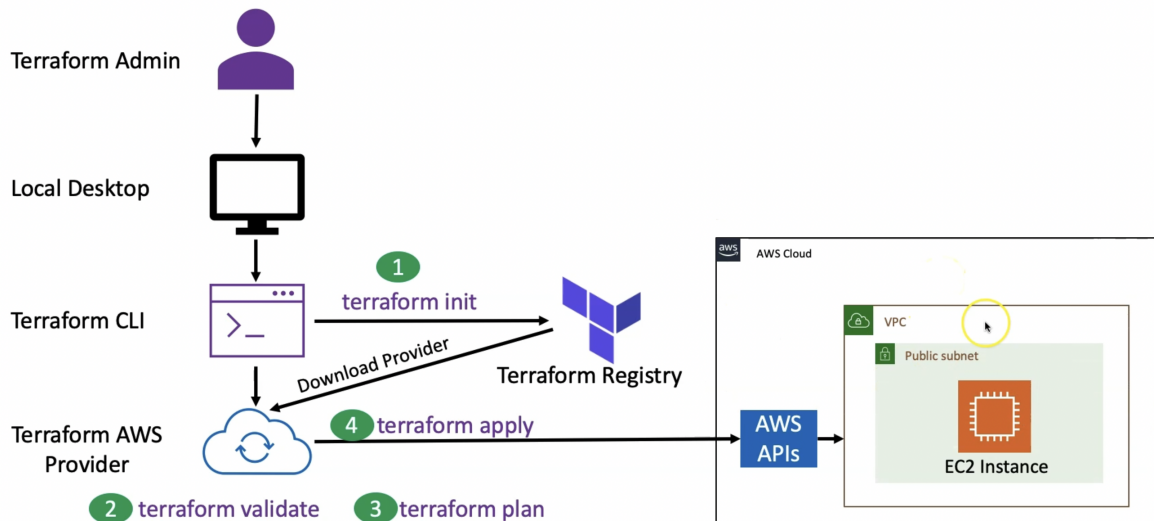


create ec2 instance 1

Terraform Providers



1, terraform-setting.tf 文件

表明使用terraform里的aws provider (也可以使用alibaba provider)

```
terraform {
  //required_version = ">= 0.14" # which means any version equal & above 0.14 like 0.15, 0.16 etc and < 1.xx
  required_version = ">= 1.2.2" # which means any version equal & above 1.2.2 like 1.2.3, 1.2.4 etc and < 1.3.xx
  required_providers {
    aws = {
      source = "hashicorp/aws"
      version = ">= 4.31.0"
    }
  }
}

# "aws" 与 required_providers里的aws 同名
# profile = "default" 可以不写, 它指的是cat $HOME/.aws/credentials 里设定的默认的access key
provider "aws" {
  region = "ap-northeast-1"
  profile = "default"
}
```

*provider的最新信息可以在这个网址查看 (点击user provider)

<https://registry.terraform.io/providers/hashicorp/aws/latest>

2, ec2-setting.tf 文件

ec2服务器的基本配置

```
# Resource: EC2 Instance,
# "aws_instance" 相当于terraform提供的一个插件, "myec2vm" 是自定义名其用处以后会学
resource "aws_instance" "myec2vm" {
  ami = "ami-078296f82eb463377"
  instance_type = "t2.micro"
}
```

```

user_data = file("${path.module}/nginx-install.sh")
tags = {
    "Name" = "EC2 Demo"
}
}

```

3, nginx-install.sh 文件,

这个文件是用来被ec2-setting.tf文件导入的

它的作用是下载nginx, 并对nginx网页做一些更改

```

#!/bin/bash
# Instance Identity Metadata Reference - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/instance-identity-documents.html
sudo yum update -y
sudo amazon-linux-extras install nginx1
sudo echo '<h1>Welcome to StackSimplify - APP-1</h1>' | sudo tee /usr/share/nginx/html/index.html
sudo mkdir /usr/share/nginx/html/app1
sudo echo '<!DOCTYPE html> <html> <body style="background-color:rgb(250, 210, 210);"> <h1>Welcome to Stack Simplify - APP-1</h1> <p>Te
sudo curl http://169.254.169.254/latest/dynamic/instance-identity/document -o /usr/share/nginx/html/app1/metadata.html
sudo systemctl enable nginx
sudo systemctl start nginx

```

4, 把这三个文件放入一个文件夹里 create-ec2-instance

5, create-ec2-instance文件夹里执行命令

```

terraform init // 自动根据terraform-setting.tf 生成 [.terraform]跟[.terraform.lock.hcl] 两个文件

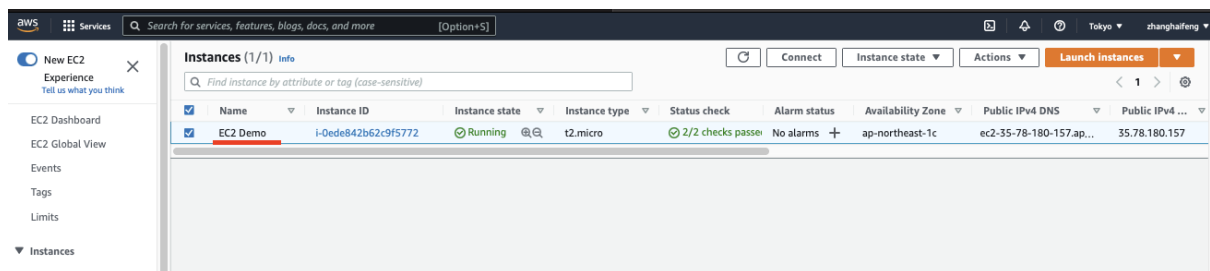
terraform validate // 验证配置是否正确

terraform plan // 列出将要做的事, 配置是什么

terraform apply -auto-approve // 执行, 这条命令会生成一个terraform.tfstate 文件, 这个文件很重要。不要乱改乱动

```

6, 打开aws管理画面, 可以看到ec2实例被建立



7, security group打开 80 端口

8, 浏览器输入以下这几个网址检查是否设置成功

```

# Access index.html
http://<PUBLIC-IP>/index.html
http://<PUBLIC-IP>/app1/index.html
http://<PUBLIC-IP>/app1/metadata.html

```

9, 终结ec2实例,aws管理画面的实例会被删除

```
# Terraform Destroy
terraform plan -destroy          # You can view destroy plan using this command
terraform destroy -auto-approve

# Clean-Up Files
rm -rf .terraform*
rm -rf terraform.tfstate*
```

*代码github link

<https://github.com/no744634936/terraform/tree/main/1-create-ec2-instance>