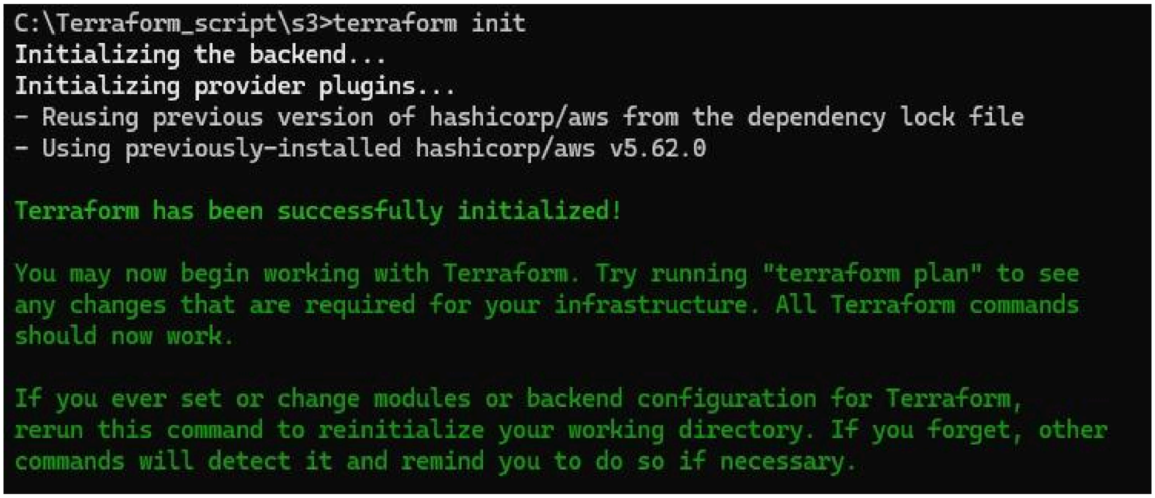
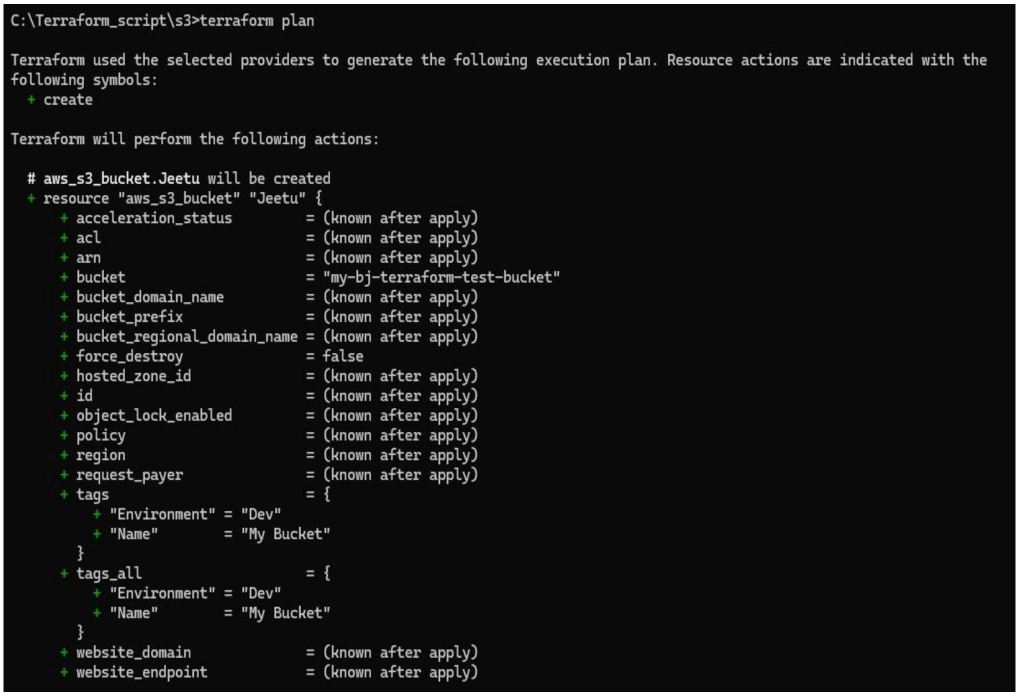
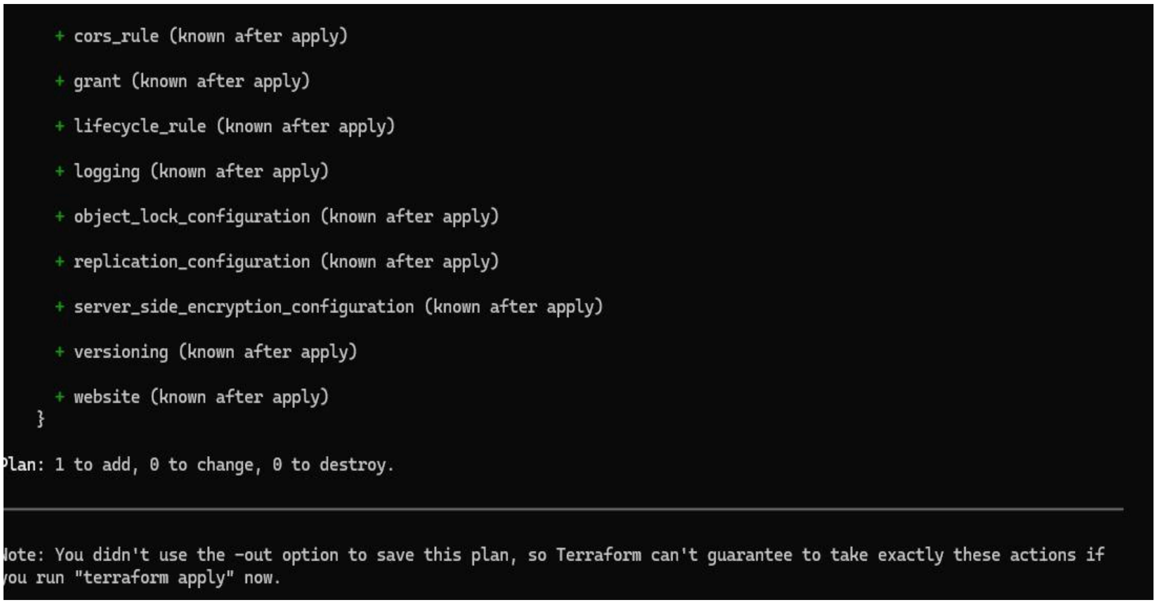


