Generating the infrastructure on AWS through Terraform

Introduction

The following Steps will allow you to create an infrastructure on AWS and access it. There is no need to manually install Terraform. It is executed via Docker using the `terraform.sh` script.

To follow these Steps, ensure you have correctly configured the `../.env` file in the root of this project.

There is a file `../.env.sample`. Copy it to `../.env` and edit the variables accordingly with your environment.

Steps

Step 1 → Initialize the project

```
$ ./terraform.sh init
```

Step 2 → Create the infrastructure

```
$ ./terraform.sh apply
```

Step 3 → Accessing the EC2 instance that run the apps in AWS

```
$ ./apps/ssh.sh
```

Inside the EC2 instance, you can control the Docker images in a similar way you do locally:

```
$ cd snowplow-demo

$ ./stats.sh # <- show the statistics for the docker containers
$ ./down.sh # <- stop the docker containers
$ ./up.sh # <- start the docker containers
```

Other commands

Check versions

```
$ ./terraform.sh --version
Terraform v1.10.0
on linux_amd64
+ provider registry.terraform.io/hashicorp/aws v5.79.0
+ provider registry.terraform.io/hashicorp/local v2.5.2
+ provider registry.terraform.io/hashicorp/tls v4.0.6
```

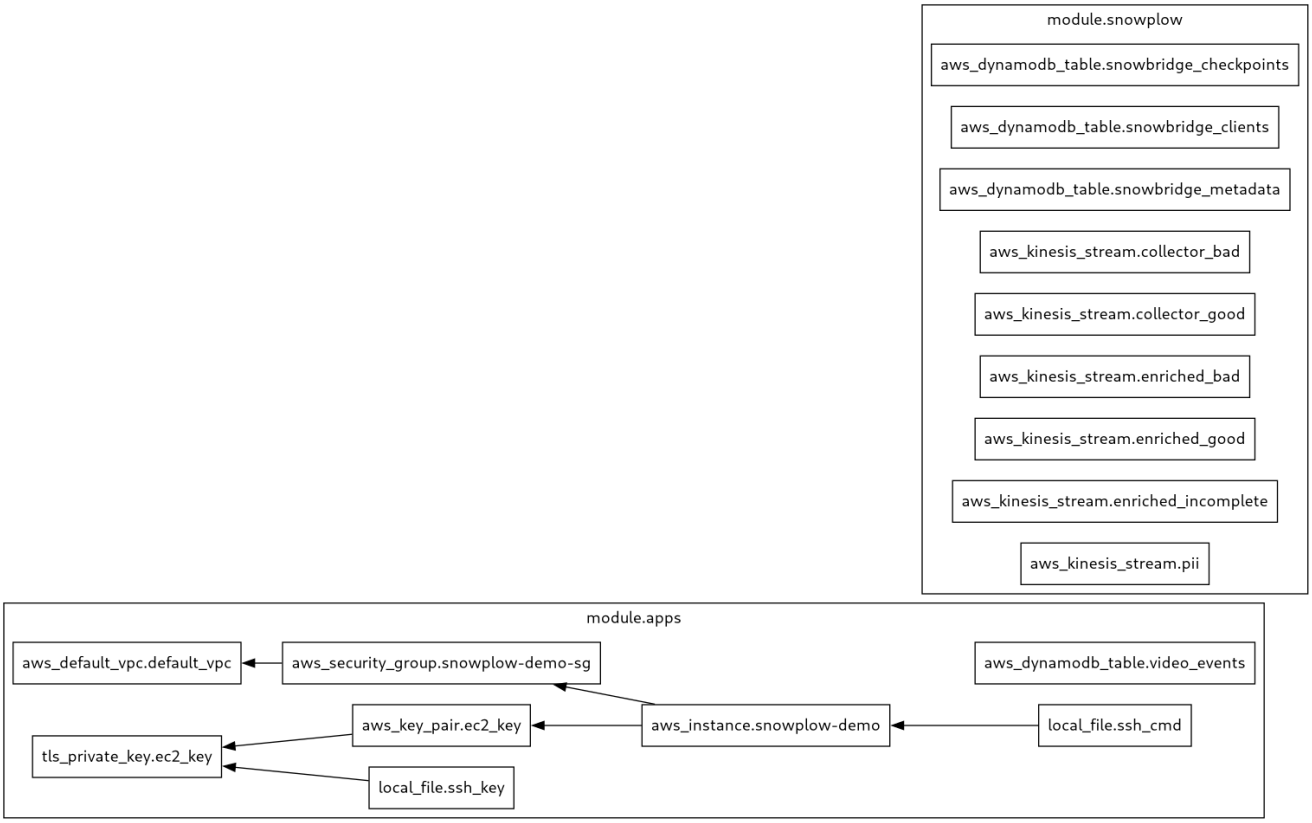
Check the Terraform plan

```
$ ./terraform.sh plan
```

Generate a PNG image for the Terraform modules in this project

```
$ ./terraform.sh png
```

Current PNG image of the available modules:



Destroy the infrastructure

```
$ ./terraform.sh destroy
```