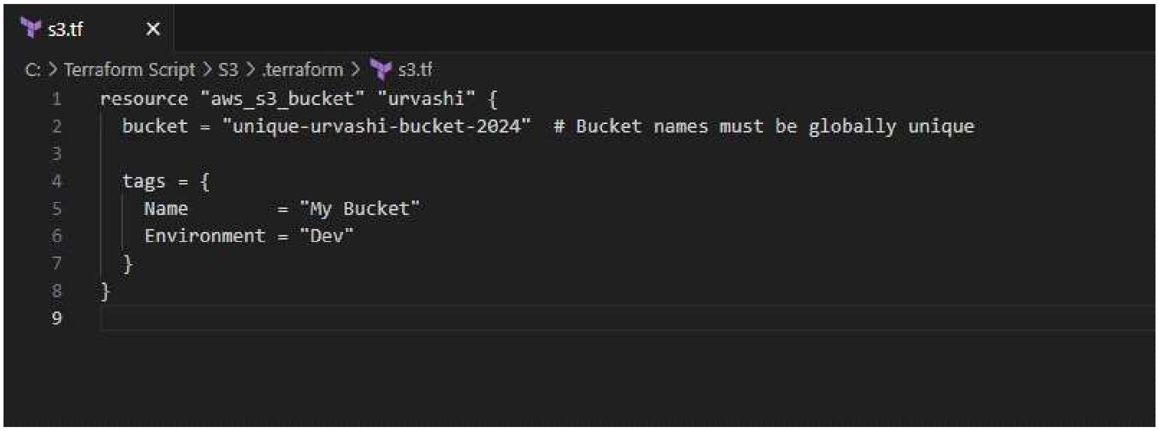
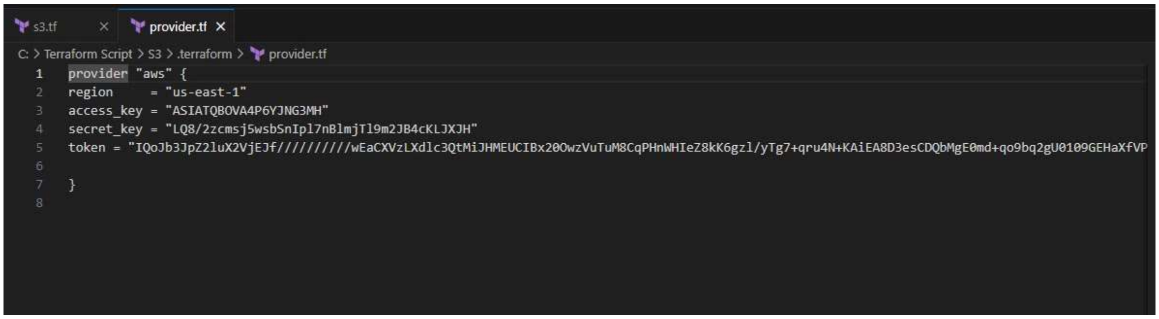
Name:Urvashi Changlani
D15B/08