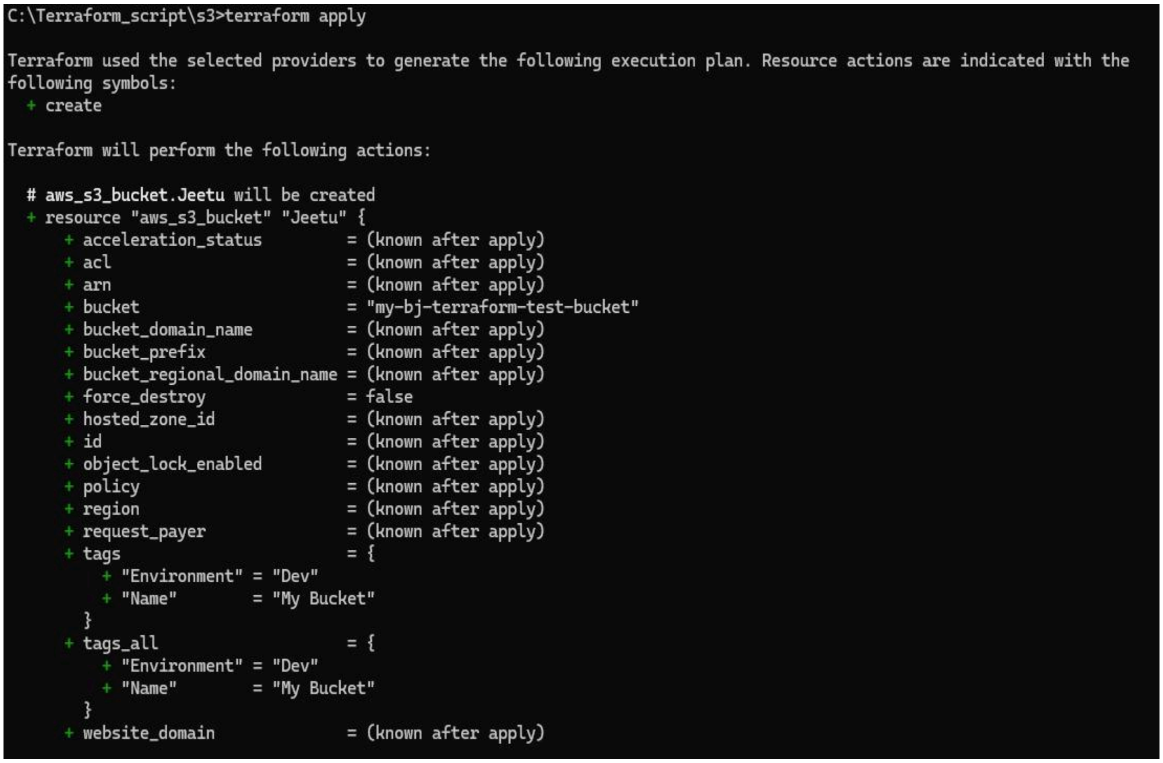
```
C:\Terraform_script\s3>terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.62.0

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.
```



