Event Buckets with Terraform - Project Documentation

This documentation will guide you through the process of creating an automated infrastructure for scalable event data storage using Terraform. The project involves dynamically provisioning and managing AWS S3 buckets for various event themes.

Prerequisites

1. **Install Terraform:**
   - Follow the official guide [here](https://learn.hashicorp.com/tutorials/terraform/install-cli) to **command:** install Terraform on your local machine.

2. **Configure AWS Credentials**:
   - Set up your AWS credentials by following the guide (https://docs.aws.amazon.com/cli/latest/userguide/cli-configure-quickstart.html).

Project Setup

1. **Create a New Terraform Project:**
   - Open a terminal and navigate to the directory where you want to create the project.
   c**ommand:** mkdir event-buckets-terraform
   command: cd event-buckets-terraform

2. **Initialize Terraform:**
   - Run the following command to initialize the Terraform project.
   command: terraform init

Terraform Configuration

1. **Create `main.tf` File:**
   - Create a file named `main.tf` in your project directory.

```
 main.tf

 Define variables
variable "event_themes" {
  type    = list(string)
  default = ["AdventureTech", "NatureEscape", "DataSummit", "CodeCarnival"]
}

 Create S3 buckets
resource "aws_s3_bucket" "event_buckets" {
  for_each = { for theme in var.event_themes : theme => theme }

  bucket = format("%s-js-bucket", each.value)
}

 Create S3 bucket ACLs
```

```
  resource "aws_s3_bucket_acl" "event_buckets_acl" {
    for_each = aws_s3_bucket.event_buckets

    bucket = each.value.bucket
    acl    = "private"
  }

   Output the names of created buckets
  output "created_buckets" {
    value = [for bucket in aws_s3_bucket.event_buckets : bucket.bucket]
  }
```

AWS Infrastructure Provisioning

1. **Plan Infrastructure Changes:**
   - Run the following command to see the planned changes.
   Command: terraform plan

2. **Apply Infrastructure Changes:**
   - Apply the changes to create S3 buckets.
   Command: terraform apply

- Confirm by typing `yes` when prompted.

Project Maintenance

1. **Adjustments and Modifications**:
   - Make adjustments to the `main.tf` file based on your requirements.

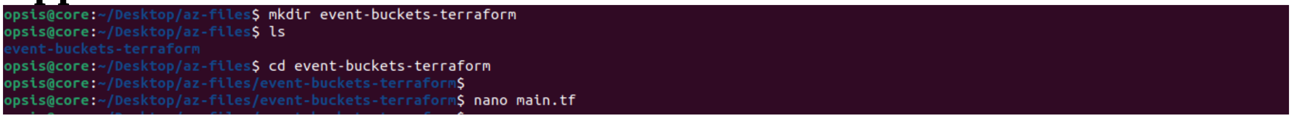
2. **Destroy Infrastructure:**
   - If needed, you can destroy the created infrastructure.
 command: terraform destroy

- Confirm by typing `yes` when prompted.

