May 29, 2025, 16:18 (UTC+05:30)

ARN
arn:aws:iam::058264486650:role/ec2-s3-access-role

Last activity
-

Maximum session duration
1 hour



IAM > Roles > ec2-s3-access-role

Permissions policies (1) Info

You can attach up to 10 managed policies.

Filter by Type

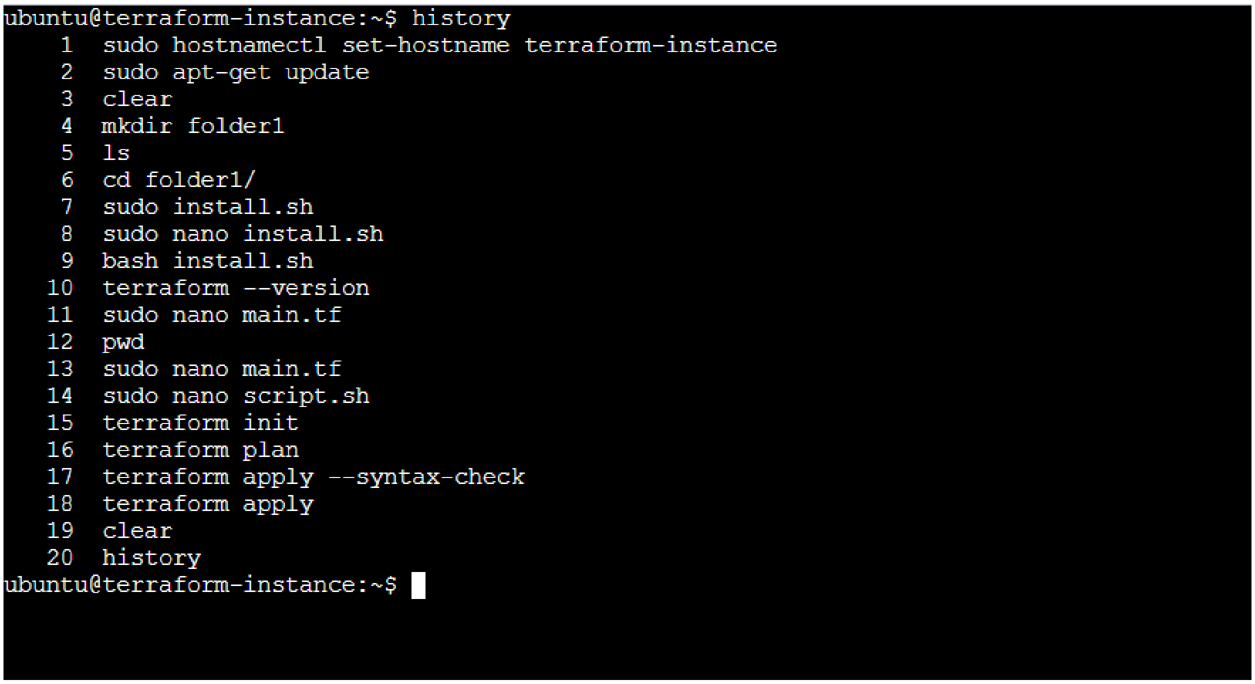
All types

□	Policy name	Type	At
□	s3-access-policy	Customer managed	1

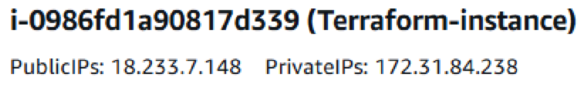
Permissions boundary (not set)

Project name: Infrastructure as a Code using AWS by Ms Minal jadhav

Commands used as follows:



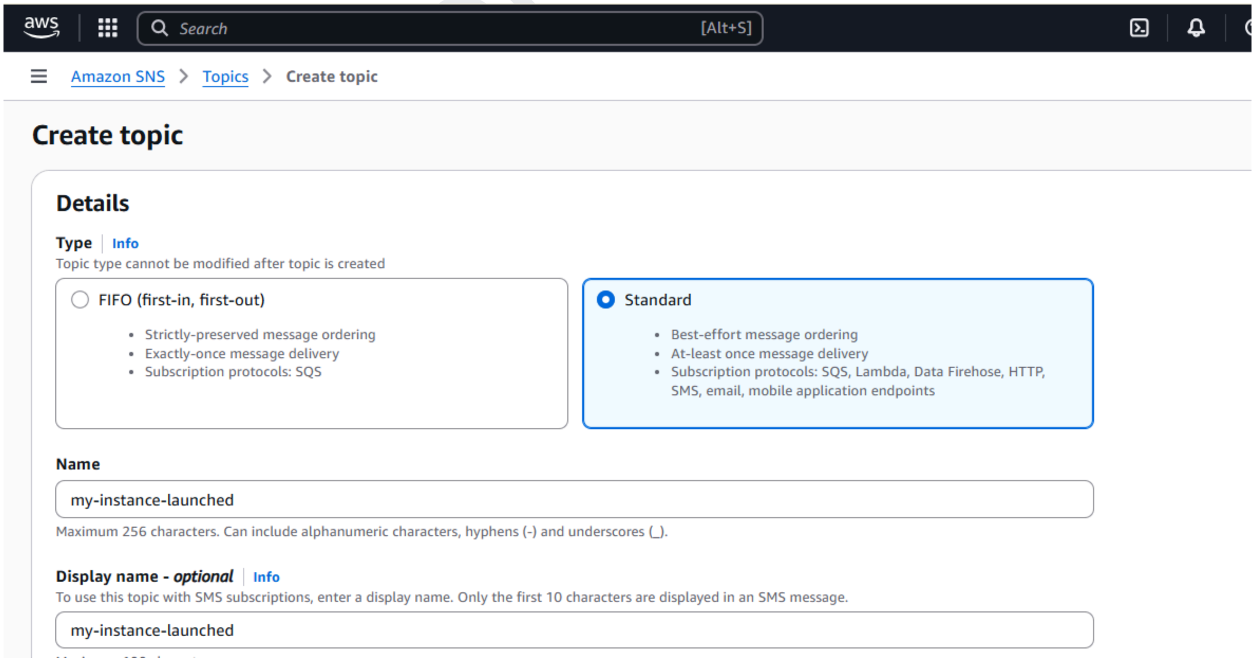
```
ubuntu@terraform-instance:~$ history
    1  sudo hostnamectl set-hostname terraform-instance
    2  sudo apt-get update
    3  clear
    4  mkdir folder1
    5  ls
    6  cd folder1/
    7  sudo install.sh
    8  sudo nano install.sh
    9  bash install.sh
   10  terraform --version
   11  sudo nano main.tf
   12  pwd
   13  sudo nano main.tf
   14  sudo nano script.sh
   15  terraform init
   16  terraform plan
   17  terraform apply --syntax-check
   18  terraform apply
   19  clear
   20  history
ubuntu@terraform-instance:~$ 
```



i-0986fd1a90817d339 (Terraform-instance)

PublicIPs: 18.233.7.148   PrivateIPs: 172.31.84.238

Creating SNS notification



Create topic

Details

Type | Info

Topic type cannot be modified after topic is created

○ FIFO (first-in, first-out)

- Strictly-preserved message ordering
- Exactly-once message delivery
- Subscription protocols: SQS

● Standard

- Best-effort message ordering
- At-least once message delivery
- Subscription protocols: SQS, Lambda, Data Firehose, HTTP, SMS, email, mobile application endpoints