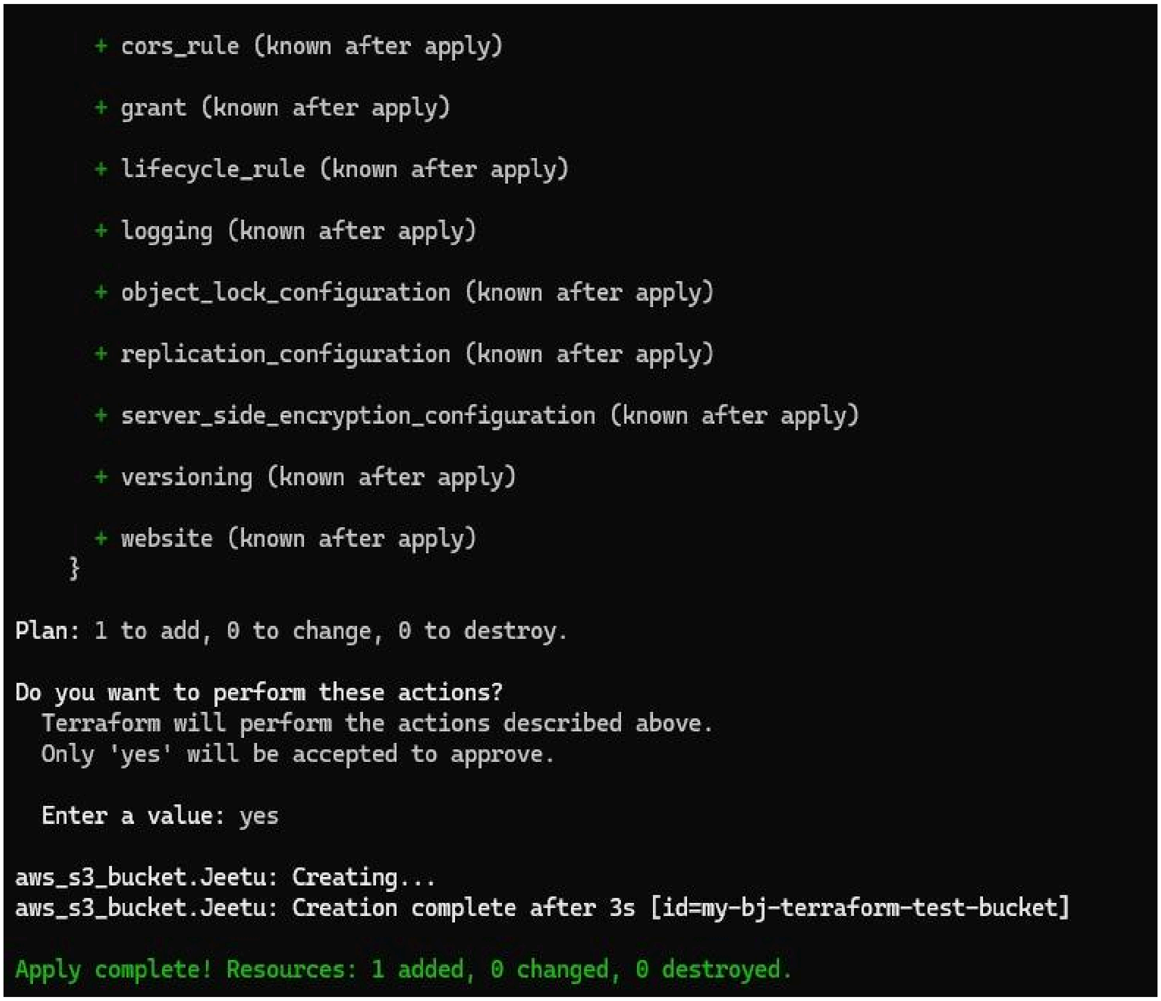
```
C:\Terraform_script\s3>terraform plan

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_s3_bucket.Jeetu will be created
  + resource "aws_s3_bucket" "Jeetu" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "my-bj-terraform-test-bucket"
      + bucket_domain_name          = (known after apply)
      + bucket_prefix               = (known after apply)
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
      + hosted_zone_id              = (known after apply)
      + id                          = (known after apply)
      + object_lock_enabled         = (known after apply)
      + policy                      = (known after apply)
      + region                      = (known after apply)
      + request_payer               = (known after apply)
      + tags                        = {
          + "Environment" = "Dev"
          + "Name"        = "My Bucket"
        }
      + tags_all                    = {
          + "Environment" = "Dev"
          + "Name"        = "My Bucket"
        }
      + website_domain              = (known after apply)
      + website_endpoint            = (known after apply)
```



