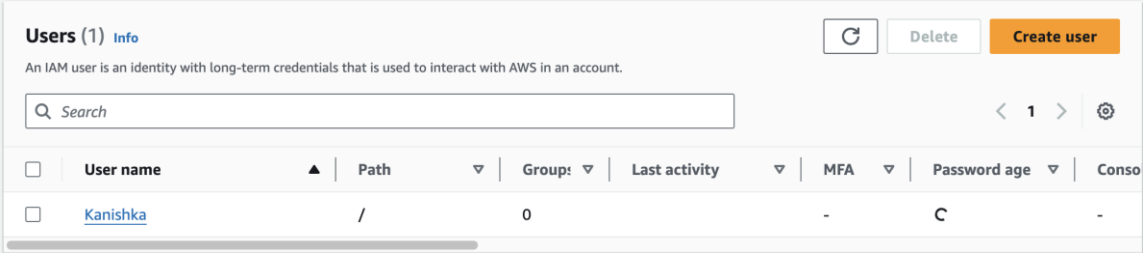
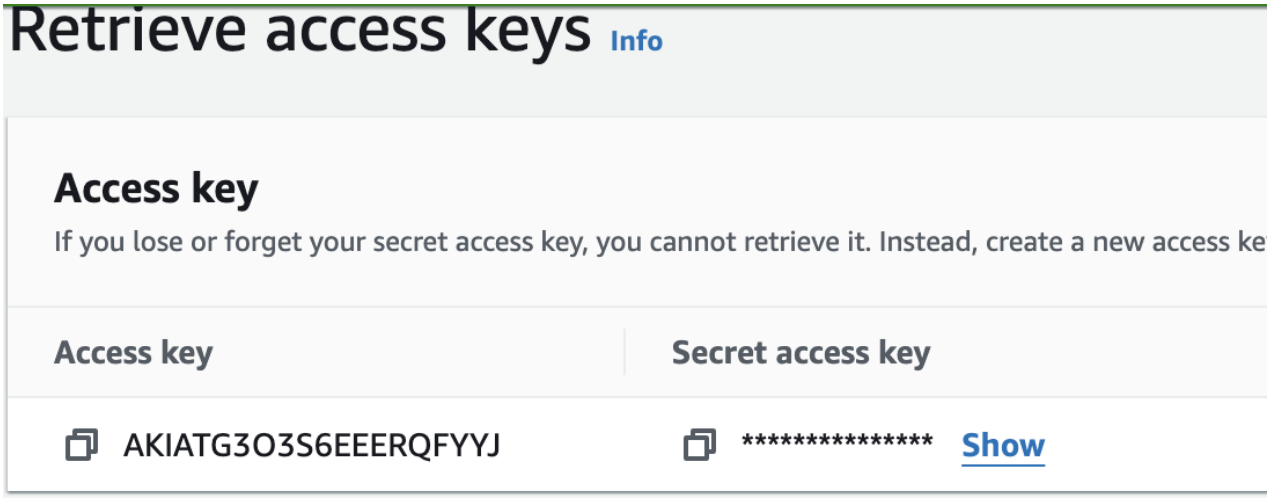
# SPCM Lab-2

Objective : Terraform AWS provider and IAM User Setting.

1. In order to provision resources on a cloud provider such as AWS, we need to create an IAM user and create the Access ID and Secret ID associated with that IAM user. (a) Create IAM role and assign privileages.




Create Access ID and Secret ID. Go to Users → your User → Security credentials → create Access keys → CLI




2. Now, create and Initialize a terraform repository on your local Machine. (a) Create a main.tf file and configure your cloud provider using the Access ID and Secret ID.

```
main.tf > terraform > required_providers > aws > source
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "5.31.0"
    }
  }
}
provider "aws" {
  region     = "ap-south-1"
  access_key = "AKIATG3O3S6ECMPNPHWX"
  secret_key = "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"
}

```

Initialize Terraform:

```
→  Terraform-SPCM-LAB  terraform init

Initializing the backend...

Initializing provider plugins...
- Finding hashicorp/aws versions matching "5.31.0"...
- 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.
→  Terraform-SPCM-LAB
```