Name

my-instance-launched

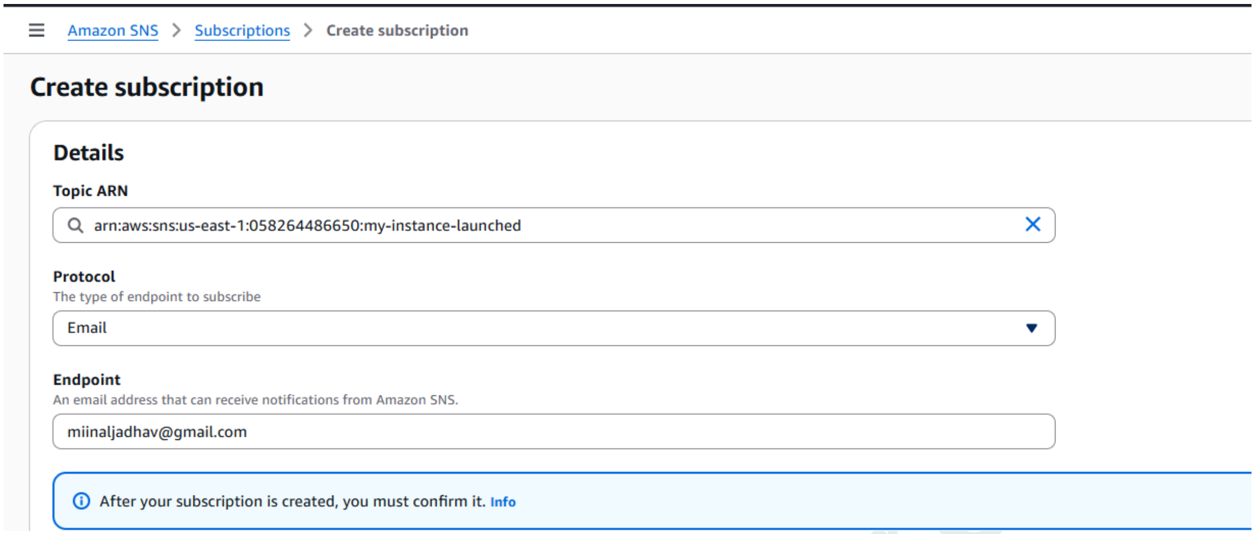
Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).

Display name - *optional* | Info

To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message.

my-instance-launched

Creating subscription to get notification on email



Create subscription

Details

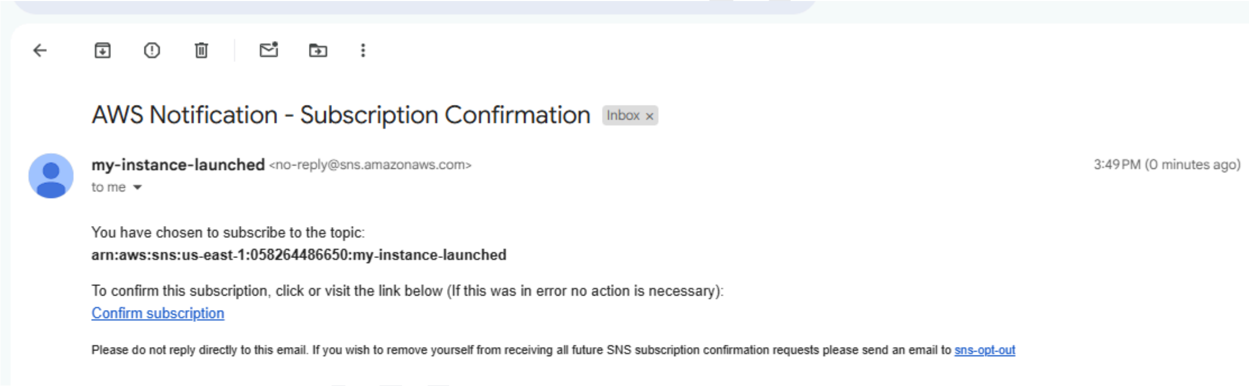
Topic ARN: arn:aws:sns:us-east-1:058264486650:my-instance-launched

Protocol
The type of endpoint to subscribe
Email

Endpoint
An email address that can receive notifications from Amazon SNS.
miinaljadhav@gmail.com

After your subscription is created, you must confirm it. Info

Received an email



AWS Notification - Subscription Confirmation

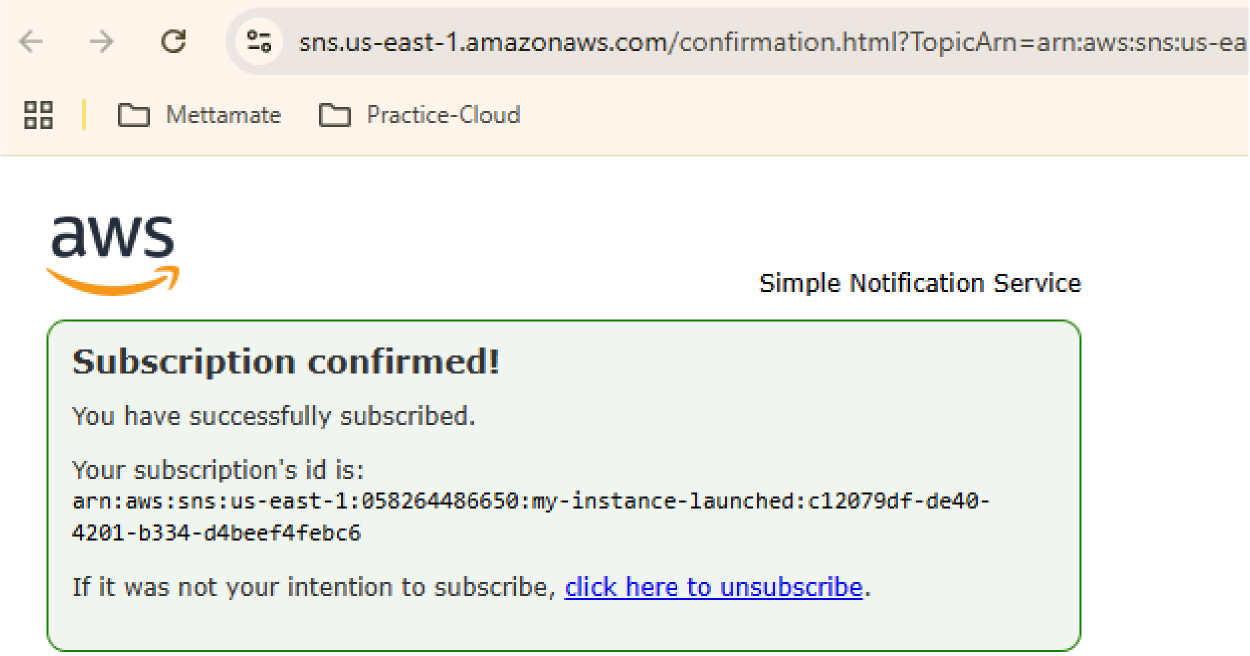
my-instance-launched <no-reply@sns.amazonaws.com>
to me

3:49 PM (0 minutes ago)

You have chosen to subscribe to the topic:
arn:aws:sns:us-east-1:058264486650:my-instance-launched