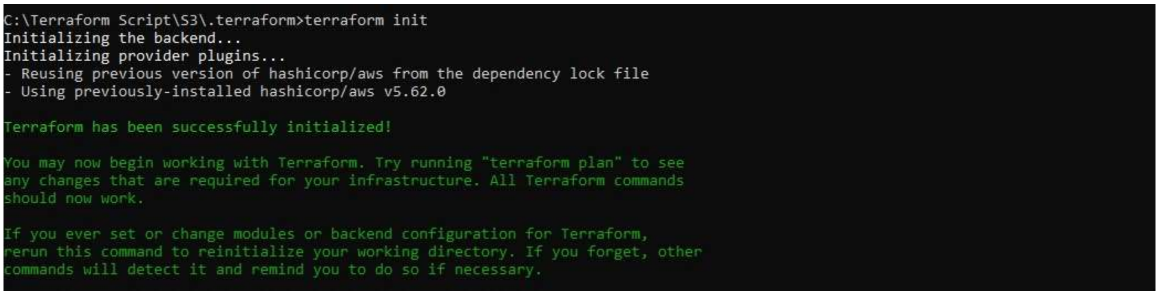
Experiment No 6



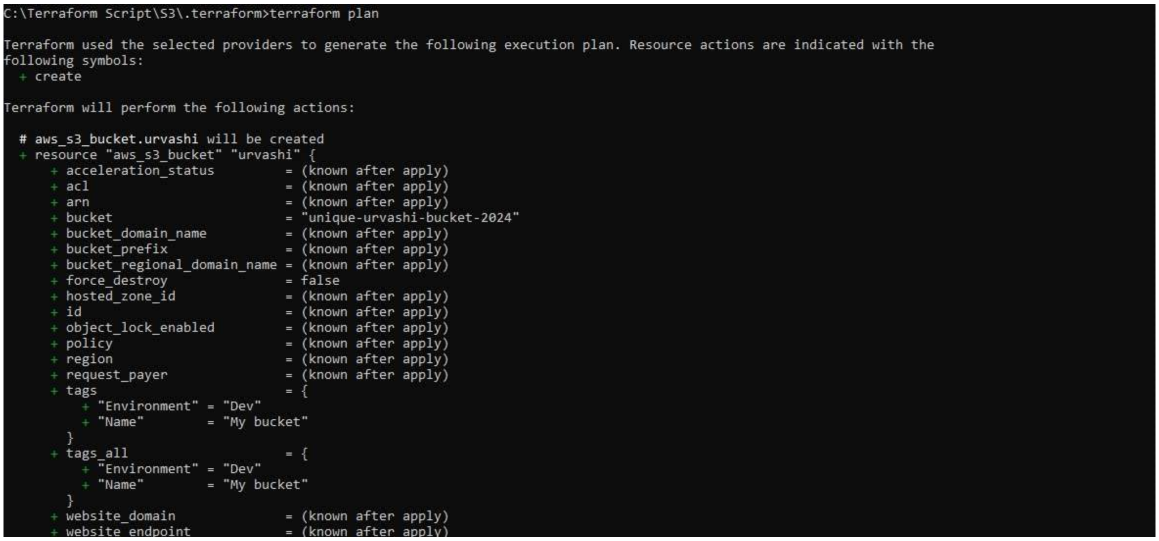
```
resource "aws_s3_bucket" "urvashi" {
  bucket = "unique-urvashi-bucket-2024"  # Bucket names must be globally unique

  tags = {
    Name        = "My Bucket"
    Environment = "Dev"
  }
}
```



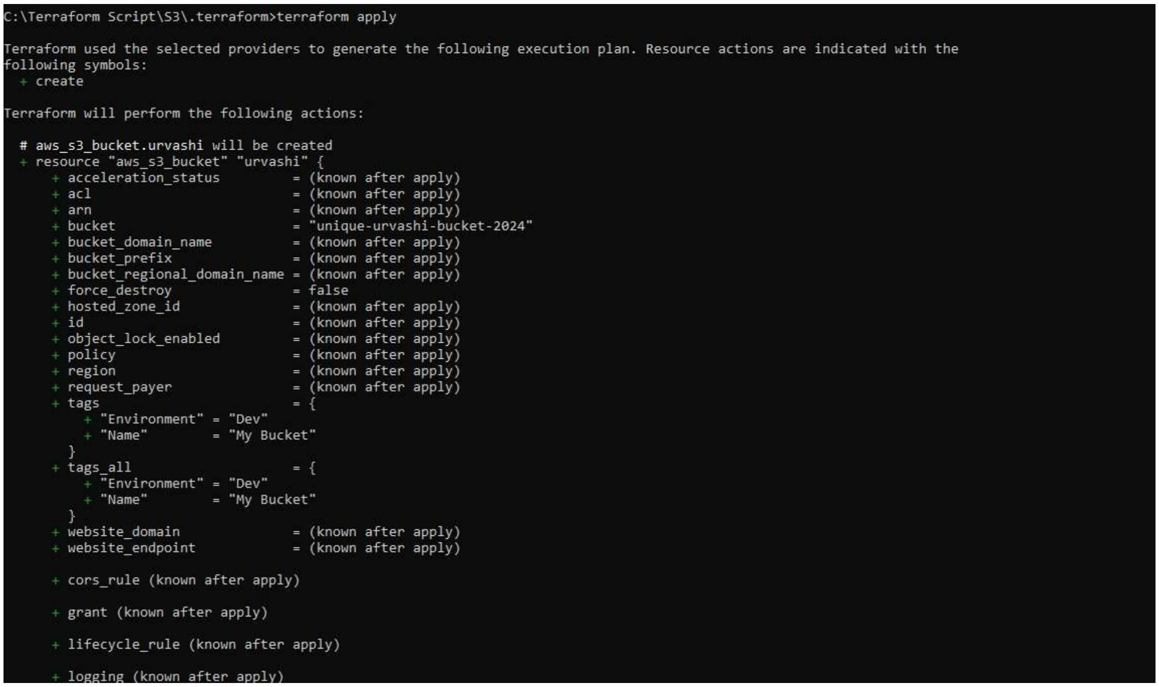
```
provider "aws" {
region     = "us-east-1"
access_key = "ASIATQBOVA4P6YJNG3MH"
secret_key = "LQ8/2zcmsj5wsbSnIpl7nBlmjTl9m2JB4cKLJXJH"
token = "IQoJb3JpZ2luX2VjEJf//////////wEaCXVzLXdlc3QtMiJHMEUCIBx20OwzVuTuM8CqPHnWHIeZ8kK6gzl/yTg7+qru4N+KAiEA8D3esCDQbMgE0md+qo9bq2gU0109GEHaXfVP

}
```



