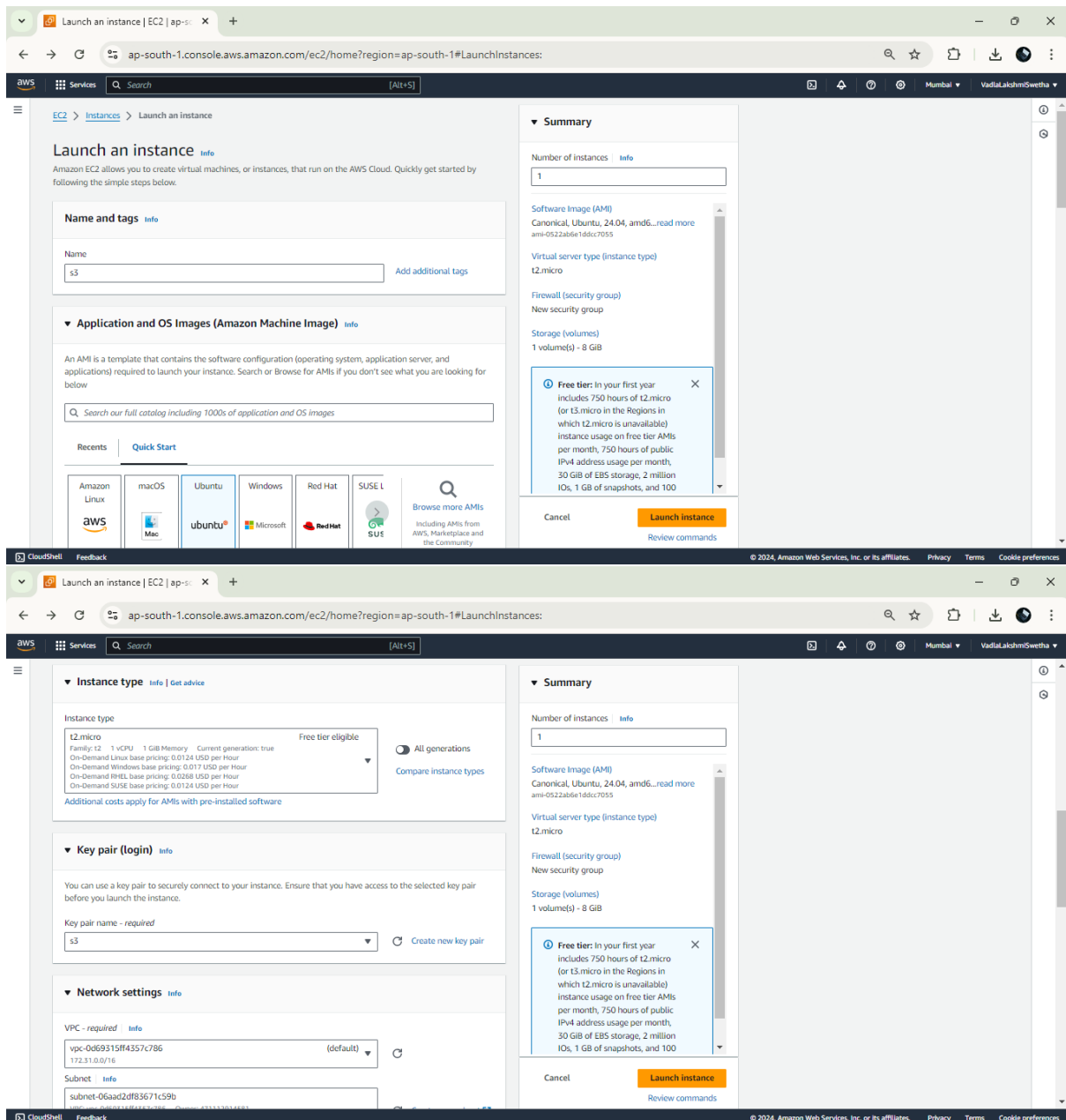


Create s3 bucket and upload a file in s3 bucket using terraform

To create an S3 bucket and upload a file to the bucket using Terraform, follow these steps:



Launch an instance | EC2 | ap-south-1

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:

Network settings

VPC - required

vpc-0d69315ff4357c786 (default)

Subnet

subnet-06aad2df83671c59b

Auto-assign public IP

Enable

Firewall (security group)

Create security group

Security group name - required

s3

Description - required

launch-wizard-1 created 2024-08-23T05:45:49.666Z

Summary

Number of instances

1

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd64...read more

ami-0522ab6e1ddc7055

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million

Launch an instance | EC2 | ap-south-1

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:

Success

Successfully initiated launch of Instance (i-06cd0d253468c60d2)

Launch log

Instance details | EC2 | ap-south-1

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#InstanceDetails:instanceid=i-06cd0d253468c60d2

Search results for 's3'

Services (8)

Features (39)

Resources (New)

Documentation (27,056)

Knowledge Articles (288)

Marketplace (1,897)

Blogs (1,428)

Events (26)

Tutorials (12)

Services

S3 Scalable Storage in the Cloud

S3 Glacier Archive Storage in the Cloud

AWS Snow Family Large Scale Data Transport

Storage Gateway Hybrid Storage Integration

Features

Imports from S3

Homepage | S3 | ap-south-1 | EC2 Instance Connect | ap-south-1 | aws_instance | Resources | hashicorp-certified-terraform-... | hashicorp-certified-terraform-...

ap-south-1.console.aws.amazon.com/s3/get-started?region=ap-south-1

Storage

Amazon S3

Store and retrieve any amount of data from anywhere

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.

Create a bucket

Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.

Create bucket

Pricing

With S3, there are no minimum fees. You only pay for what you use. Prices are based on the location of your S3 bucket.

Estimate your monthly bill using the [AWS Simple Monthly Calculator](#)

How it works

Introduction to Amazon S3

Copy link

CloudShell Feedback

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

Create S3 bucket | S3 | ap-south-1 | EC2 Instance Connect | ap-south-1 | aws_instance | Resources | hashicorp-certified-terraform-... | hashicorp-certified-terraform-...

ap-south-1.console.aws.amazon.com/s3/bucket/create?region=ap-south-1&bucketType=general

Create bucket

Buckets are containers for data stored in S3.

General configuration

AWS Region
Asia Pacific (Mumbai) ap-south-1

Bucket name [Info](#)
s3-taskk

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

Copy settings from existing bucket - optional
Only the bucket settings in the following configuration are copied.

Choose bucket

Format: s3://bucket/prefix

Create S3 bucket | S3 | ap-south-1 | EC2 Instance Connect | ap-south-1 | aws_instance | Resources | hashicorp-certified-terraform-... | hashicorp-certified-terraform-...

ap-south-1.console.aws.amazon.com/s3/bucket/create?region=ap-south-1&bucketType=general