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
Confirm subscription

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to sns-opt-out

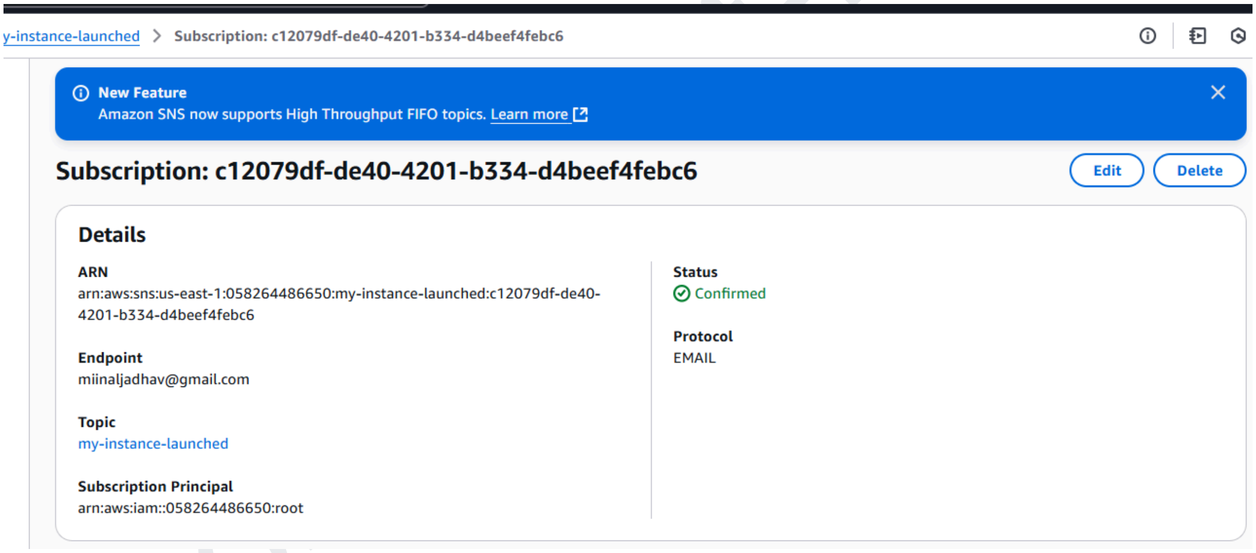

sns.us-east-1.amazonaws.com/confirmation.html?TopicArn=arn:aws:sns:us-ea
Mettamate
Practice-Cloud
aws
Simple Notification Service

Subscription confirmed!

You have successfully subscribed.

Your subscription's id is:
arn:aws:sns:us-east-1:058264486650:my-instance-launched:c12079df-de40-4201-b334-d4beef4febc6

If it was not your intention to subscribe, click here to unsubscribe.



New Feature
Amazon SNS now supports High Throughput FIFO topics. Learn more

Subscription: c12079df-de40-4201-b334-d4beef4febc6

Details

ARN
arn:aws:sns:us-east-1:058264486650:my-instance-launched:c12079df-de40-4201-b334-d4beef4febc6

Endpoint
miinaljadhav@gmail.com

Topic
my-instance-launched

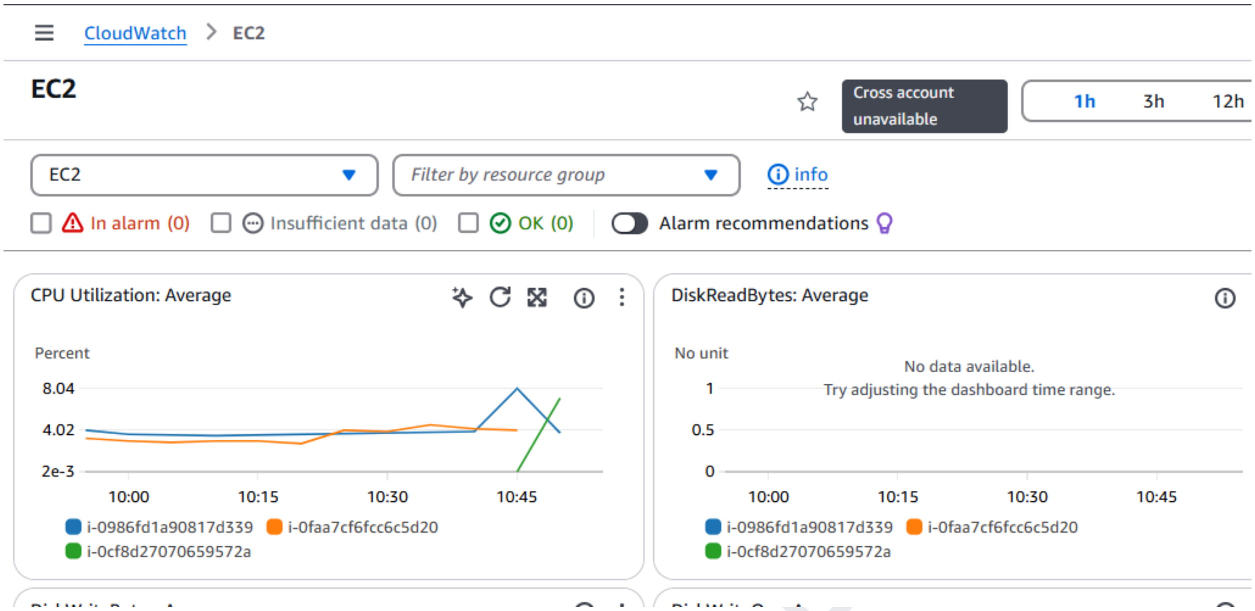
Subscription Principal
arn:aws:iam::058264486650:root

Status
Confirmed

Protocol
EMAIL

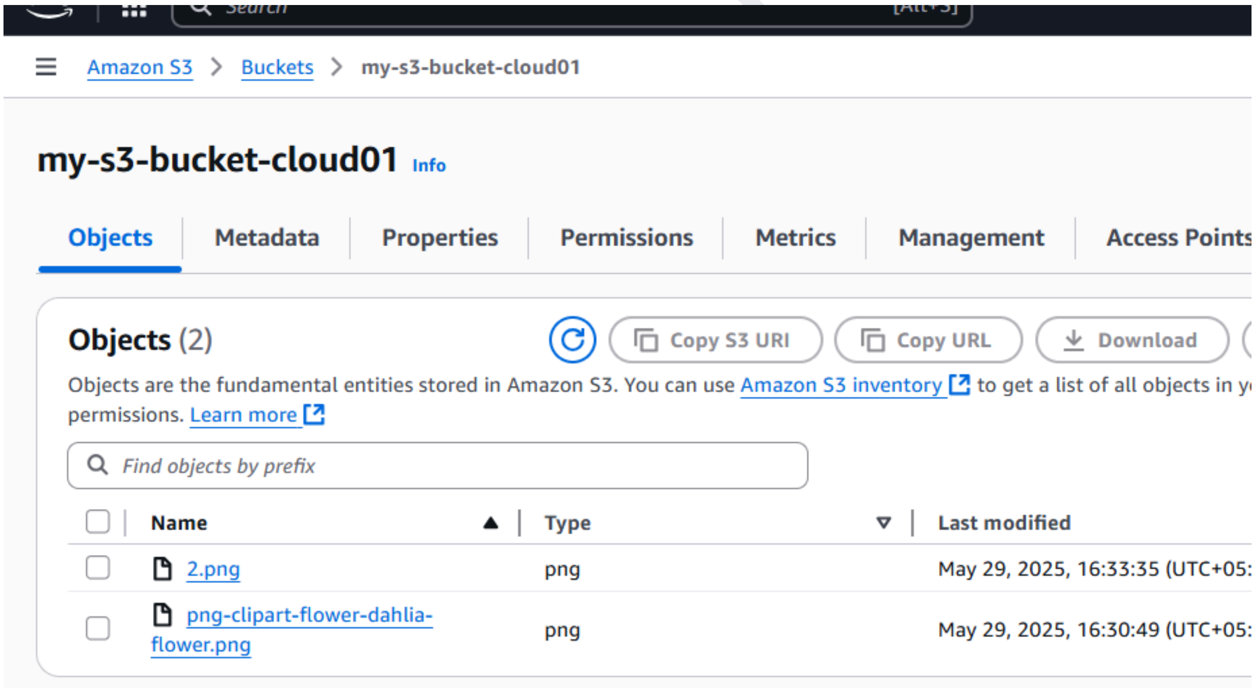
Project name: Infrastructure as a Code using AWS by Ms Minal jadhav

Cloudwatch dashboard



No data available.
Try adjusting the dashboard time range.