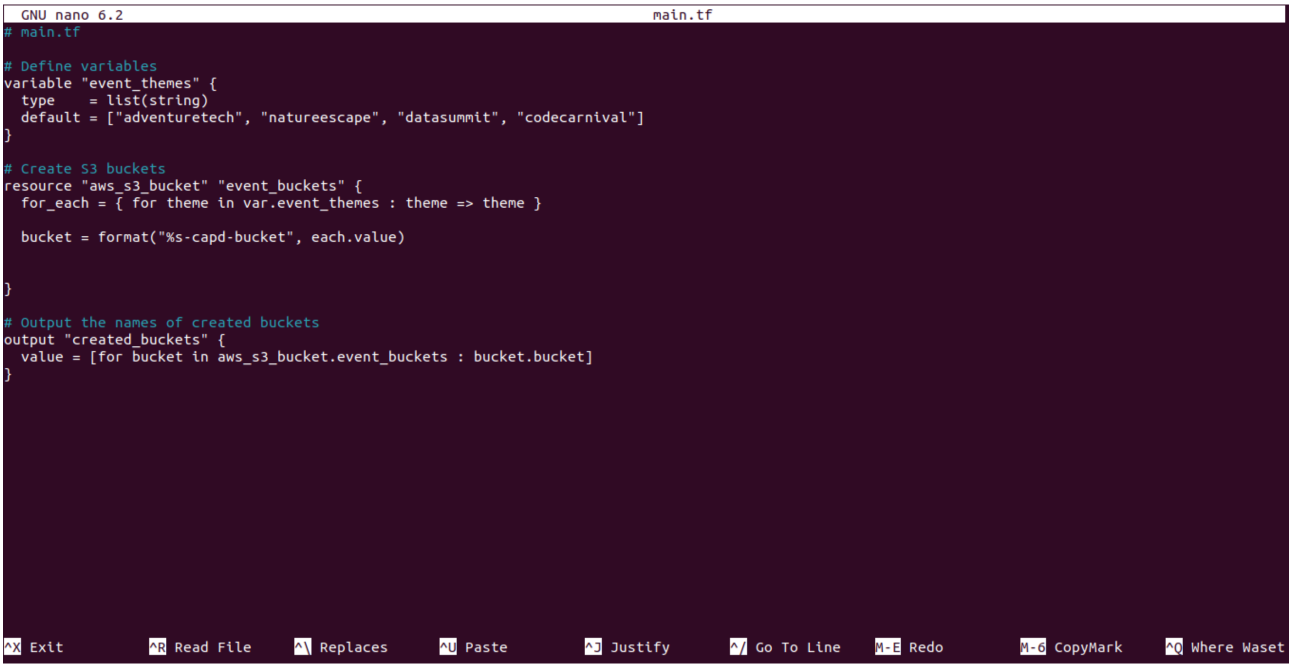
Conclusion

The Event Buckets with Terraform project provides a foundation for dynamically managing AWS S3 buckets.

