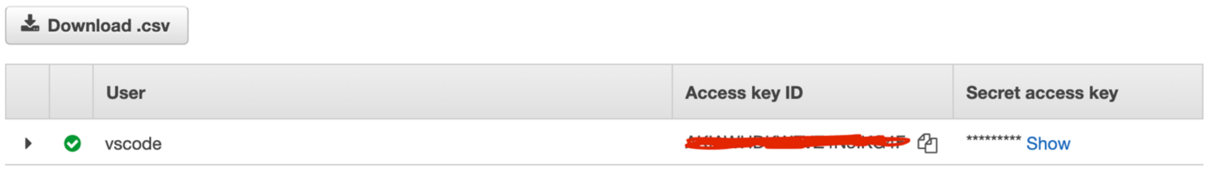
Darren Freeman
June 2022

Creating a Virtual Environment using Terraform

Step 1: We are going to need to access our AWS account to use the IAM management Console.

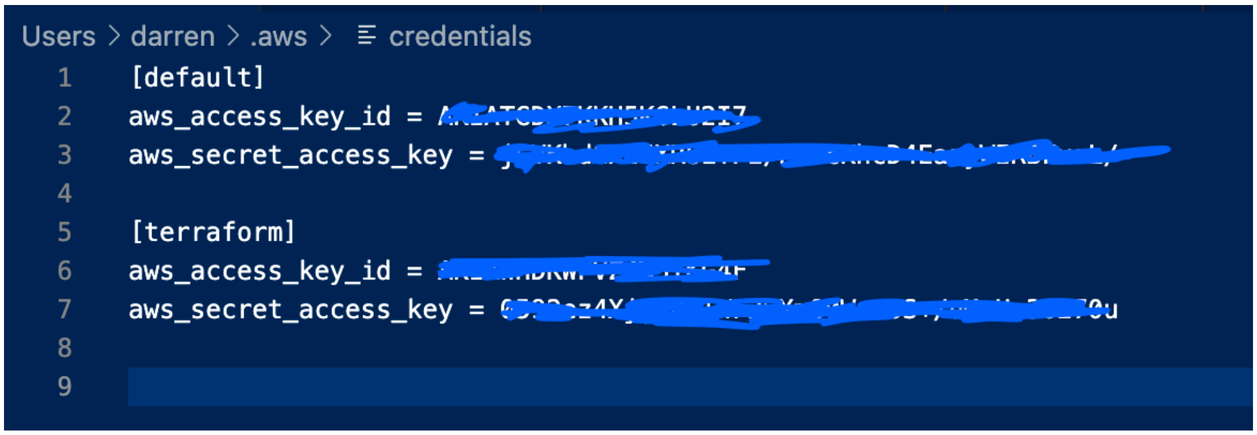
Next, we are going to create a user (this can be named anything), finally we must give the user admin access and the ability to make programmatic changes



User	Access key ID	Secret access key
vscode	[redacted]	********* Show

Step 2: Now it is time to open VSCode and from the extensions tab on the left sidebar, install the AWS toolkit. After installing you should restart VSCode. Upon restart you will be asked to set up tor credentials the next time you click on the AWS icon.

This part is simple, all you need to do is enter the access ID and secret key in the designated locations.



```
[default]
aws_access_key_id = [redacted]
aws_secret_access_key = [redacted]

[terraform]
aws_access_key_id = [redacted]
aws_secret_access_key = [redacted]

```

Next create a folder called terraform, this will be where we create all our terraform files. Open a new terminal and cd into the terraform folder.

Create a new file called ***providers.tf***

In this file we are going to put the following:

```
terraform {
   required_providers {
      aws = {
         source = "hashicorp/aws"
      }
   }
}

provider "aws"  {
   region                    = "us-east-1"
   shared_credentials_file     = "~/.aws/credentials"
   profile                   = "terraform"
}
```

This code lets terraform know that we will be using AWS for our cloud environment. If we were using Azure or Google Cloud Platform, we would have to specify that all the same.

Now, we will be creating another file named ***main.tf***
This new file is where we will be using code to spin up our resources

First, we will spin up a VPC with the following code:

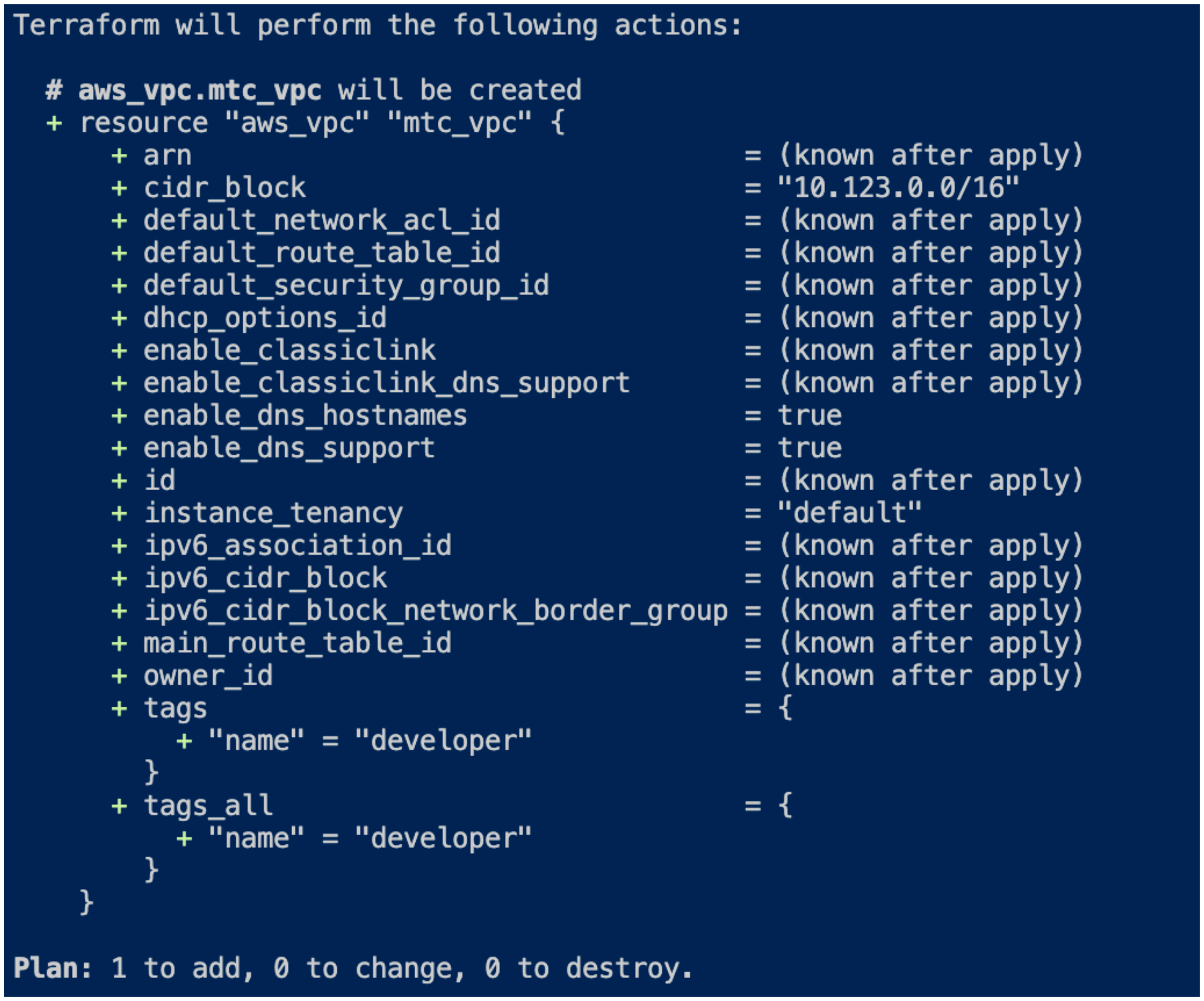
```
resource "aws_vpc" "mtc_vpc" {
   cidr_block = "10.123.0.0/16"
   enable_dns_hostnames = true
   enable_dns_support = true

   tags = {
      name = "developer"
   }
}
```

Tags are not necessary but they can be useful for labeling and differentiating your resources.

Now we can initialize this by running the command ***terraform init***
This will initialize terraform and create a file called ***.terraform.lock.hcl*** . All this file does is lock the version of terraform that you are using.

After we have initialized terraform now we can run *terraform plan* this shows us all the resources will be spun up and all the changes that are slated to be made.

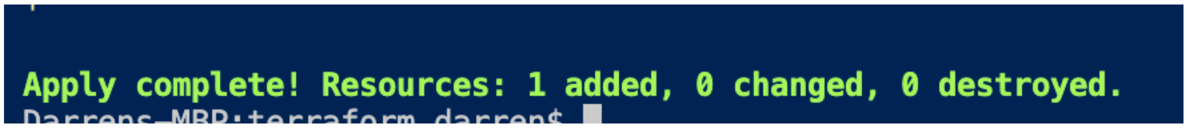


```
Terraform will perform the following actions:

  # aws_vpc.mtc_vpc will be created
  + resource "aws_vpc" "mtc_vpc" {
      + arn                                  = (known after apply)
      + cidr_block                           = "10.123.0.0/16"
      + default_network_acl_id               = (known after apply)
      + default_route_table_id               = (known after apply)
      + default_security_group_id            = (known after apply)
      + dhcp_options_id                      = (known after apply)
      + enable_classiclink                   = (known after apply)
      + enable_classiclink_dns_support       = (known after apply)
      + enable_dns_hostnames                 = true
      + enable_dns_support                   = true
      + id                                   = (known after apply)
      + instance_tenancy                     = "default"
      + ipv6_association_id                  = (known after apply)
      + ipv6_cidr_block                      = (known after apply)
      + ipv6_cidr_block_network_border_group = (known after apply)
      + main_route_table_id                  = (known after apply)
      + owner_id                             = (known after apply)
      + tags                                 = {
          + "name" = "developer"
        }
      + tags_all                             = {
          + "name" = "developer"
        }
    }

Plan: 1 to add, 0 to change, 0 to destroy.
```

Once satisfied it is now time to run *terraform apply* to get this up and running
You will once again be shown what is slated to take place and you will be prompted to type ‘yes’ or ‘no’



```
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
Darrens-MBP:terraform darren$
```

We have now created an environment in terraform
SUCCESS!