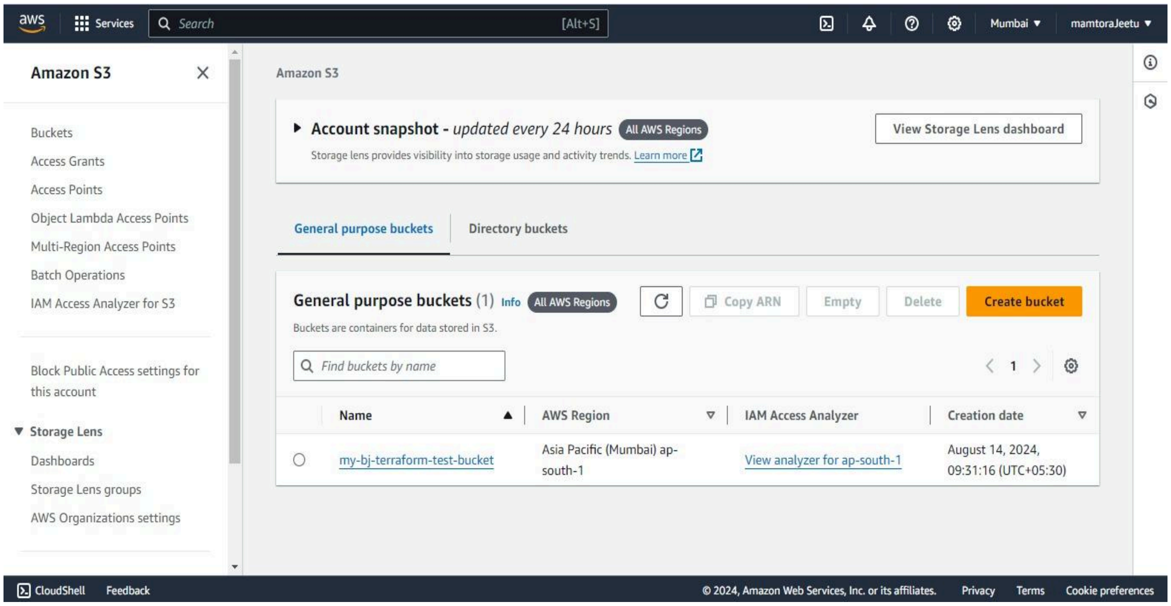
```
      + cors_rule (known after apply)

      + grant (known after apply)

      + lifecycle_rule (known after apply)

      + logging (known after apply)

      + object_lock_configuration (known after apply)

      + replication_configuration (known after apply)

      + server_side_encryption_configuration (known after apply)

      + versioning (known after apply)

      + website (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.
```



```
C:\Terraform_script\s3>terraform apply

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

Terraform will perform the following actions:

  # aws_s3_bucket.Jeetu will be created
  + resource "aws_s3_bucket" "Jeetu" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "my-bj-terraform-test-bucket"
      + bucket_domain_name          = (known after apply)
      + bucket_prefix               = (known after apply)
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
      + hosted_zone_id              = (known after apply)
      + id                          = (known after apply)
      + object_lock_enabled         = (known after apply)
      + policy                      = (known after apply)
      + region                      = (known after apply)
      + request_payer               = (known after apply)
      + tags                        = {
          + "Environment" = "Dev"
          + "Name"        = "My Bucket"
        }
      + tags_all                    = {
          + "Environment" = "Dev"
          + "Name"        = "My Bucket"
        }
      + website_domain              = (known after apply)
```