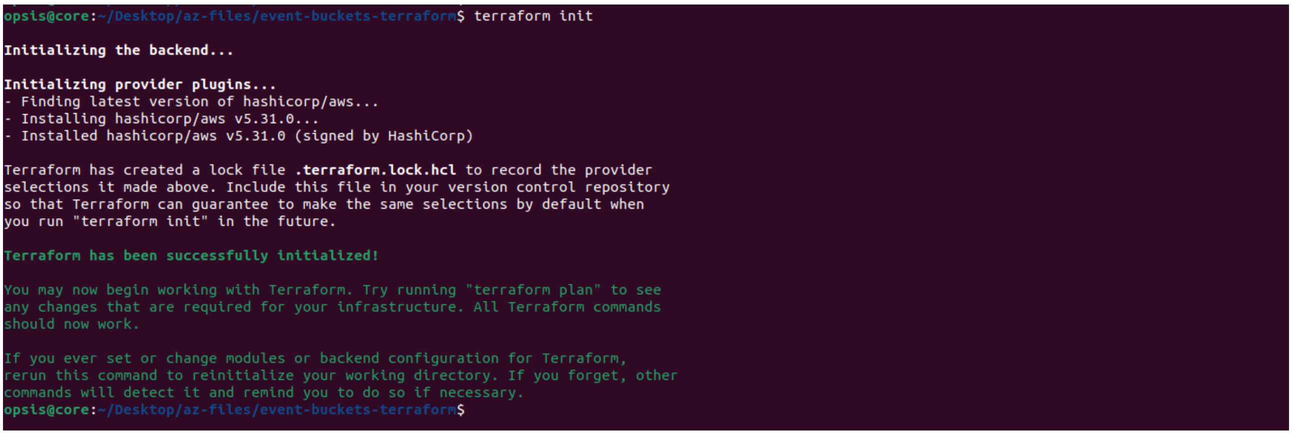
Appendix



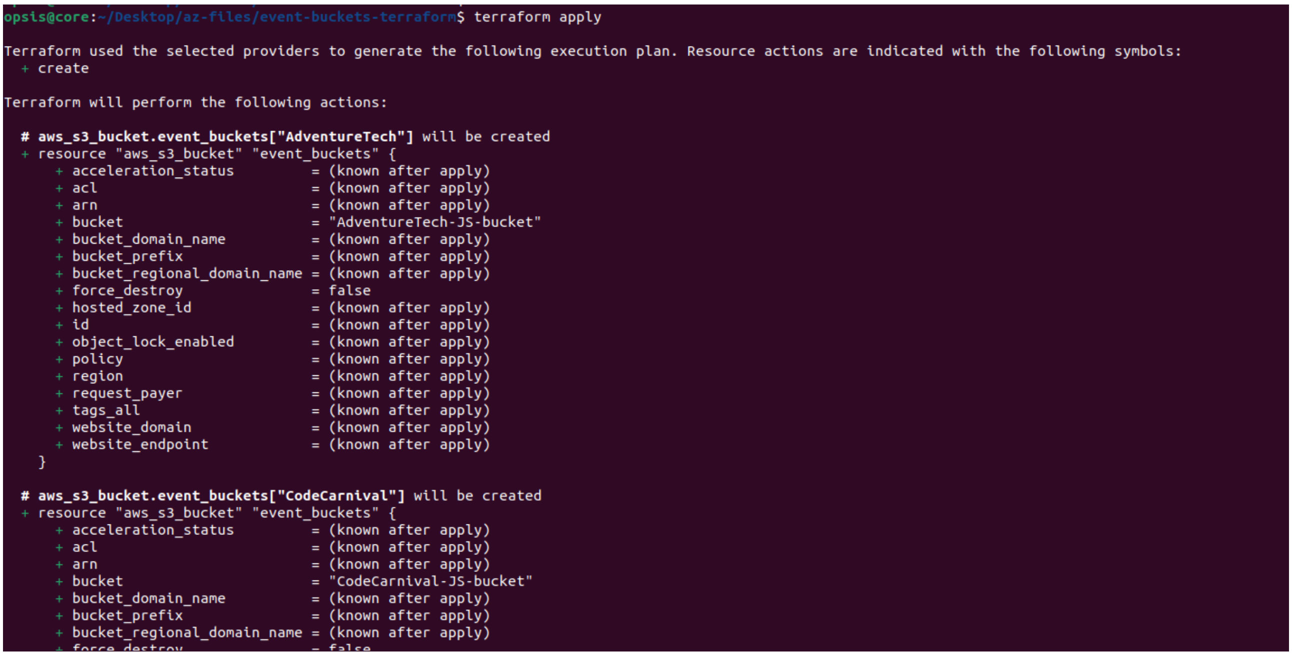
```
opsis@core:~/Desktop/az-files$ mkdir event-buckets-terraform
opsis@core:~/Desktop/az-files$ ls
event-buckets-terraform
opsis@core:~/Desktop/az-files$ cd event-buckets-terraform
opsis@core:~/Desktop/az-files/event-buckets-terraform$
opsis@core:~/Desktop/az-files/event-buckets-terraform$ nano main.tf
```



