Primero debemos de instalar Chocolately para poder instalar terraform y para esto debemos hacer que la ExecutionPolicy sea de tipo ByPass

```
PS C:\Users\FELIPE> [Enum]::GetNames([Net.SecurityProtocolType]) -contains '
Tls12'
True
PS C:\Users\FELIPE> [System.Net.ServicePointManager]::SecurityProtocol.HasFl
ag([Net.SecurityProtocolType]::Tls12)
True
PS C:\Users\FELIPE> [Net.ServicePointManager]::SecurityProtocol = [Net.Servi
cePointManager]::SecurityProtocol -bor [Net.SecurityProtocolType]::Tls12
PS C:\Users\FELIPE> [System.Net.ServicePointManager]::SecurityProtocol.HasFl
ag([Net.SecurityProtocolType]::Tls12)
True
PS C:\Users\FELIPE> Get-ExecutionPolicy
Bypass
PS C:\Users\FELIPE> Set-ExecutionPolicy Bypass -Scope Process -Force; [Syste
m.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManag
er]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).Down
loadString('https://community.chocolatey.org/install.ps1'))
```

Luego de descargar el Chocolately se instala el terraform

```
Ensuring chocolatey.nupkchoco install terraform
Chocolatey v2.3.0
Installing the following packages:
terraform
By installing, you accept licenses for the packages.
Downloading package from source 'https://community.chocolatey.org/api/v2/'
Progress: Downloading terraform 1.9.8... 100%

terraform v1.9.8 [Approved]
terraform package files install completed. Performing other installation steps.
The package terraform wants to run 'chocolateyInstall.ps1'.
Note: If you don't run this script, the installation will fail.
Note: To confirm automatically next time, use '-y' or consider:
choco feature enable -n allowGlobalConfirmation
Do you want to run the script?([Y]es/[A]ll - yes to all/[N]o/[P]rint): Y

Removing old terraform plugins
Downloading terraform 64 bit
  from 'https://releases.hashicorp.com/terraform/1.9.8/terraform_1.9.8_windows_amd64.zip'
Progress: 27% - Saving 7.18 MB of 26.1 MB
```

Miramos la versión del AWS que tenemos

```
PS C:\Windows\system32> aws -version

usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

  aws help
  aws <command> help
  aws <command> <subcommand> help

aws: error: the following arguments are required: command

PS C:\Windows\system32> aws --version
aws-cli/2.19.0 Python/3.12.6 Windows/11 exe/AMD64
PS C:\Windows\system32> aws configure
AWS Access Key ID [None]: ASIAWY2XLAVLR6EEQDEA
AWS Secret Access Key [None]: csj+R61WpyEO4uZ13U2G1seXyvupj9IYaBiQCELN
Default region name [None]: us-east-1
Default output format [None]: json
```

Creamos un archivo llamada main.tf, luego de codificarlo inicializamos el terraform, luego le damos a terraform apply para subir el terraform.

```
PS C:\Windows\system32\servicio> code .\main.tf
PS C:\Windows\system32\servicio> terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v4.67.0

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.
PS C:\Windows\system32\servicio> terraform apply

| Error: configuring Terraform AWS Provider: validating provider credentials: retrieving caller identity
 from STS: operation error STS: GetCallerIdentity, https response error StatusCode: 403, RequestID: fabb
1f4f-25ec-4fd2-b1c6-60197f156716, api error InvalidClientTokenId: The security token included in the req
uest is invalid.
|
|   with provider["registry.terraform.io/hashicorp/aws"],
|   on main.tf line 10, in provider "aws":
|   10: provider "aws" {
|
```

Este es el archivo del main.tf

C: > Users > FELIPE > Documents > Telemática > main.tf

```
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 4.16"
    }
  }

  required_version = ">= 1.2.0"
}

provider "aws" {
  region  = "us-east-1"
}

resource "aws_instance" "app_server" {
  ami           = "ami-0c02fb55956c7d316"
  instance_type = "t2.micro"

  tags = {
    Name = "ExampleAppServerInstance"
  }
}

```