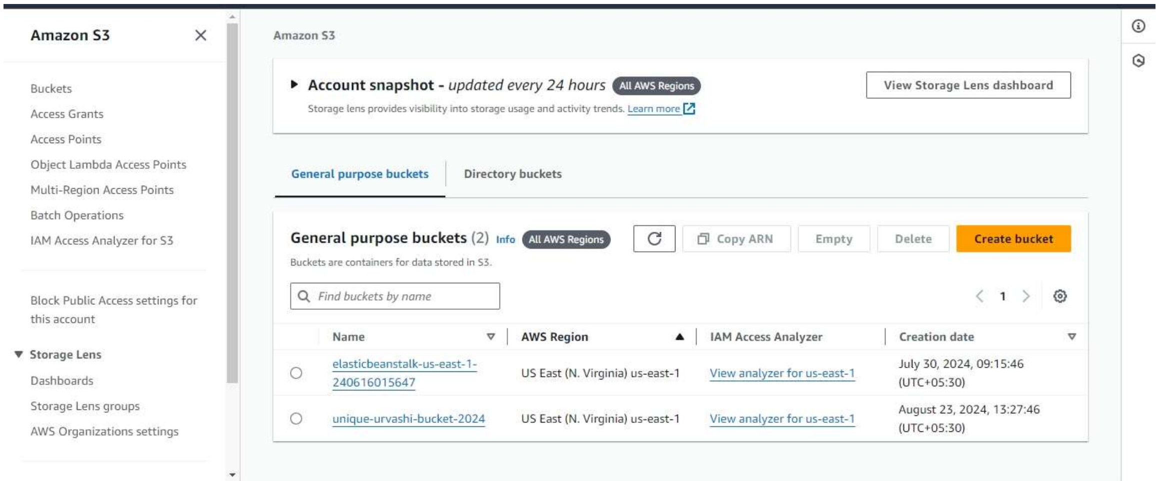
```
C:\Terraform Script\S3\.terraform>terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.62.0

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>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.urvashi will be created
  + resource "aws_s3_bucket" "urvashi" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "unique-urvashi-bucket-2024"
      + 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)
```



```
C:\Terraform Script\S3\.terraform>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.urvashi will be created
  + resource "aws_s3_bucket" "urvashi" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "unique-urvashi-bucket-2024"
      + 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)
```