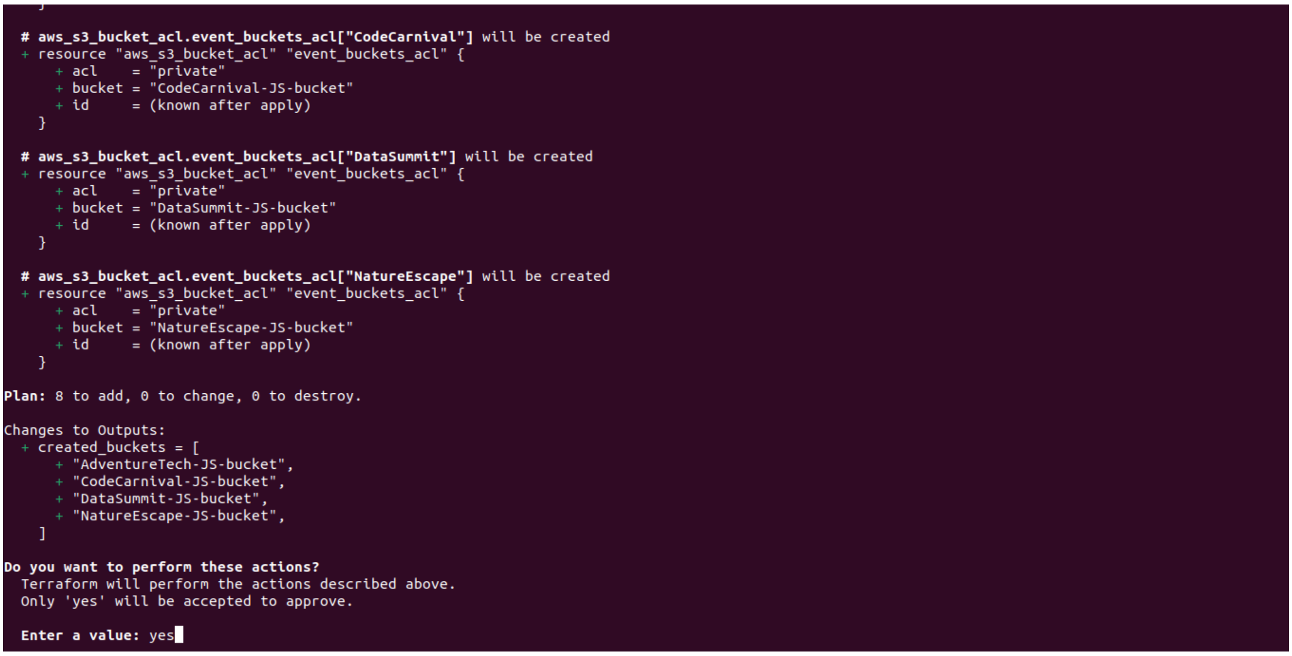
```
  GNU nano 6.2                                main.tf
# main.tf

# Define variables
variable "event_themes" {
  type    = list(string)
  default = ["adventuretech", "natureescape", "datasummit", "codecarnival"]
}

# Create S3 buckets
resource "aws_s3_bucket" "event_buckets" {
  for_each = { for theme in var.event_themes : theme => theme }

  bucket = format("%s-capd-bucket", each.value)


}

# Output the names of created buckets
output "created_buckets" {
  value = [for bucket in aws_s3_bucket.event_buckets : bucket.bucket]
}
```