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

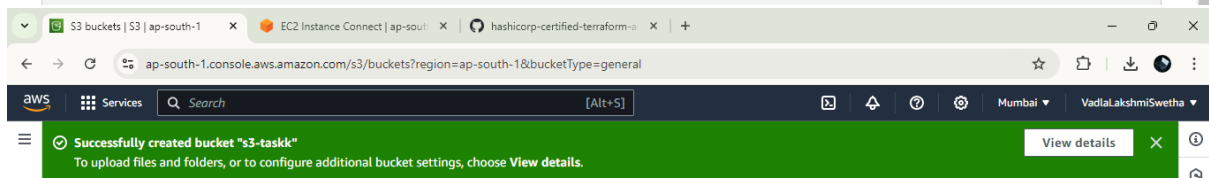
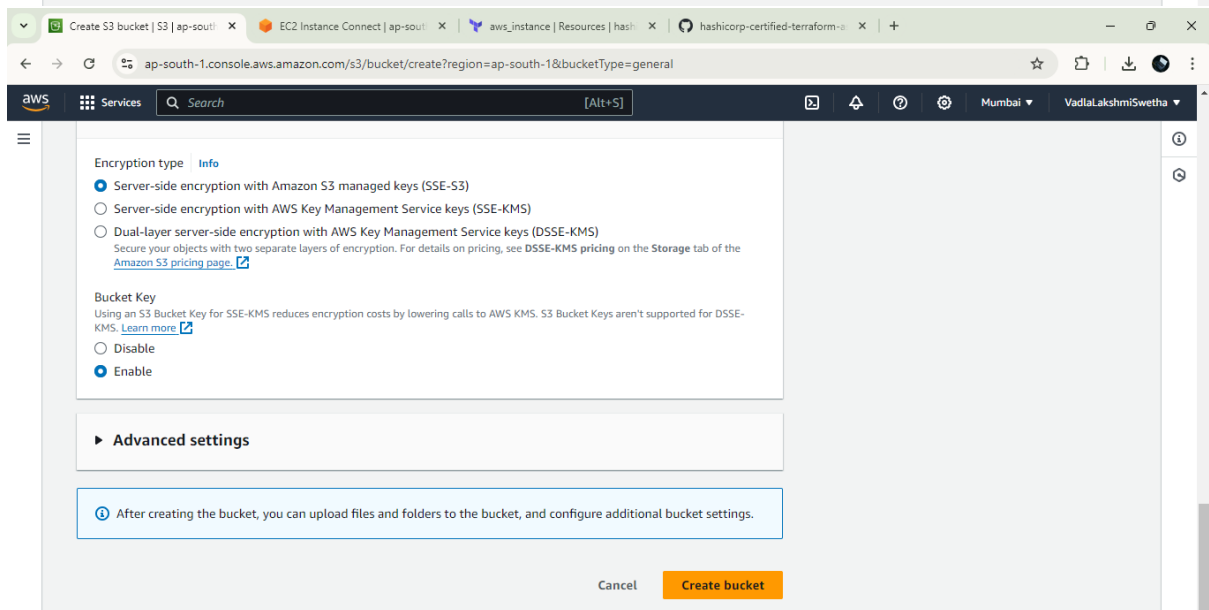
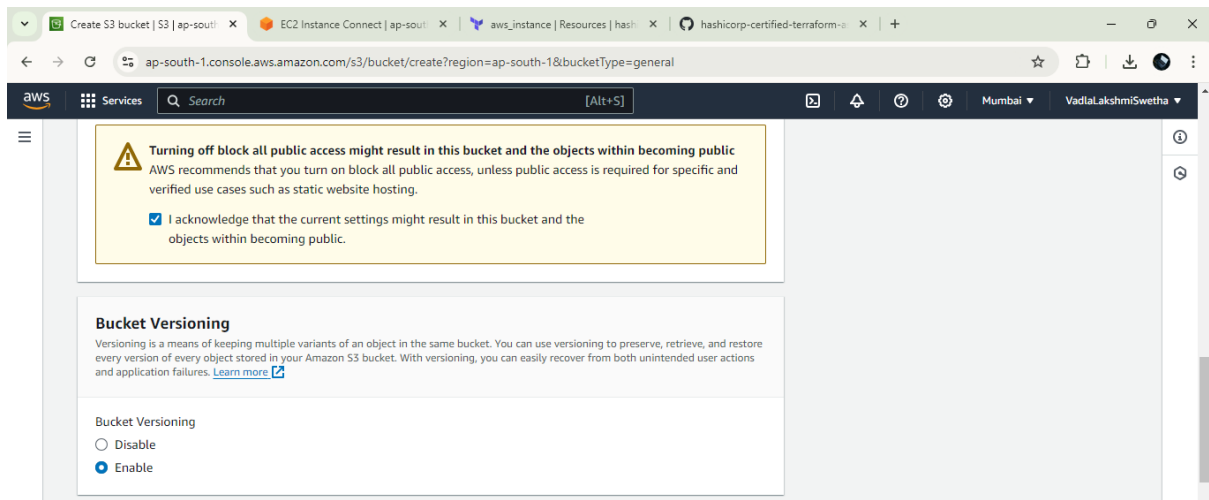
☐ **Block all public access**
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☐ **Block public access to buckets and objects granted through new access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☐ **Block public access to buckets and objects granted through any access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.