Uploading objects in s3 bucket



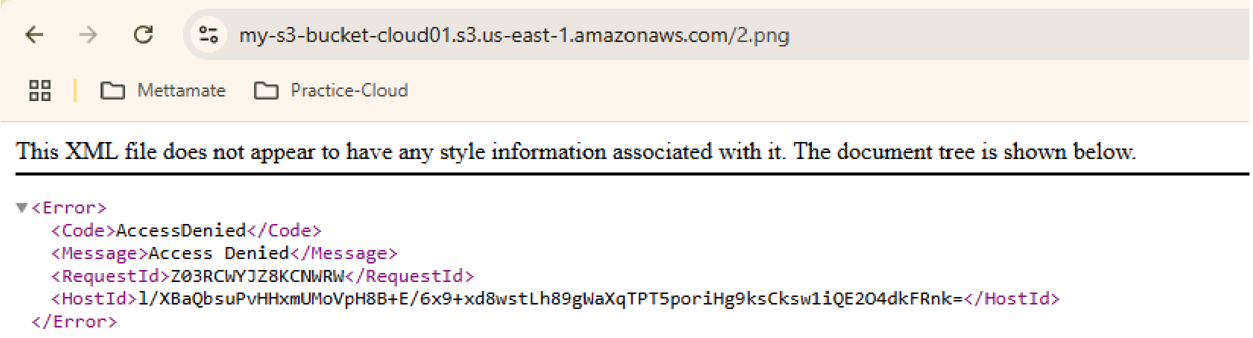
my-s3-bucket-cloud01

Objects (2)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in y
permissions. Learn more

□	Name	Type	Last modified
□	2.png	png	May 29, 2025, 16:33:35 (UTC+05:
□	png-clipart-flower-dahlia-flower.png	png	May 29, 2025, 16:30:49 (UTC+05:

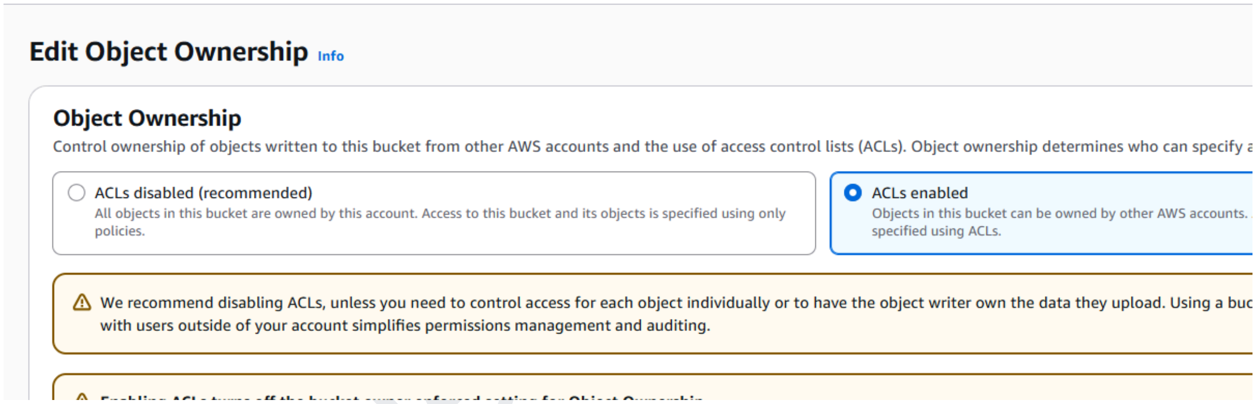
Objects are not public hence not visible through URL