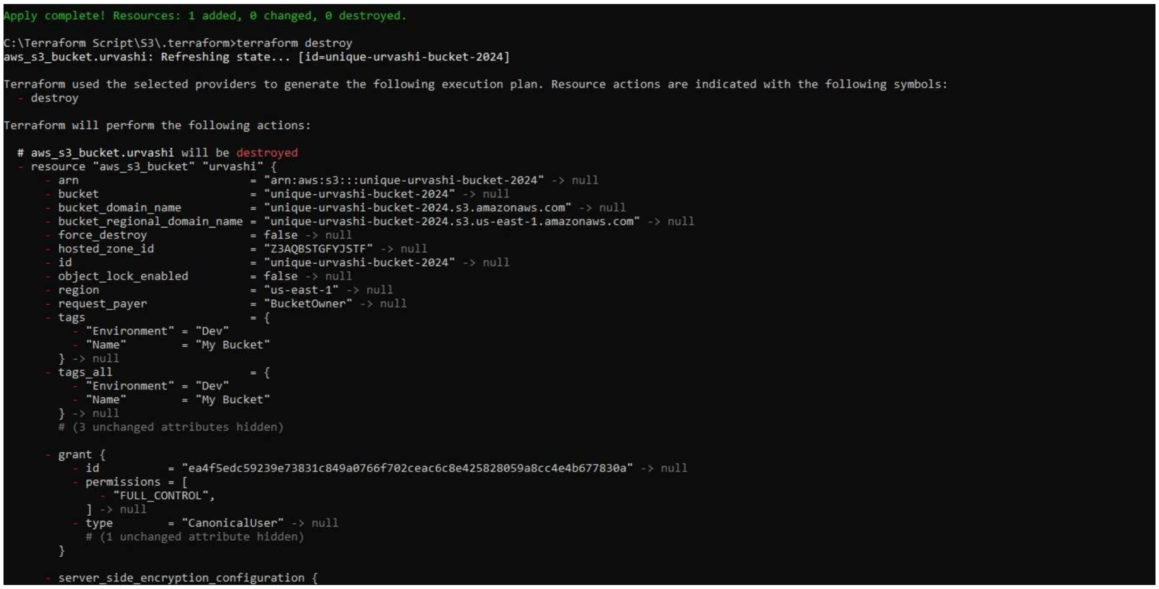
Account snapshot - *updated every 24 hours* All AWS Regions

Storage lens provides visibility into storage usage and activity trends. Learn more

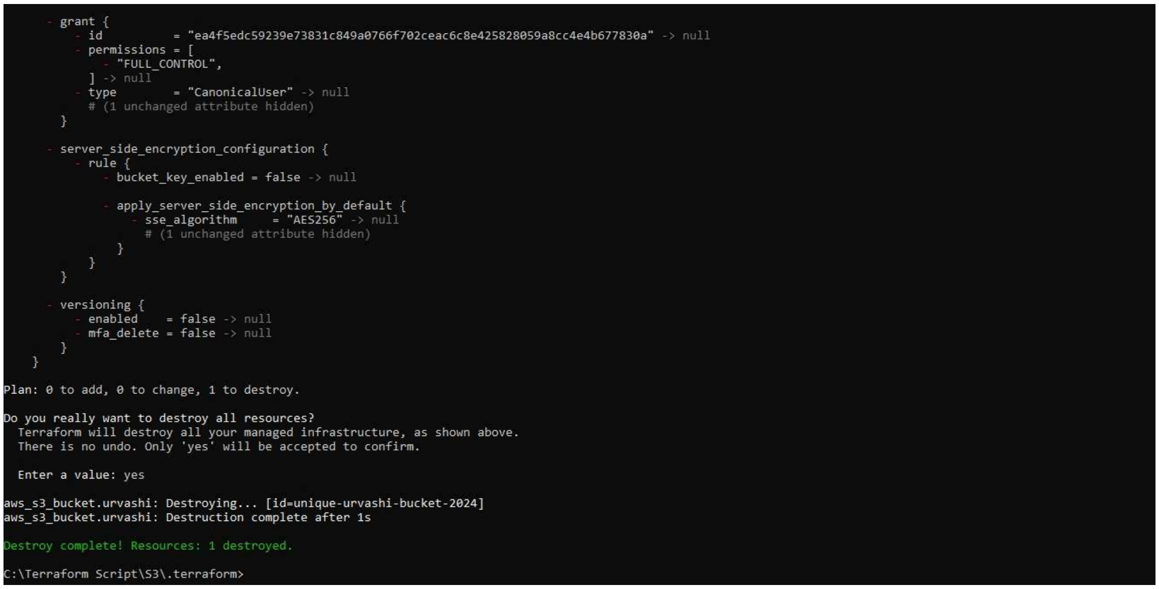
General purpose buckets (2)

Buckets are containers for data stored in S3.