☐ **Block public access to buckets and objects granted through new public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

☐ **Block public and cross-account access to buckets and objects through any public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.



S3 buckets | S3 | ap-south-1

EC2 Instance Connect | ap-south-1

hashicorp-certified-terraform-0

ap-south-1.console.aws.amazon.com/s3/buckets?region=ap-south-1

aws

Services

Search

[Alt+S]

Mumbai

VadlaLakshmiSwetha

Amazon S3

Buckets

Access Grants

Access Points

Object Lambda Access Points

Multi-Region Access Points

Batch Operations

IAM Access Analyzer for S3

Block Public Access settings for this account

Storage Lens

Dashboards

Storage Lens groups

AWS Organizations settings

Account snapshot - updated every 24 hours

All AWS Regions

View Storage Lens dashboard

Storage lens provides visibility into storage usage and activity trends.

General purpose buckets

Directory buckets

General purpose buckets (1)

Info

All AWS Regions

Refresh

Copy ARN

Empty

Delete

Create bucket

Buckets are containers for data stored in S3.

Find buckets by name

1

Settings

Name	AWS Region	IAM Access Analyzer
s3-taskk	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1

Instances | EC2 | ap-south-1

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:v=3;\$case=tag:true%5Cclient:false;\$regex=tags:false%5Cclient:false

aws

Services

Search

[Alt+S]

Mumbai

VadlaLakshmiSwetha

EC2 Dashboard

EC2 Global View

Events

Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

Network & Security

Security Groups

Elastic IPs

Placement Groups

Instances (1)

Info

Last updated less than a minute ago

Refresh

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	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elasti
s3	i-06cd0d253468c60d2	Running	t2.micro	Initializing	View alarms	ap-south-1a	ec2-13-200-253-235.ap...	13.200.253.235	-