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Error>
   <Code>AccessDenied</Code>
   <Message>Access Denied</Message>
   <RequestId>Z03RCWYJZ8KCNWRW</RequestId>
   <HostId>l/XBaQbsuPvHHxmUMoVpH8B+E/6x9+xd8wstLh89gWaXqTPT5poriHg9ksCksw1iQE2O4dkFRnk=</HostId>
 </Error>
```

Removing block access and enabling ACLs to make objects public



Edit Object Ownership Info

Object Ownership

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify a

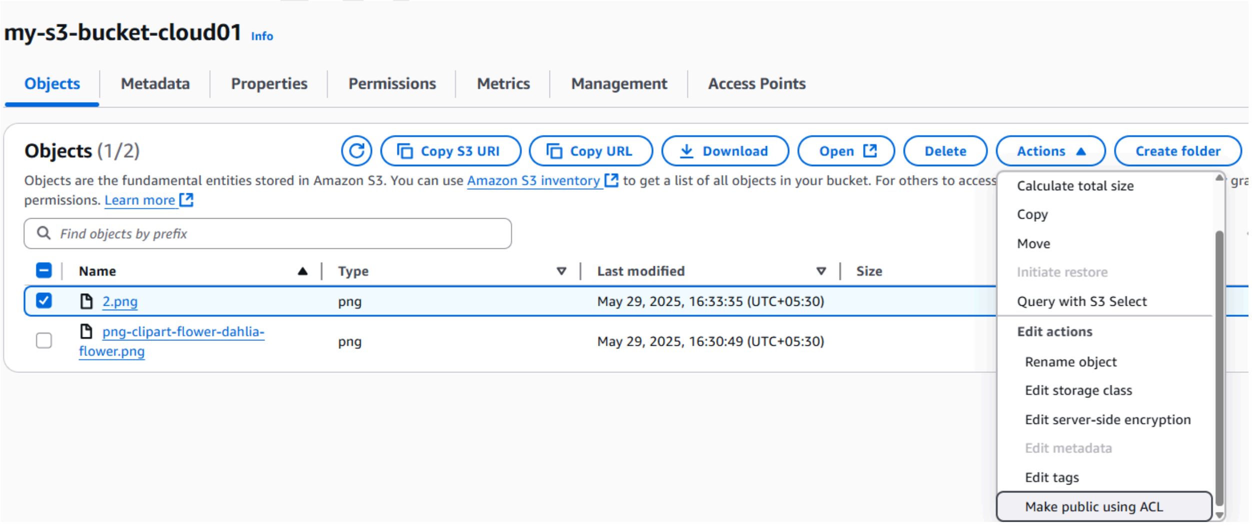
○ ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

● ACLs enabled
Objects in this bucket can be owned by other AWS accounts. specified using ACLs.

⚠ We recommend disabling ACLs, unless you need to control access for each object individually or to have the object writer own the data they upload. Using a buc with users outside of your account simplifies permissions management and auditing.

⚠ Enabling ACLs turns off the bucket owner enforced setting for Object Ownership

Now objects are public

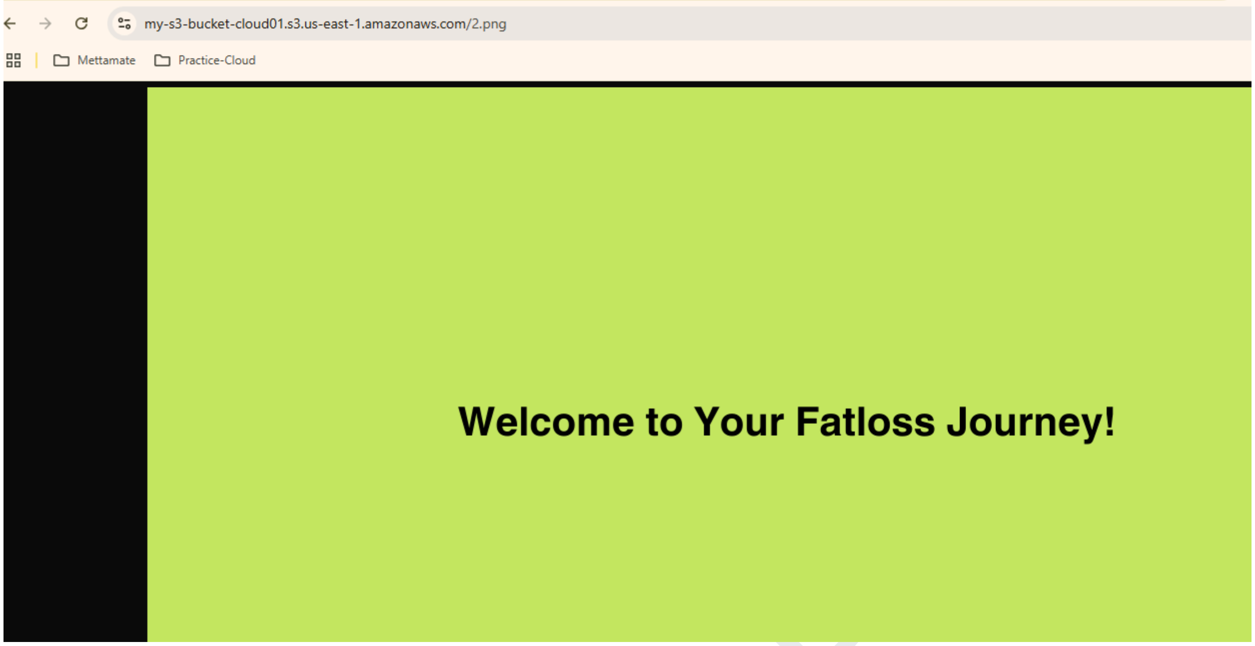


Objects (1/2)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access gra permissions. Learn more

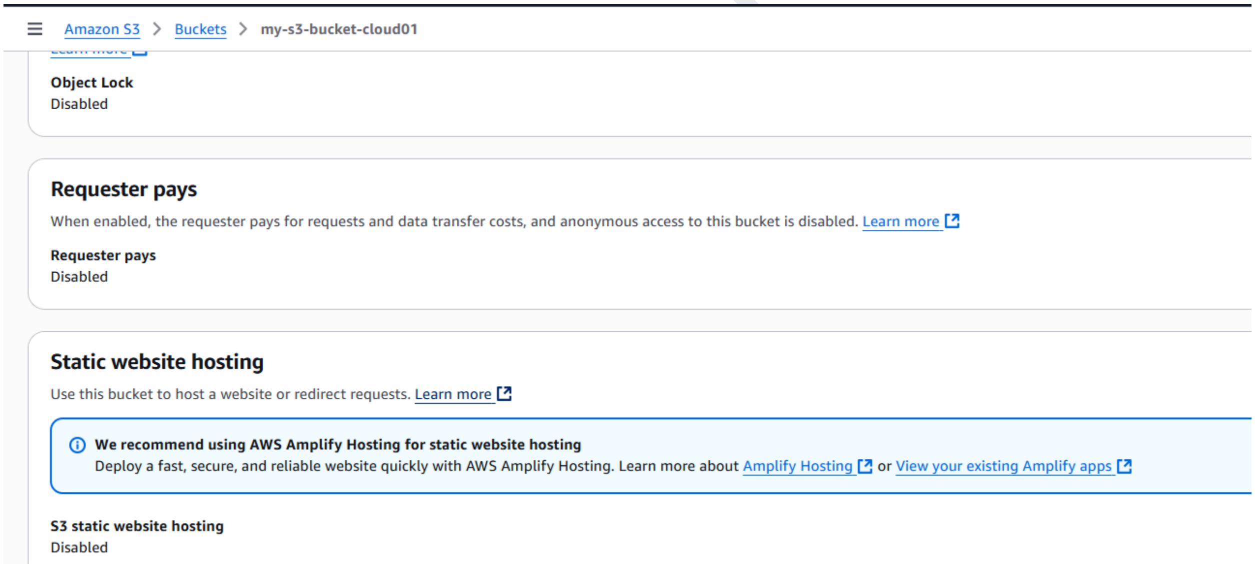
Find objects by prefix

	Name	Type	Last modified	Size
☑	2.png	png	May 29, 2025, 16:33:35 (UTC+05:30)	
□	png-clipart-flower-dahlia-flower.png	png	May 29, 2025, 16:30:49 (UTC+05:30)	


my-s3-bucket-cloud01.s3.us-east-1.amazonaws.com/2.png

Welcome to Your Fatloss Journey!

Creating a static website



Object Lock
Disabled

Requester pays

When enabled, the requester pays for requests and data transfer costs, and anonymous access to this bucket is disabled. Learn more