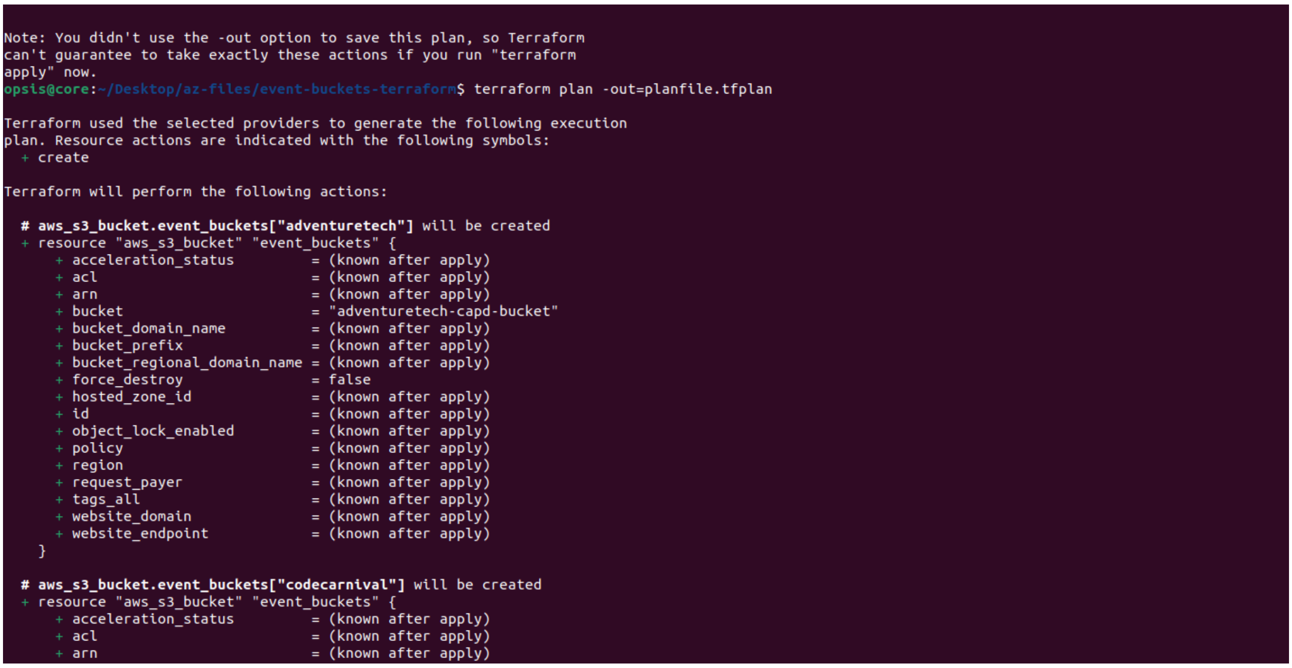
```
opsis@core:~/Desktop/az-files/event-buckets-terraform$ terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.31.0...
- Installed hashicorp/aws v5.31.0 (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.
opsis@core:~/Desktop/az-files/event-buckets-terraform$
```



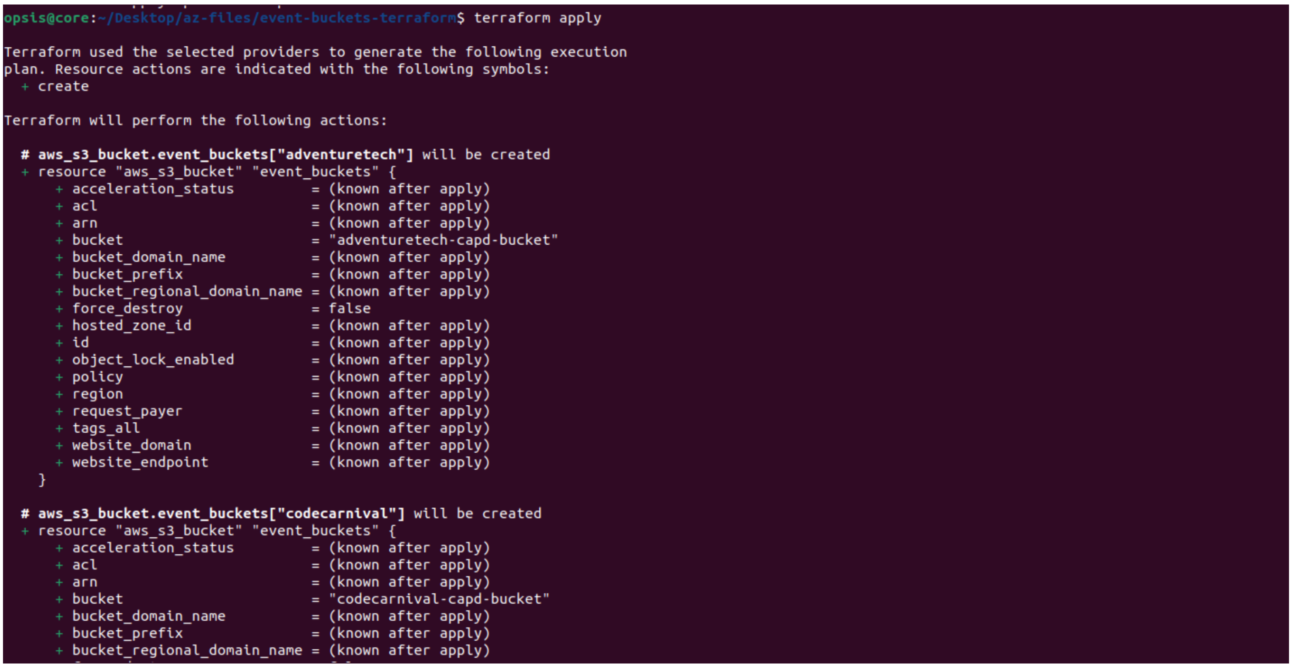
```
opsis@core:~/Desktop/az-files/event-buckets-terraform$ terraform apply

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.event_buckets["AdventureTech"] will be created
  + resource "aws_s3_bucket" "event_buckets" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "AdventureTech-JS-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_all                    = (known after apply)
      + website_domain              = (known after apply)
      + website_endpoint            = (known after apply)
    }

  # aws_s3_bucket.event_buckets["CodeCarnival"] will be created
  + resource "aws_s3_bucket" "event_buckets" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "CodeCarnival-JS-bucket"
      + bucket_domain_name          = (known after apply)
      + bucket_prefix               = (known after apply)
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
```