Select an instance

CloudShell

Feedback

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

The image displays two screenshots of an AWS CloudShell terminal session. The top screenshot shows the initial login to an Ubuntu instance with system information and security updates. The bottom screenshot shows the continuation of the session where the user runs 'sudo -i' to become root and then 'apt update -y' followed by 'apt install unzip -y'.

```
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/pro

System information as of Fri Aug 23 06:21:30 UTC 2024

System load:  0.08           Processes:      106
Usage of /:   34.9% of 6.71GB Users logged in:  0
Memory usage: 22%          IPv4 address for enx0: 172.31.38.86
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

40 updates can be applied immediately.
27 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

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

i-06cd0d253468c60d2 (s3)
PublicIPs: 13.200.253.235  PrivateIPs: 172.31.38.86

CloudShell Feedback

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

ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-06cd0d253468c60d2&osUser=ubuntu...

AWS Services Search [Alt+S] Mumbai VadiaLak

Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

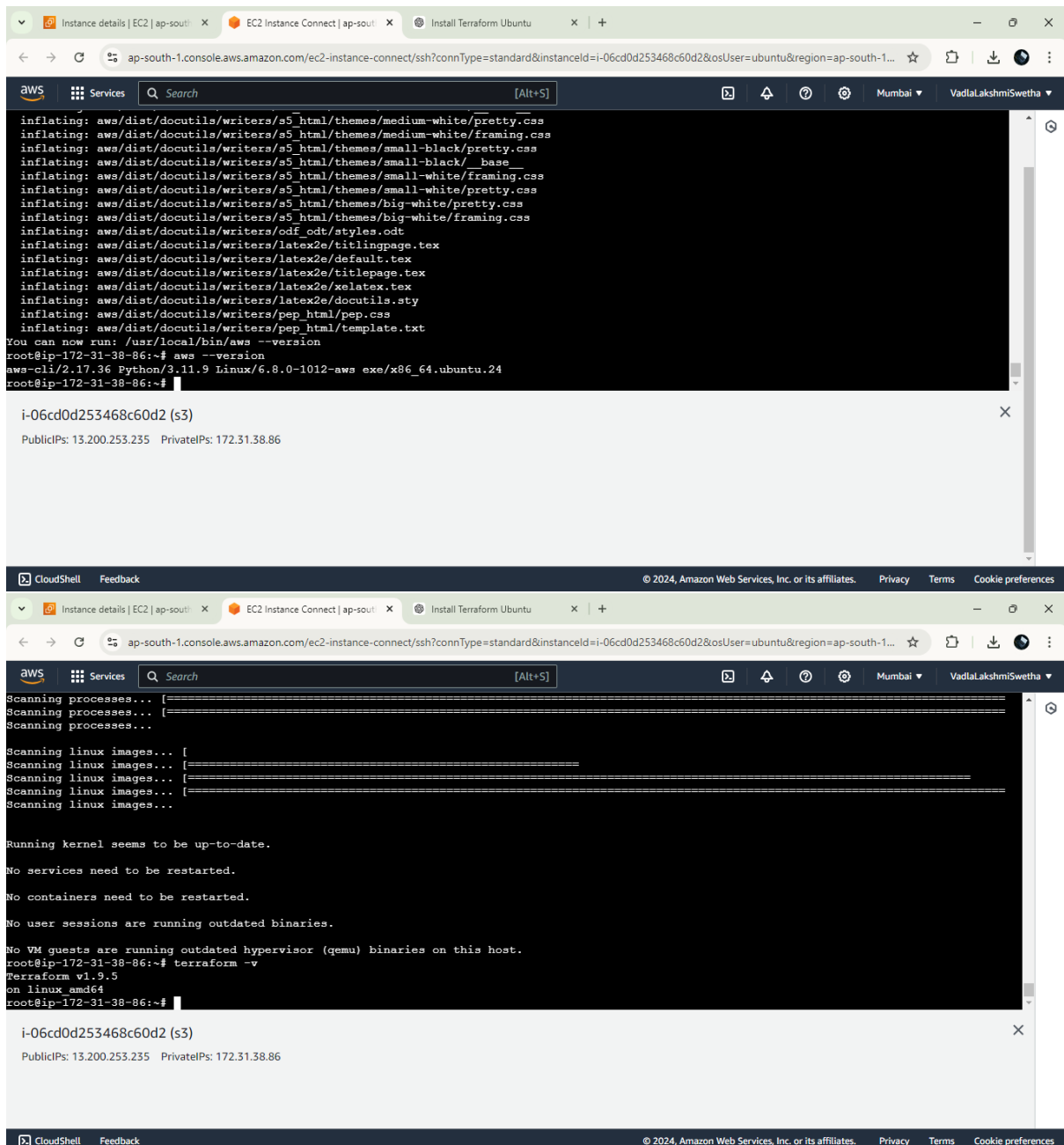
40 updates can be applied immediately.
27 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

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

Last login: Fri Aug 23 05:51:33 2024 from 13.233.177.5
ubuntu@ip-172-31-38-86:~$ sudo -i
root@ip-172-31-38-86:~# apt update -y && apt install unzip -y
```

1. Set Up Terraform

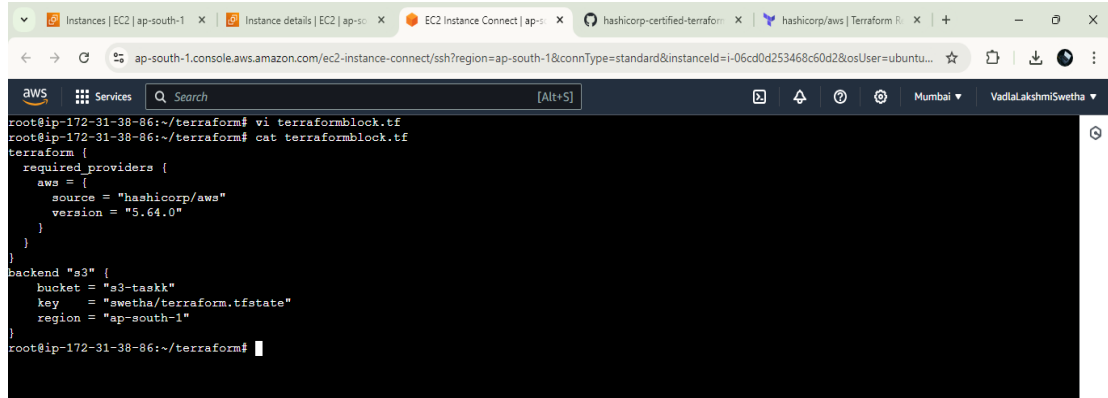
- Ensure you have Terraform installed on your local machine.
- Create a working directory for your Terraform files.