Requester pays
Disabled

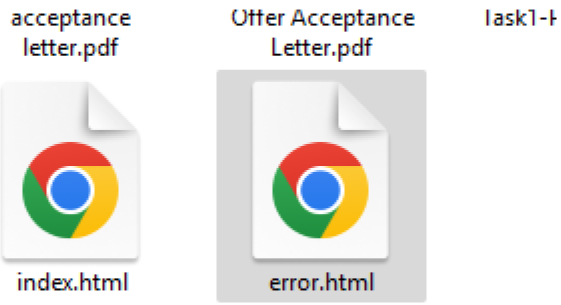
Static website hosting

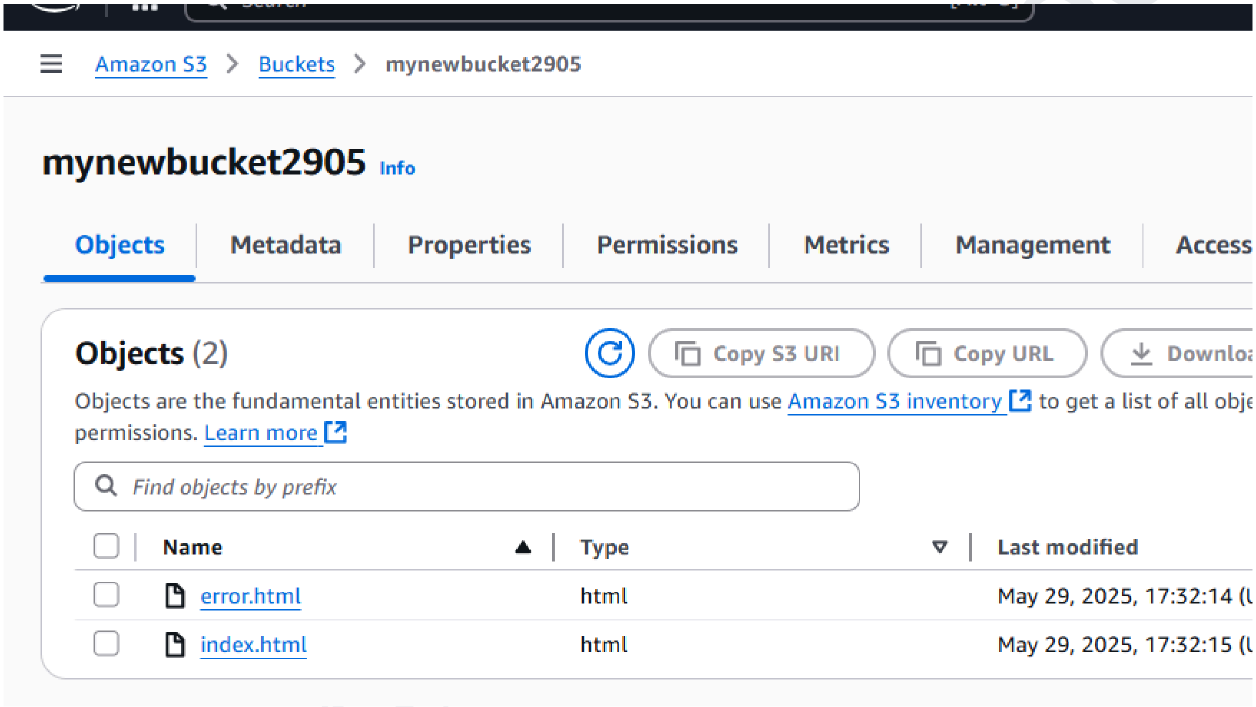
Use this bucket to host a website or redirect requests. Learn more

We recommend using AWS Amplify Hosting for static website hosting
Deploy a fast, secure, and reliable website quickly with AWS Amplify Hosting. Learn more about Amplify Hosting or View your existing Amplify apps

S3 static website hosting
Disabled

Created files for static website hosting and uploaded in bucket





mynewbucket2905 Info

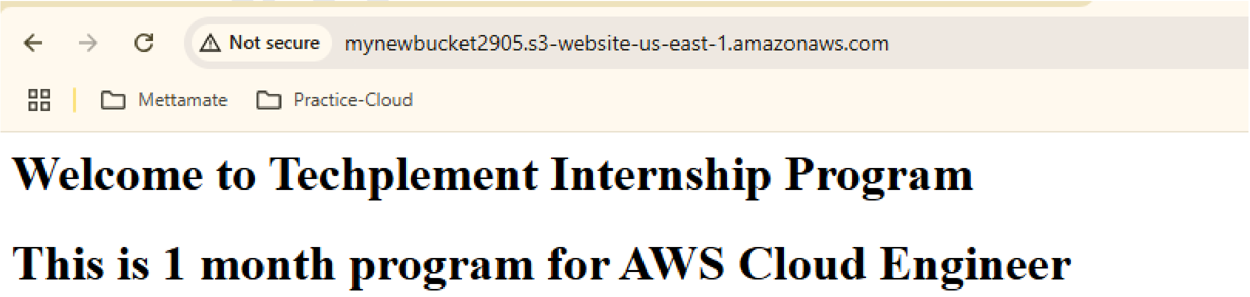
Objects (2)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all obje permissions. Learn more

Find objects by prefix

□	Name	Type	Last modified
□	error.html	html	May 29, 2025, 17:32:14 (U
□	index.html	html	May 29, 2025, 17:32:15 (U

Index.html page

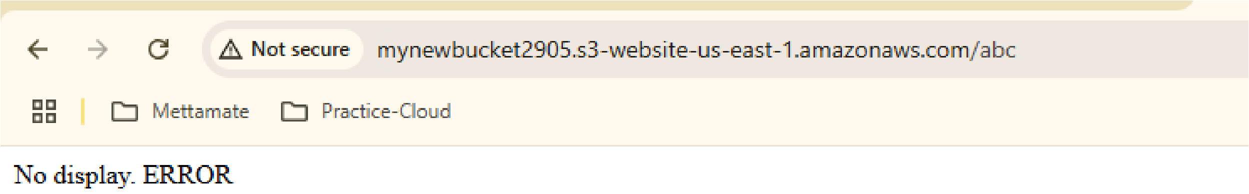


Welcome to Techplement Internship Program

This is 1 month program for AWS Cloud Engineer

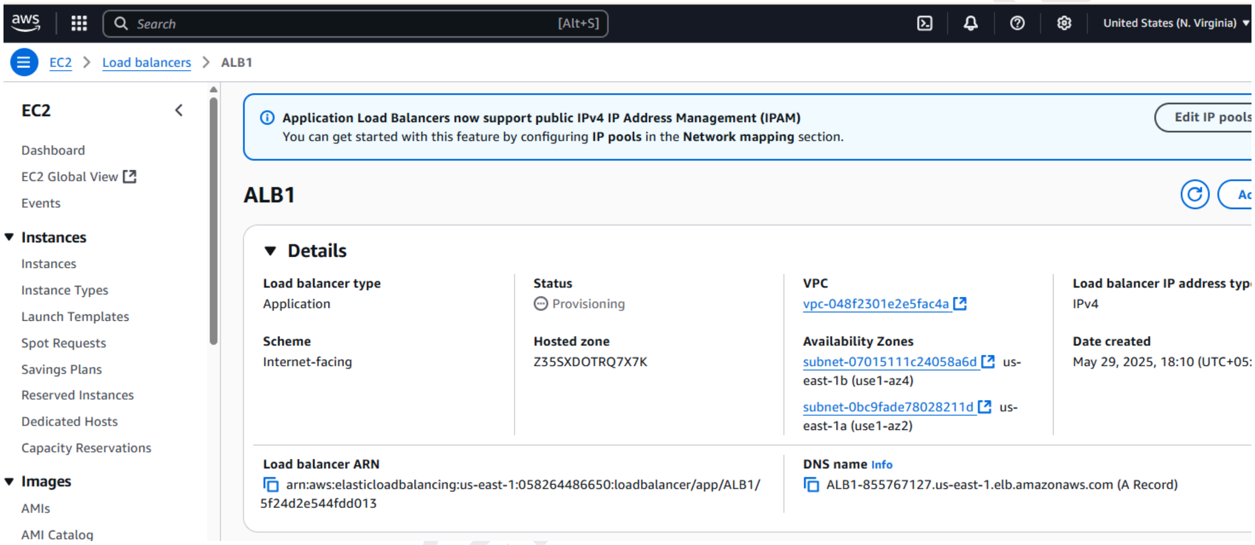
Project name: Infrastructure as a Code using AWS by Ms Minal jadhav

Error page



No display. ERROR

Creating Application load balancer



Application Load Balancers now support public IPv4 IP Address Management (IPAM)
You can get started with this feature by configuring **IP pools** in the **Network mapping** section.

ALB1

▼ Details

Load balancer type
Application

Status
Provisioning

VPC
vpc-048f2301e2e5fac4a

Load balancer IP address typ
IPv4

Scheme
Internet-facing

Hosted zone
Z35SXDOTRQ7X7K

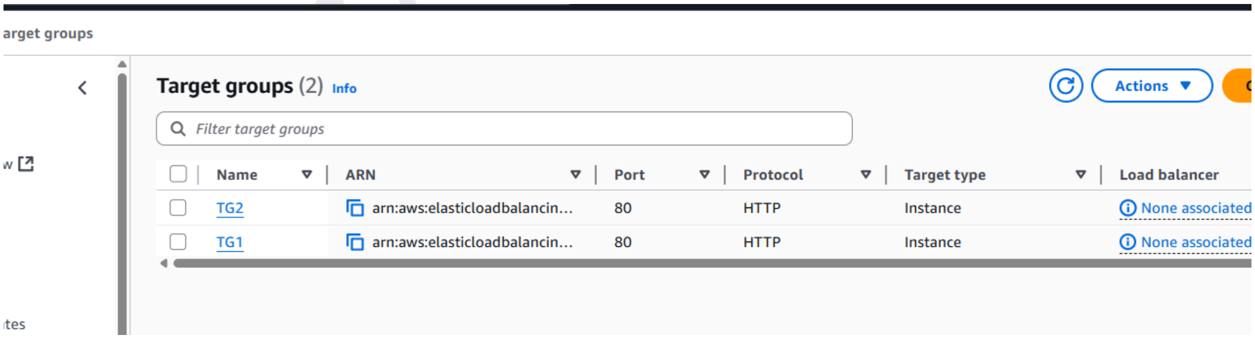