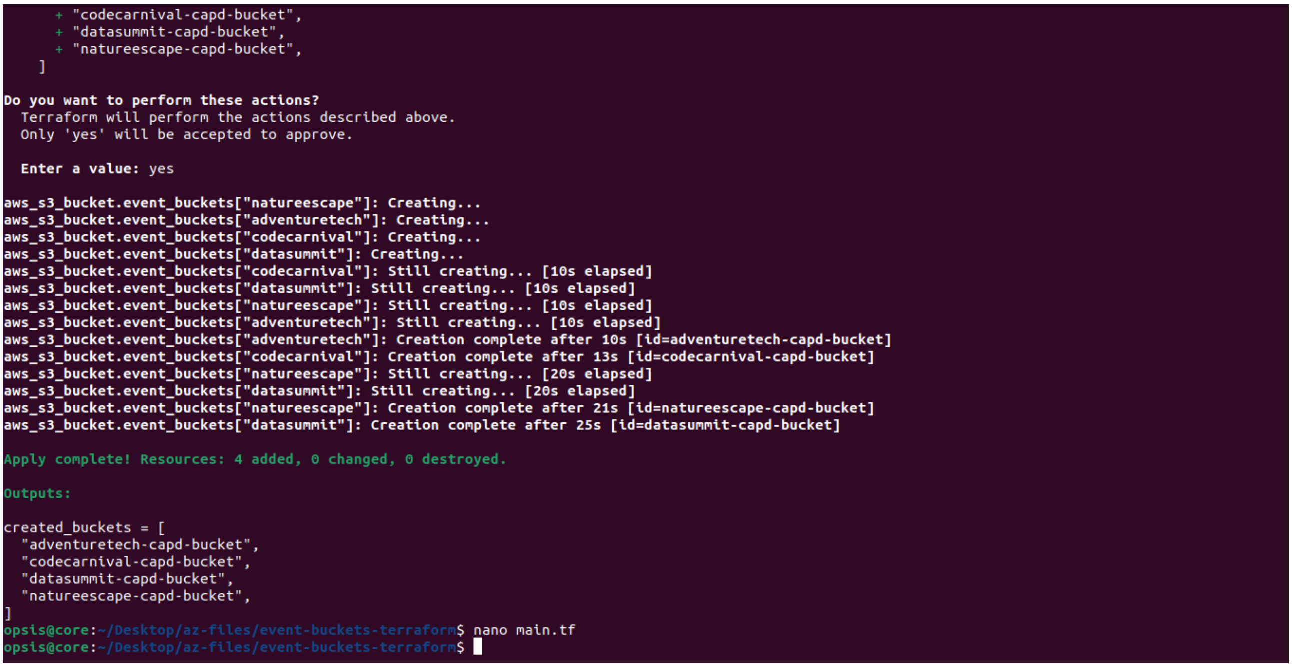
```
    }

  # aws_s3_bucket_acl.event_buckets_acl["CodeCarnival"] will be created
  + resource "aws_s3_bucket_acl" "event_buckets_acl" {
      + acl    = "private"
      + bucket = "CodeCarnival-JS-bucket"
      + id     = (known after apply)
    }

  # aws_s3_bucket_acl.event_buckets_acl["DataSummit"] will be created
  + resource "aws_s3_bucket_acl" "event_buckets_acl" {
      + acl    = "private"
      + bucket = "DataSummit-JS-bucket"
      + id     = (known after apply)
    }

  # aws_s3_bucket_acl.event_buckets_acl["NatureEscape"] will be created
  + resource "aws_s3_bucket_acl" "event_buckets_acl" {
      + acl    = "private"
      + bucket = "NatureEscape-JS-bucket"
      + id     = (known after apply)
    }

Plan: 8 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + created_buckets = [
      + "AdventureTech-JS-bucket",
      + "CodeCarnival-JS-bucket",
      + "DataSummit-JS-bucket",
      + "NatureEscape-JS-bucket",
    ]

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
```



```
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.
opsis@core:~/Desktop/az-files/event-buckets-terraform$ terraform plan -out=planfile.tfplan

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.event_buckets["adventuretech"] will be created
  + resource "aws_s3_bucket" "event_buckets" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "adventuretech-capd-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_all                    = (known after apply)
      + website_domain              = (known after apply)
      + website_endpoint            = (known after apply)
    }

  # aws_s3_bucket.event_buckets["codecarnival"] will be created
  + resource "aws_s3_bucket" "event_buckets" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
```



```
opsis@core:~/Desktop/az-files/event-buckets-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.event_buckets["adventuretech"] will be created
  + resource "aws_s3_bucket" "event_buckets" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "adventuretech-capd-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_all                    = (known after apply)