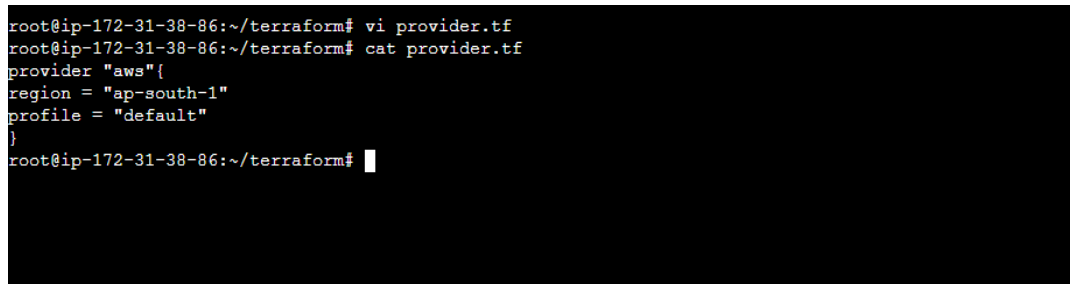
2. Create the Terraform Configuration File

- Create a new file called terraformblock.tf in your working directory.

- Add the following configuration to terraformblock.tf to create an S3 bucket:



```
root@ip-172-31-38-86:~/terraform# vi terraformblock.tf
root@ip-172-31-38-86:~/terraform# cat terraformblock.tf
terraform {
  required_providers {
    aws = [
      source = "hashicorp/aws"
      version = "5.64.0"
    ]
  }
}
backend "s3" {
  bucket = "s3-taskk"
  key    = "swetha/terraform.tfstate"
  region = "ap-south-1"
}
root@ip-172-31-38-86:~/terraform#
```



```
root@ip-172-31-38-86:~/terraform# vi provider.tf
root@ip-172-31-38-86:~/terraform# cat provider.tf
provider "aws" {
  region = "ap-south-1"
  profile = "default"
}
root@ip-172-31-38-86:~/terraform#
```

- Replace the bucket name with a globally unique name.
- Adjust the region, source, and key as needed.

3. Initialize Terraform

- Run the following command to initialize Terraform. This will download the necessary provider plugins:

terraform init

The screenshot displays the AWS CloudShell interface with a terminal window. The terminal shows the execution of Terraform commands on an Ubuntu instance. The first command is 'terraform init', which initializes the Terraform backend and provider plugins. The output indicates that the AWS provider v5.64.0 is installed. The second command is 'terraform validate', which returns a success message stating that the configuration is valid. The terminal window is titled 'i-06cd0d253468c60d2 (s3)' and shows the public and private IP addresses of the instance. The AWS CloudShell interface includes a search bar, a list of services, and a footer with copyright information and links to privacy, terms, and cookie preferences.

```
root@ip-172-31-38-86:~/terraform# terraform init
Initializing the backend...
Initializing provider plugins...
- Finding hashicorp/aws versions matching "5.64.0"...
- Installing hashicorp/aws v5.64.0...
+ Installed hashicorp/aws v5.64.0 (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.
root@ip-172-31-38-86:~/terraform# terraform validate
Success! The configuration is valid.
root@ip-172-31-38-86:~/terraform#
```