Availability Zones
subnet-07015111c24058a6d us-east-1b (use1-az4)
subnet-0bc9fade78028211d us-east-1a (use1-az2)

Date created
May 29, 2025, 18:10 (UTC+05:

Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:058264486650:loadbalancer/app/ALB1/5f24d2e544fdd013

DNS name Info
ALB1-855767127.us-east-1.elb.amazonaws.com (A Record)

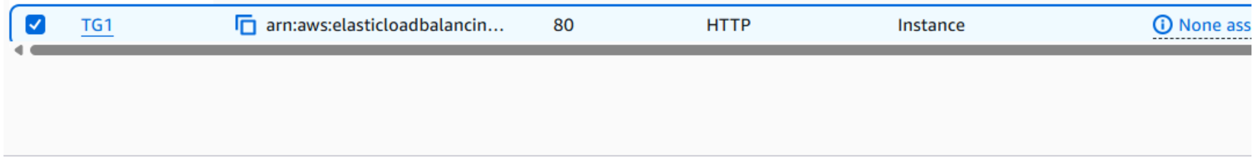
Creating Target group TG1 & TG2



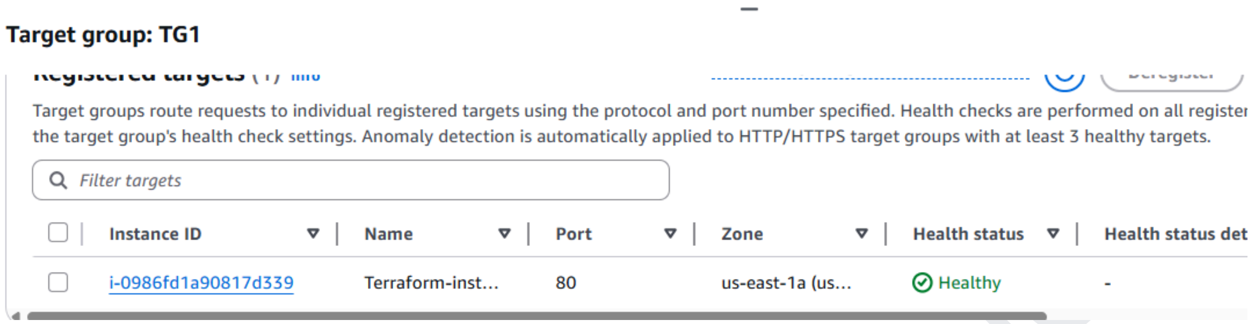
Target groups (2) Info

□	Name	ARN	Port	Protocol	Target type	Load balancer
□	TG2	arn:aws:elasticloadbalancin…	80	HTTP	Instance	None associated
□	TG1	arn:aws:elasticloadbalancin…	80	HTTP	Instance	None associated

Project name: Infrastructure as a Code using AWS by Ms Minal jadhav



☑	TG1	arn:aws:elasticloadbalancin...	80	HTTP	Instance	ⓘ None ass



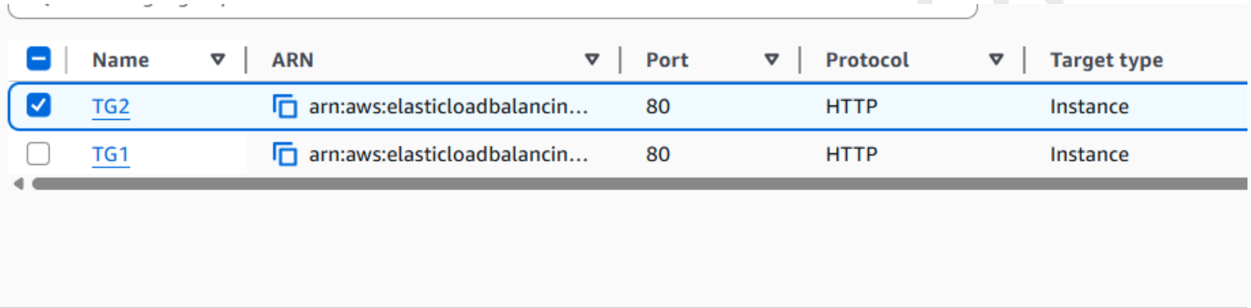
Target group: TG1

Registered targets (1) Info

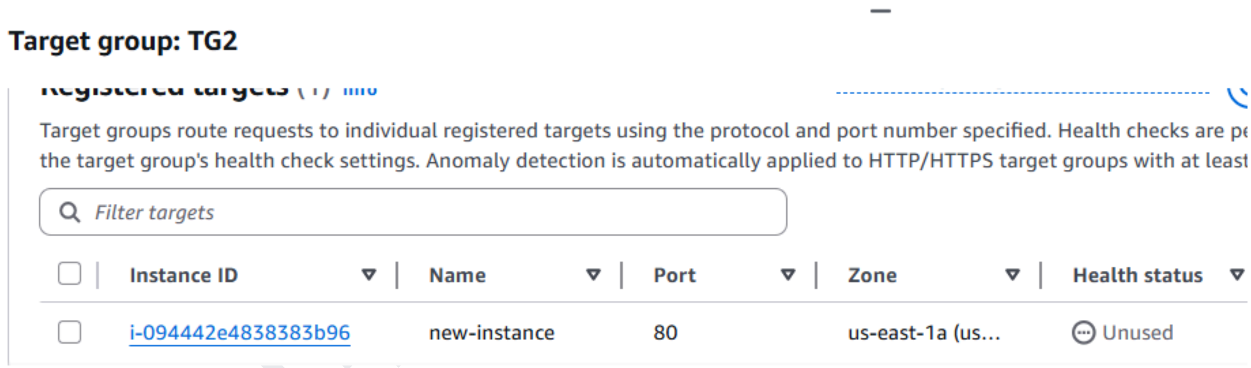
Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all register
the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

Filter targets

□	Instance ID	Name	Port	Zone	Health status	Health status det
□	i-0986fd1a90817d339	Terraform-inst...	80	us-east-1a (us...	✓ Healthy	-



☐	Name	ARN	Port	Protocol	Target type
☑	TG2	arn:aws:elasticloadbalancin...	80	HTTP	Instance
□	TG1	arn:aws:elasticloadbalancin...	80	HTTP	Instance



Target group: TG2

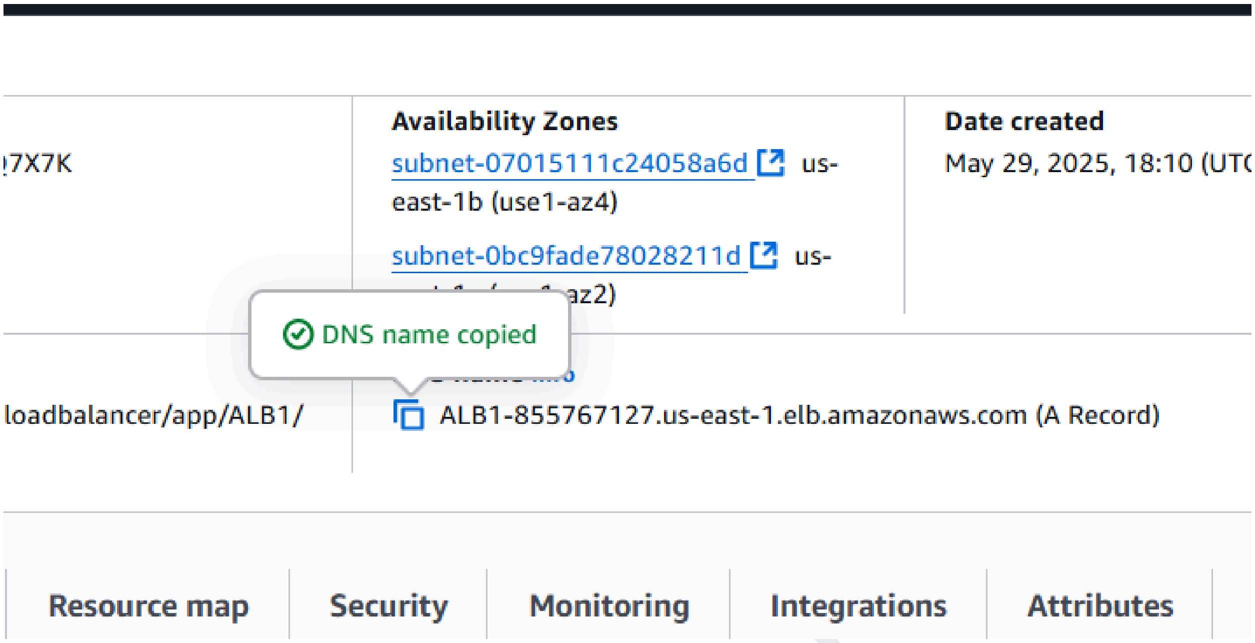
Registered targets (1) Info

Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are pe
the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least

Filter targets

□	Instance ID	Name	Port	Zone	Health status