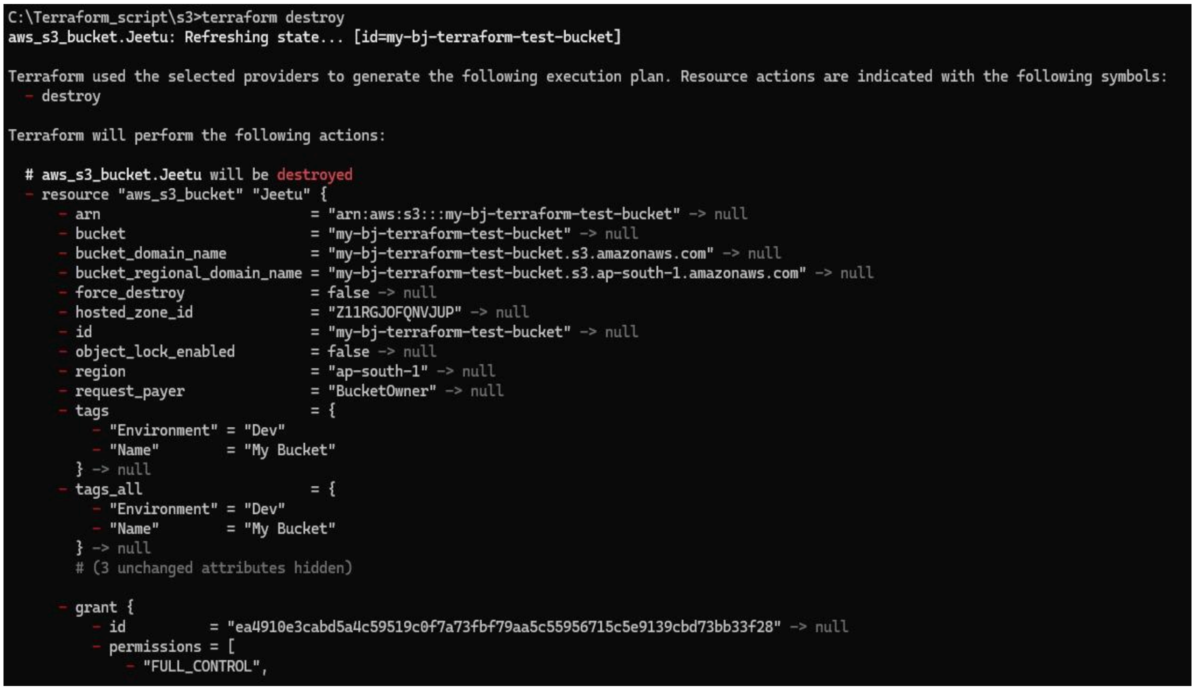
```
      + cors_rule (known after apply)

      + grant (known after apply)

      + lifecycle_rule (known after apply)

      + logging (known after apply)

      + object_lock_configuration (known after apply)

      + replication_configuration (known after apply)

      + server_side_encryption_configuration (known after apply)

      + versioning (known after apply)

      + website (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_s3_bucket.Jeetu: Creating...
aws_s3_bucket.Jeetu: Creation complete after 3s [id=my-bj-terraform-test-bucket]

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


aws
Services
Search
[Alt+S]
Mumbai ▾
mamtoraJeetu ▾