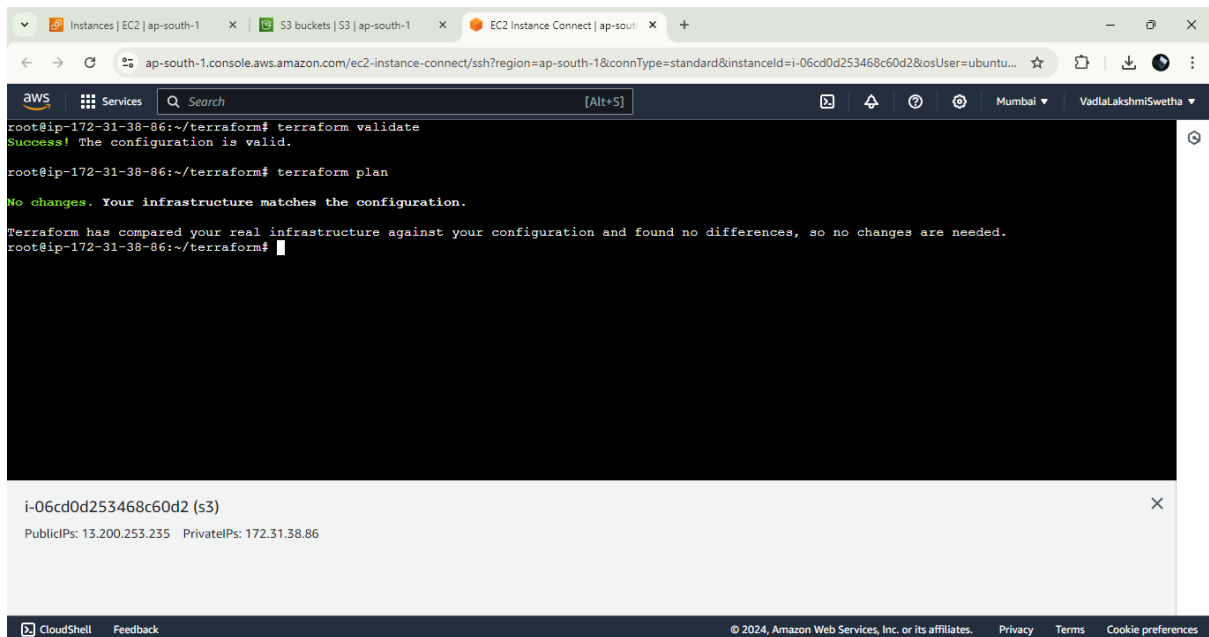
i-06cd0d253468c60d2 (s3)
PublicIPs: 13.200.253.235 PrivateIPs: 172.31.38.86

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

4. Plan the Infrastructure

- Run the following command to see what resources will be created:

terraform plan



The screenshot shows the AWS CloudShell interface. The terminal window displays the following commands and output:

```
root@ip-172-31-38-86:~/terraform# terraform validate
Success! The configuration is valid.

root@ip-172-31-38-86:~/terraform# terraform plan

No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no changes are needed.
root@ip-172-31-38-86:~/terraform#
```

Below the terminal window, the instance details are shown:

i-06cd0d253468c60d2 (s3)
PublicIPs: 13.200.253.235 PrivateIPs: 172.31.38.86

5. Apply the Configuration

- Apply the Terraform configuration to create the S3 bucket and upload the file:
terraform apply
- Type yes when prompted to confirm the changes.

6. Verify the S3 Bucket and Uploaded File

After the apply is complete, you can verify the S3 bucket and the uploaded file in the AWS Management Console.

Instances | EC2 | ap-south-1

S3 buckets | S3 | ap-south-1

+

ap-south-1.console.aws.amazon.com/s3/buckets?region=ap-south-1

aws

Services

Search

[Alt+S]

Mumbai

VadlaLakshmiSwetha

Amazon S3

Buckets

Access Grants

Access Points

Object Lambda Access Points

Multi-Region Access Points

Batch Operations

IAM Access Analyzer for S3

Block Public Access settings for this account

Storage Lens

Dashboards

Storage Lens groups

AWS Organizations settings

Amazon S3 > Buckets

Account snapshot - updated every 24 hours

All AWS Regions

View Storage Lens dashboard

General purpose buckets

Directory buckets

General purpose buckets (1)

Info

All AWS Regions

Refresh

Copy ARN

Empty

Delete

Create bucket

Buckets are containers for data stored in S3.