□	i-094442e4838383b96	new-instance	80	us-east-1a (us...	⊖ Unused

Copied DNS of ALB



7X7K

Availability Zones
subnet-07015111c24058a6d us-east-1b (use1-az4)
subnet-0bc9fade78028211d us-

az2)

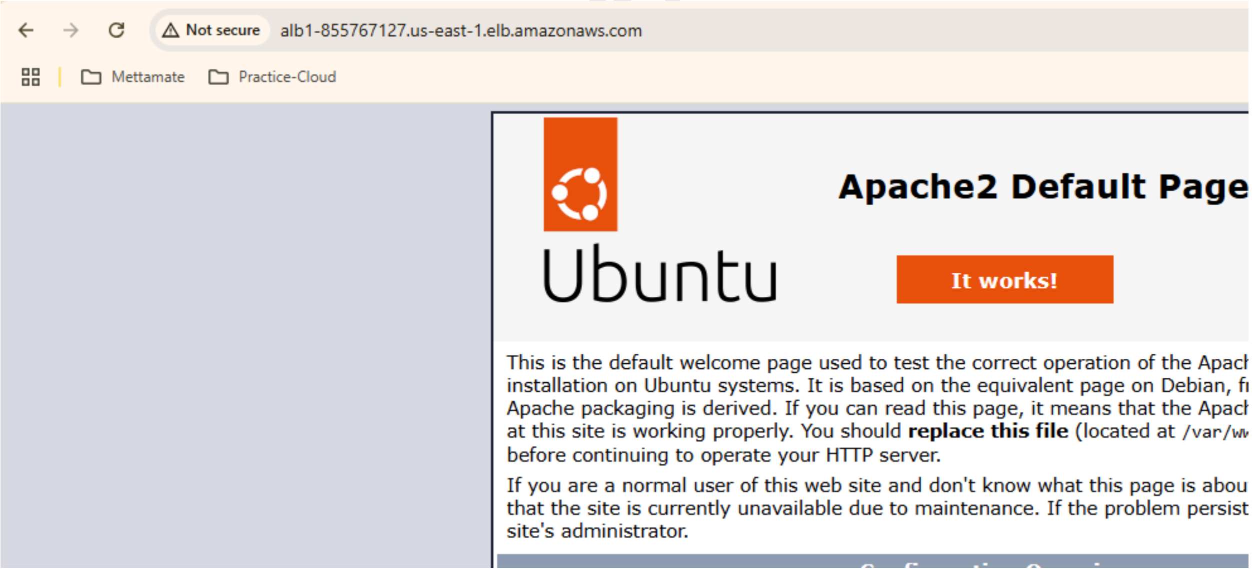
Date created
May 29, 2025, 18:10 (UTC

DNS name copied

loadbalancer/app/ALB1/

ALB1-855767127.us-east-1.elb.amazonaws.com (A Record)

Server1: 70% traffic



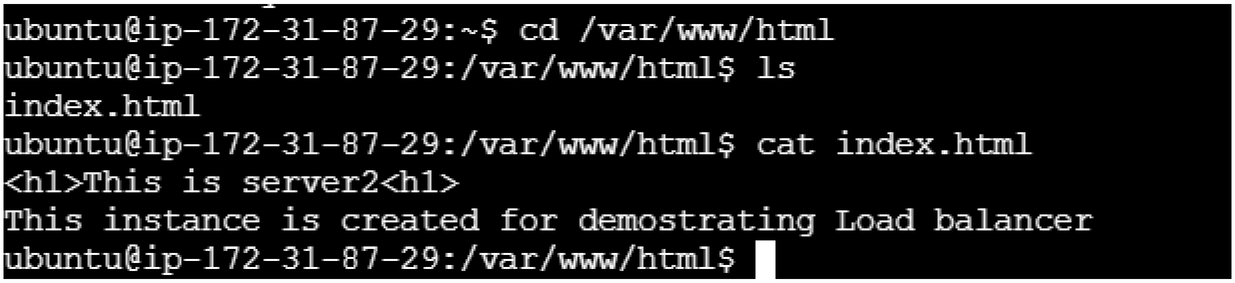
Not secure alb1-855767127.us-east-1.elb.amazonaws.com

Apache2 Default Page

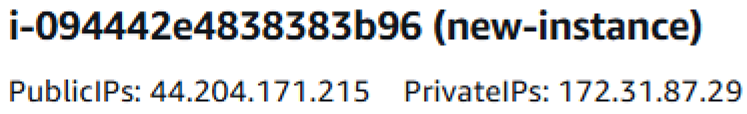
This is the default welcome page used to test the correct operation of the Apach installation on Ubuntu systems. It is based on the equivalent page on Debian, fr Apache packaging is derived. If you can read this page, it means that the Apach at this site is working properly. You should **replace this file** (located at /var/ww before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is abou that the site is currently unavailable due to maintenance. If the problem persist site's administrator.

Content in index.html of server2



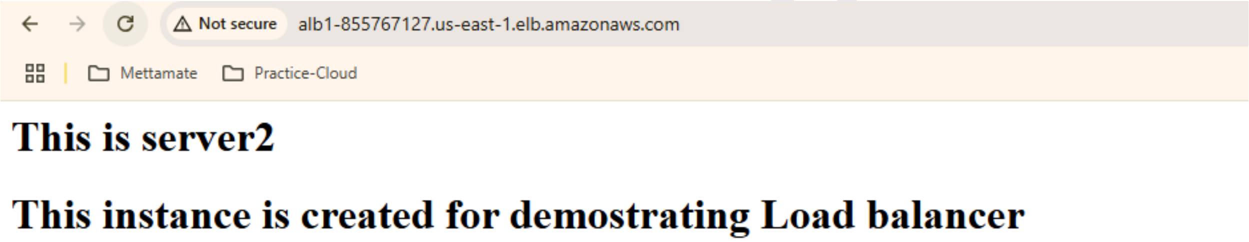
```
ubuntu@ip-172-31-87-29:~$ cd /var/www/html
ubuntu@ip-172-31-87-29:/var/www/html$ ls
index.html
ubuntu@ip-172-31-87-29:/var/www/html$ cat index.html
<h1>This is server2<h1>
This instance is created for demostrating Load balancer
ubuntu@ip-172-31-87-29:/var/www/html$
```



i-094442e4838383b96 (new-instance)

PublicIPs: 44.204.171.215   PrivateIPs: 172.31.87.29

Server2: 30% traffic



This is server2

This instance is created for demostrating Load balancer