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
Account snapshot - updated every 24 hours
All AWS Regions
View Storage Lens dashboard
Storage lens provides visibility into storage usage and activity trends. Learn more
General purpose buckets
Directory buckets
Copy ARN
Empty
Delete
Create bucket
Info
All AWS Regions
Find buckets by name
1
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

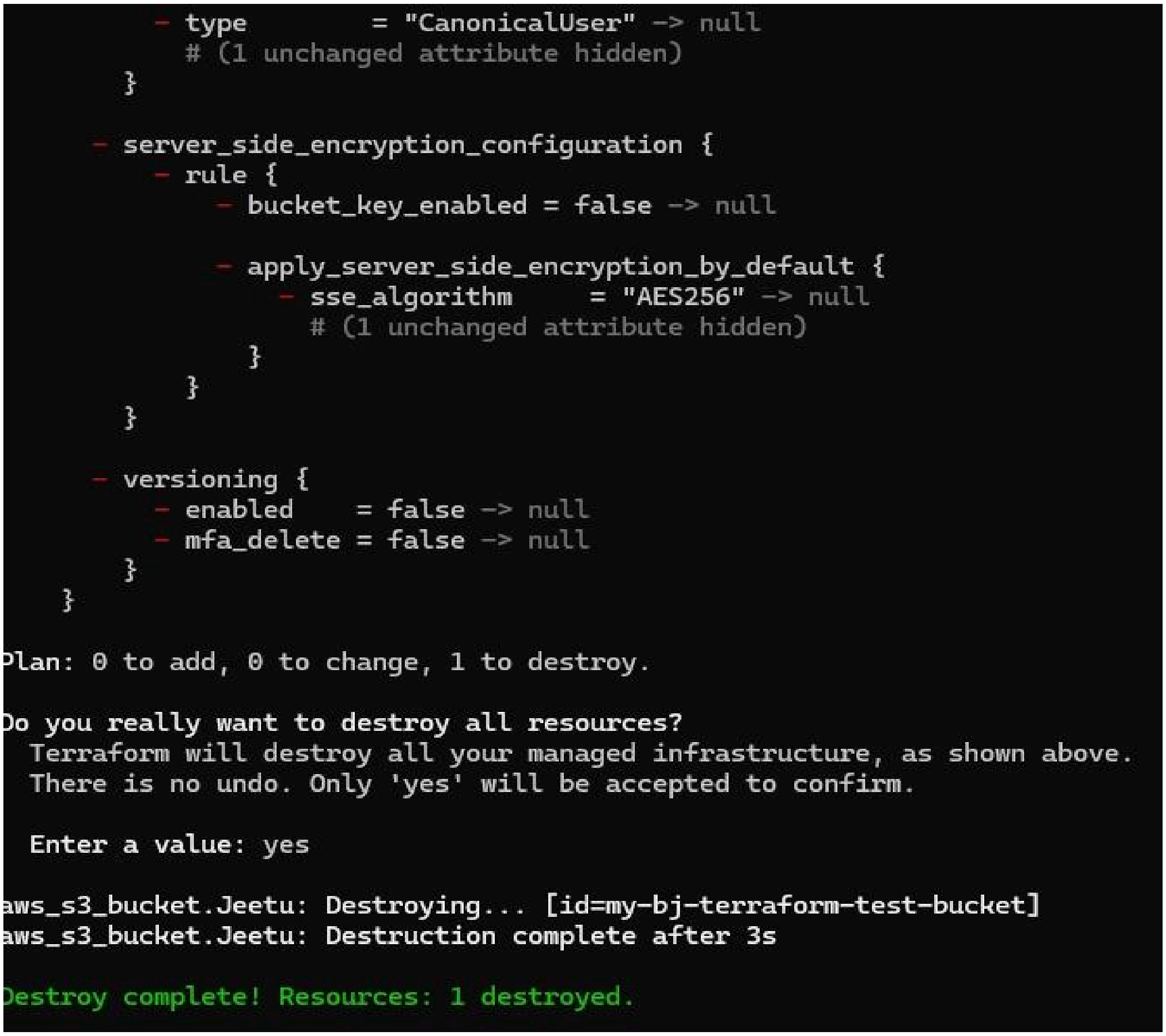
General purpose buckets (1)

Buckets are containers for data stored in S3.

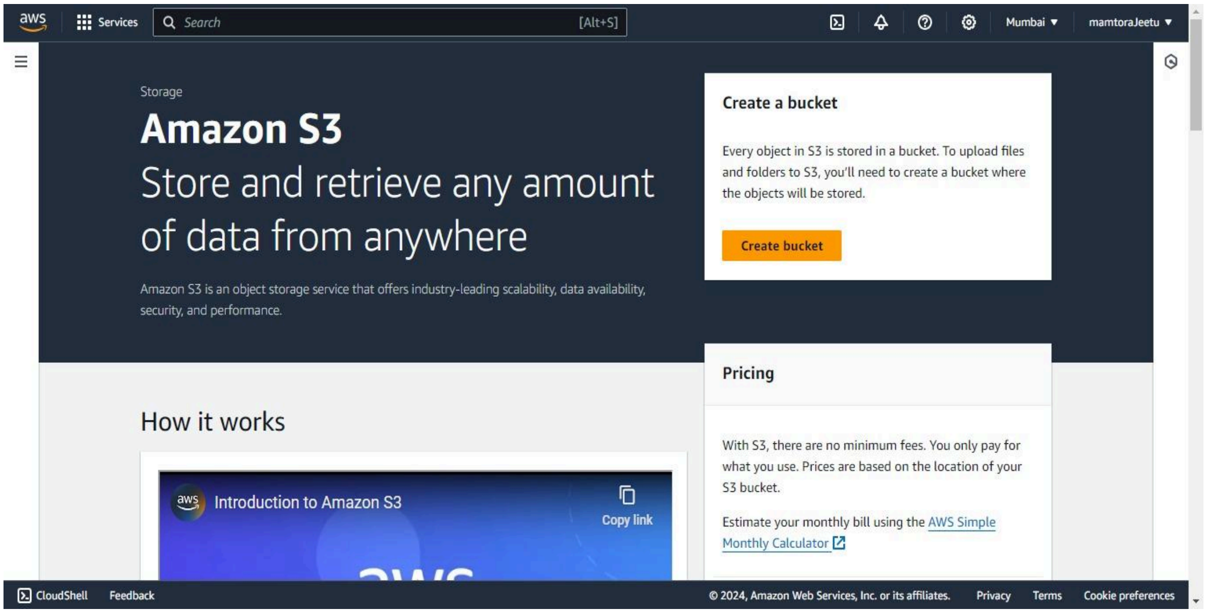
	Name ▲	AWS Region	IAM Access Analyzer	Creation date
○	my-bj-terraform-test-bucket	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1	August 14, 2024, 09:31:16 (UTC+05:30)