Find buckets by name

< 1 >

Settings

	Name	AWS Region	IAM Access Analyzer	Creation date
<input type="radio"/>	s3-taskk	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1	August 23, 2024, 11:42:32 (UTC+05:30)

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences

Instances | EC2 | ap-south-1

s3-taskk - S3 bucket | S3 | ap-south-1

+

ap-south-1.console.aws.amazon.com/s3/buckets/s3-taskk?region=ap-south-1&bucketType=general&tab=objects

aws

Services

Search

[Alt+S]

Mumbai

VadlaLakshmiSwetha

Amazon S3

Buckets

Access Grants

Access Points

Object Lambda Access Points

Multi-Region Access Points

Batch Operations

IAM Access Analyzer for S3

Block Public Access settings for this account

Storage Lens

Dashboards

Storage Lens groups

AWS Organizations settings

Amazon S3 > Buckets > s3-taskk

s3-taskk

Info

Objects

Properties

Permissions

Metrics

Management

Access Points

Objects (1)

Info

Refresh

Copy S3 URI

Copy URL

Download

Open

Delete

Actions

Create folder

Upload

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 your objects, you'll need to explicitly grant them permissions. [Learn more](#)

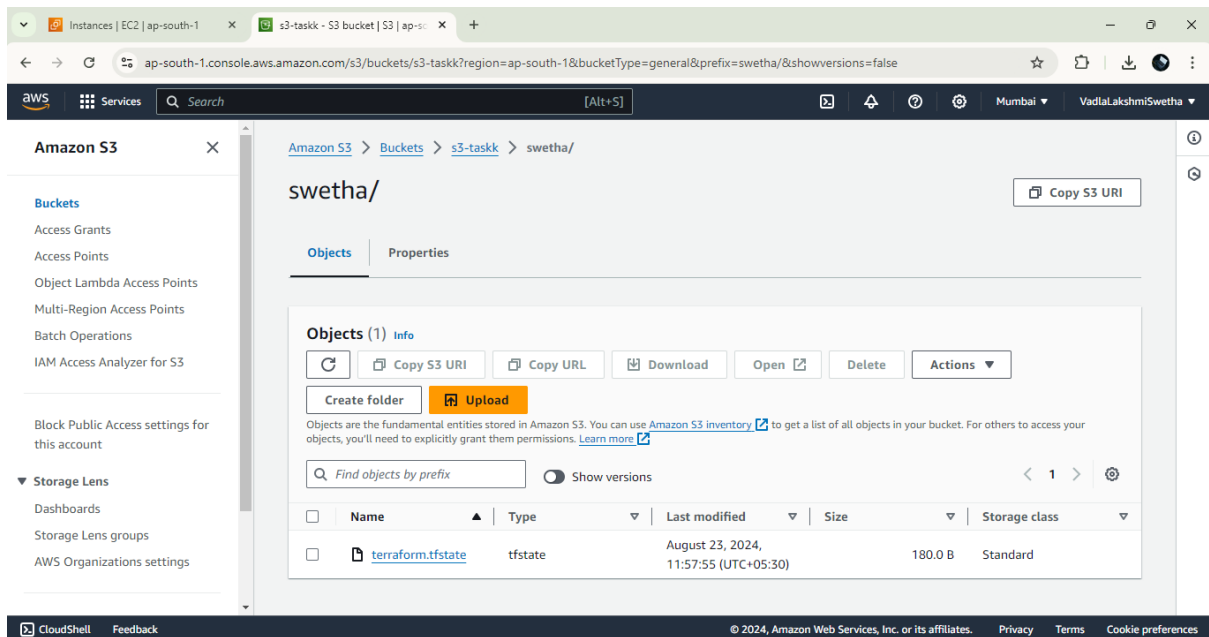
Find objects by prefix

Show versions

< 1 >

Settings

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	swetha/	Folder	-	-	-



7. Cleanup :

If you want to destroy the S3 bucket and the uploaded file, run:

terraform destroy

- Type yes when prompted to confirm the destruction.

This process creates an S3 bucket and uploads a specified file using Terraform.