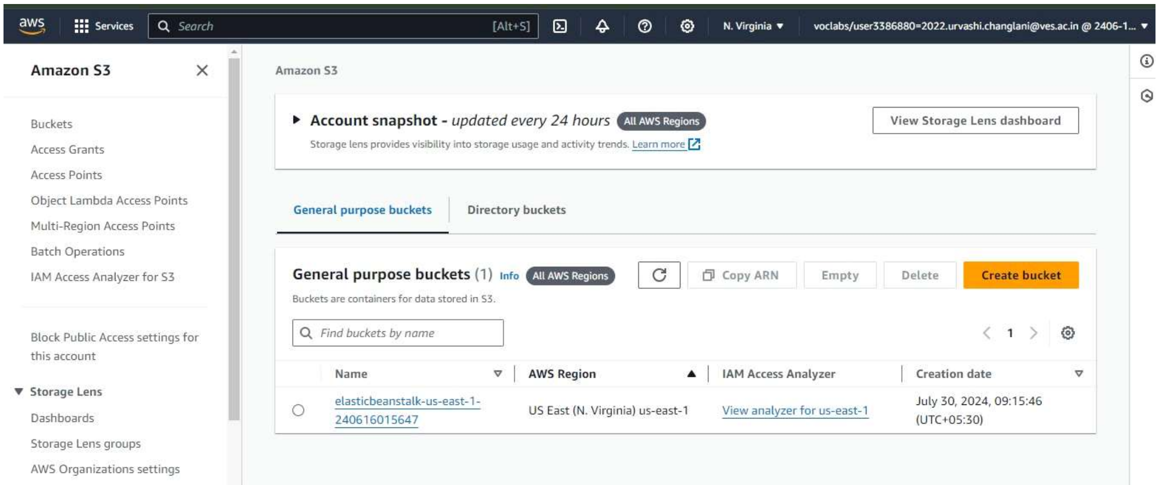
	Name	AWS Region	IAM Access Analyzer	Creation date
○	elasticbeanstalk-us-east-1-240616015647	US East (N. Virginia) us-east-1	View analyzer for us-east-1	July 30, 2024, 09:15:46 (UTC+05:30)
○	unique-urvashi-bucket-2024	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 13:27:46 (UTC+05:30)



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

C:\Terraform Script\S3\.terraform>terraform destroy
aws_s3_bucket.urvashi: Refreshing state... [id=unique-urvashi-bucket-2024]

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.urvashi will be destroyed
  - resource "aws_s3_bucket" "urvashi" {
      - arn                         = "arn:aws:s3:::unique-urvashi-bucket-2024" -> null
      - bucket                      = "unique-urvashi-bucket-2024" -> null
      - bucket_domain_name          = "unique-urvashi-bucket-2024.s3.amazonaws.com" -> null
      - bucket_regional_domain_name = "unique-urvashi-bucket-2024.s3.us-east-1.amazonaws.com" -> null
      - force_destroy               = false -> null
      - hosted_zone_id              = "Z3AQBSTGFYJSTF" -> null
      - id                          = "unique-urvashi-bucket-2024" -> null
      - object_lock_enabled         = false -> null
      - region                      = "us-east-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          = "ea4f5edc59239e73831c849a0766f702ceac6c8e425828059a8cc4e4b677830a" -> null
          - permissions = [
              - "FULL_CONTROL",
            ] -> null
          - type        = "CanonicalUser" -> null
            # (1 unchanged attribute hidden)
        }

      - server_side_encryption_configuration {
```



```
      - grant {
          - id          = "ea4f5edc59239e73831c849a0766f702ceac6c8e425828059a8cc4e4b677830a" -> null
          - permissions = [
              - "FULL_CONTROL",
            ] -> null
          - 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.urvashi: Destroying... [id=unique-urvashi-bucket-2024]
aws_s3_bucket.urvashi: Destruction complete after 1s

Destroy complete! Resources: 1 destroyed.

C:\Terraform Script\S3\.terraform>
```



aws
Services
Search
[Alt+S]
N. Virginia ▾
voclabs/user3386880=2022.urvashi.changlani@ves.ac.in @ 2406-1... ▾

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

Storage lens provides visibility into storage usage and activity trends. Learn more

View Storage Lens dashboard

General purpose buckets | Directory buckets

General purpose buckets (1) Info All AWS Regions

Copy ARN | Empty | Delete | Create bucket

Buckets are containers for data stored in S3.

Find buckets by name

	Name	AWS Region	IAM Access Analyzer	Creation date
○	elasticbeanstalk-us-east-1-240616015647	US East (N. Virginia) us-east-1	View analyzer for us-east-1	July 30, 2024, 09:15:46 (UTC+05:30)