      + website_domain              = (known after apply)
      + website_endpoint            = (known after apply)
    }

  # aws_s3_bucket.event_buckets["codecarnival"] will be created
  + resource "aws_s3_bucket" "event_buckets" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "codecarnival-capd-bucket"
      + bucket_domain_name          = (known after apply)
      + bucket_prefix               = (known after apply)
      + bucket_regional_domain_name = (known after apply)
```



```
      + "codecarnival-capd-bucket",
      + "datasummit-capd-bucket",
      + "natureescape-capd-bucket",
    ]

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.event_buckets["natureescape"]: Creating...
aws_s3_bucket.event_buckets["adventuretech"]: Creating...
aws_s3_bucket.event_buckets["codecarnival"]: Creating...
aws_s3_bucket.event_buckets["datasummit"]: Creating...
aws_s3_bucket.event_buckets["codecarnival"]: Still creating... [10s elapsed]
aws_s3_bucket.event_buckets["datasummit"]: Still creating... [10s elapsed]
aws_s3_bucket.event_buckets["natureescape"]: Still creating... [10s elapsed]
aws_s3_bucket.event_buckets["adventuretech"]: Still creating... [10s elapsed]
aws_s3_bucket.event_buckets["adventuretech"]: Creation complete after 10s [id=adventuretech-capd-bucket]
aws_s3_bucket.event_buckets["codecarnival"]: Creation complete after 13s [id=codecarnival-capd-bucket]
aws_s3_bucket.event_buckets["natureescape"]: Still creating... [20s elapsed]
aws_s3_bucket.event_buckets["datasummit"]: Still creating... [20s elapsed]
aws_s3_bucket.event_buckets["natureescape"]: Creation complete after 21s [id=natureescape-capd-bucket]
aws_s3_bucket.event_buckets["datasummit"]: Creation complete after 25s [id=datasummit-capd-bucket]

Apply complete! Resources: 4 added, 0 changed, 0 destroyed.

Outputs:

created_buckets = [
  "adventuretech-capd-bucket",
  "codecarnival-capd-bucket",
  "datasummit-capd-bucket",
  "natureescape-capd-bucket",
]
opsis@core:~/Desktop/az-files/event-buckets-terraform$ nano main.tf
opsis@core:~/Desktop/az-files/event-buckets-terraform$ 
```