```
C:\Terraform_script\s3>terraform destroy
aws_s3_bucket.Jeetu: Refreshing state... [id=my-bj-terraform-test-bucket]

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_s3_bucket.Jeetu will be destroyed
  - resource "aws_s3_bucket" "Jeetu" {
      - arn                         = "arn:aws:s3:::my-bj-terraform-test-bucket" -> null
      - bucket                      = "my-bj-terraform-test-bucket" -> null
      - bucket_domain_name          = "my-bj-terraform-test-bucket.s3.amazonaws.com" -> null
      - bucket_regional_domain_name = "my-bj-terraform-test-bucket.s3.ap-south-1.amazonaws.com" -> null
      - force_destroy               = false -> null
      - hosted_zone_id              = "Z11RGJOFQNVJUP" -> null
      - id                          = "my-bj-terraform-test-bucket" -> null
      - object_lock_enabled         = false -> null
      - region                      = "ap-south-1" -> null
      - request_payer               = "BucketOwner" -> null
      - tags                        = {
          - "Environment" = "Dev"
          - "Name"        = "My Bucket"
        } -> null
      - tags_all                    = {
          - "Environment" = "Dev"
          - "Name"        = "My Bucket"
        } -> null
        # (3 unchanged attributes hidden)

      - grant {
          - id          = "ea4910e3cabd5a4c59519c0f7a73fbf79aa5c55956715c5e9139cbd73bb33f28" -> null
          - permissions = [
              - "FULL_CONTROL",
```



```
          - type         = "CanonicalUser" -> null
            # (1 unchanged attribute hidden)
        }

      - server_side_encryption_configuration {
          - rule {
              - bucket_key_enabled = false -> null

              - apply_server_side_encryption_by_default {
                  - sse_algorithm     = "AES256" -> null
                    # (1 unchanged attribute hidden)
                }
            }
        }

      - versioning {
          - enabled    = false -> null
          - mfa_delete = false -> null
        }
    }

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_s3_bucket.Jeetu: Destroying... [id=my-bj-terraform-test-bucket]
aws_s3_bucket.Jeetu: Destruction complete after 3s

Destroy complete! Resources: 1 destroyed.
```



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.

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