**Creation of Resource Groups and Virtual Network with Subnets**

**Step1:** Create the provider Block by authenticating it using a Service Principal with a Client Secret.

#provider Block

provider "azurerm" {

  features {}

  client\_id       = "1f79e427-2ac4-4eb6-9ca0-f4dd4b3f31ee"

  client\_secret   = "KDQ8Q~H-\_St9118keMeU-ADFzsiY.3y.GMYEnbeS"

  tenant\_id       = "4a623a04-9917-4ee2-8f59-02586964c992"

  subscription\_id = "51c6d184-6756-4a9a-ade4-cd0f3d57cded"

}

Step2: Create resource Block with two or more Resources Groups (RG1 & RG2).

#ResourceBlock, resourcetype, referencename

resource "azurerm\_resource\_group" "TFRG-01" {

  name = "RG1"

  location = "eastus"

}

resource "azurerm\_resource\_group" "TFRG-02" {

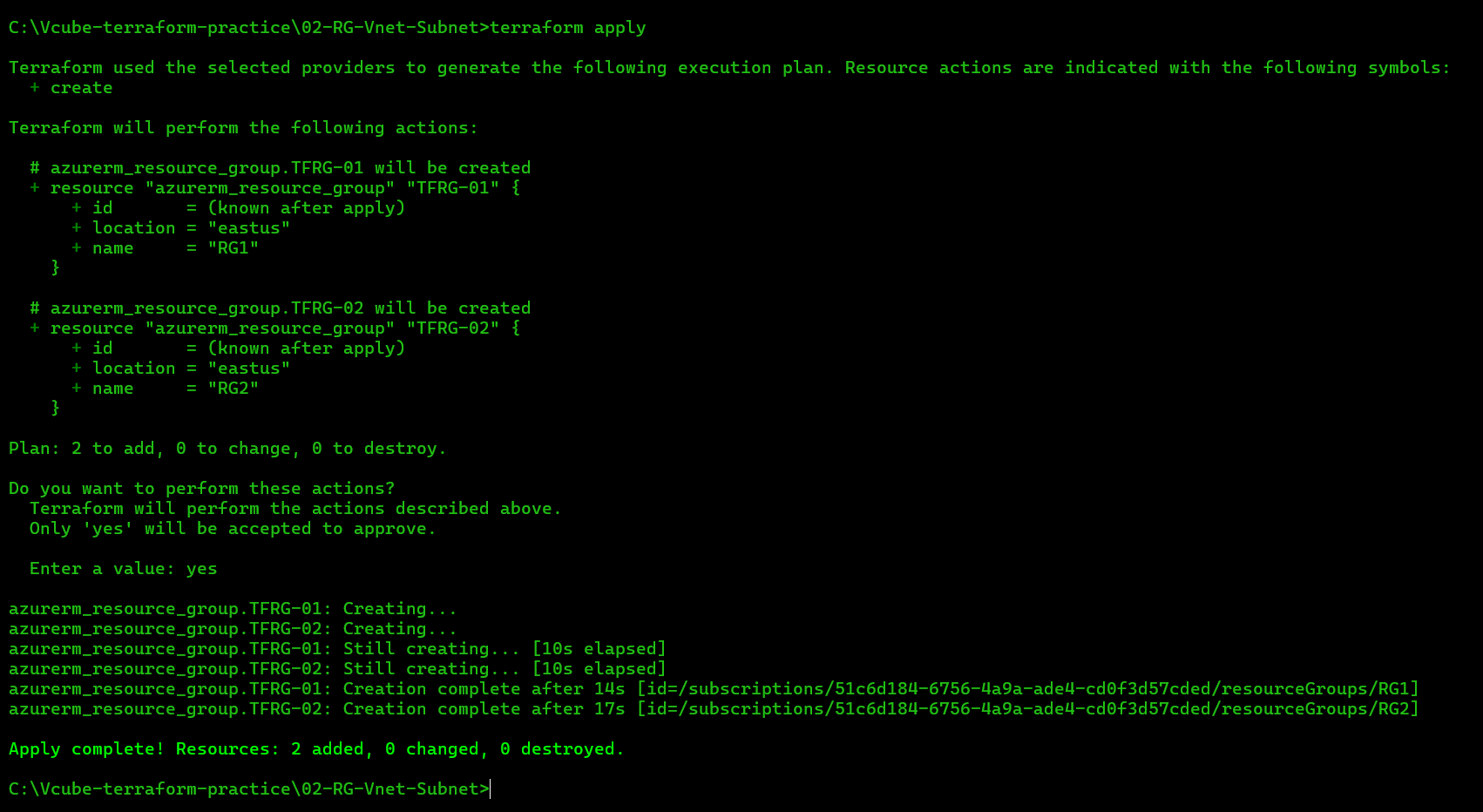
  name = "RG2"

  location = "eastus"

}

**Note:** While creating the similar resources (two resources groups or storage accounts) the Reference name and Name of the resource should be Unique or different as show above fig.

**Step3:** Now execute the entire terraform configuration file by using “terraform apply”.



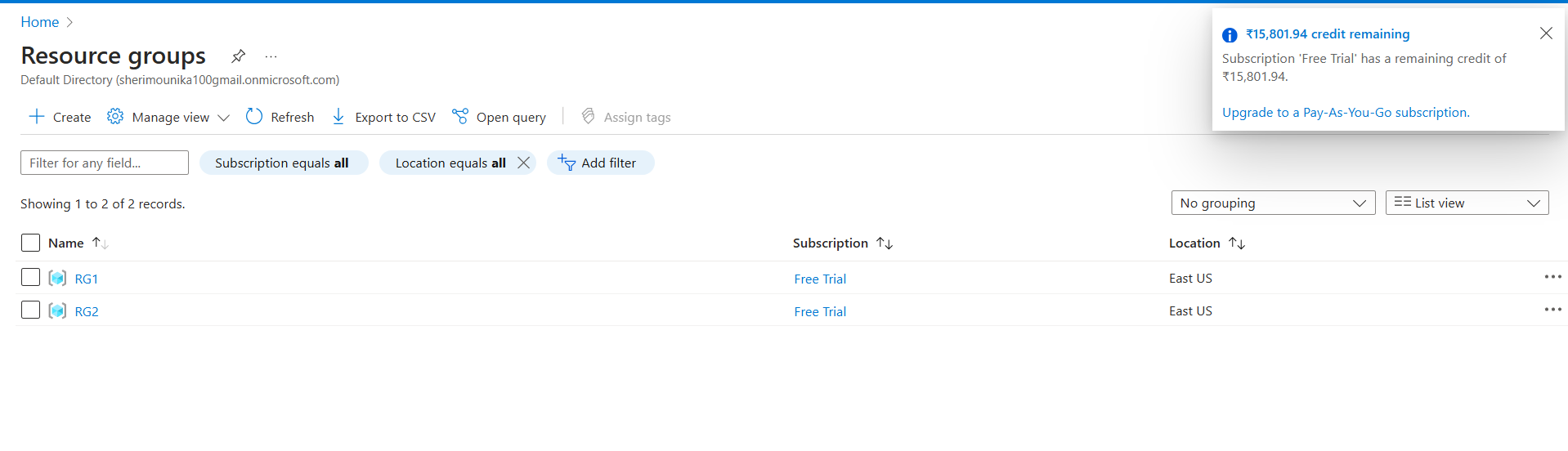
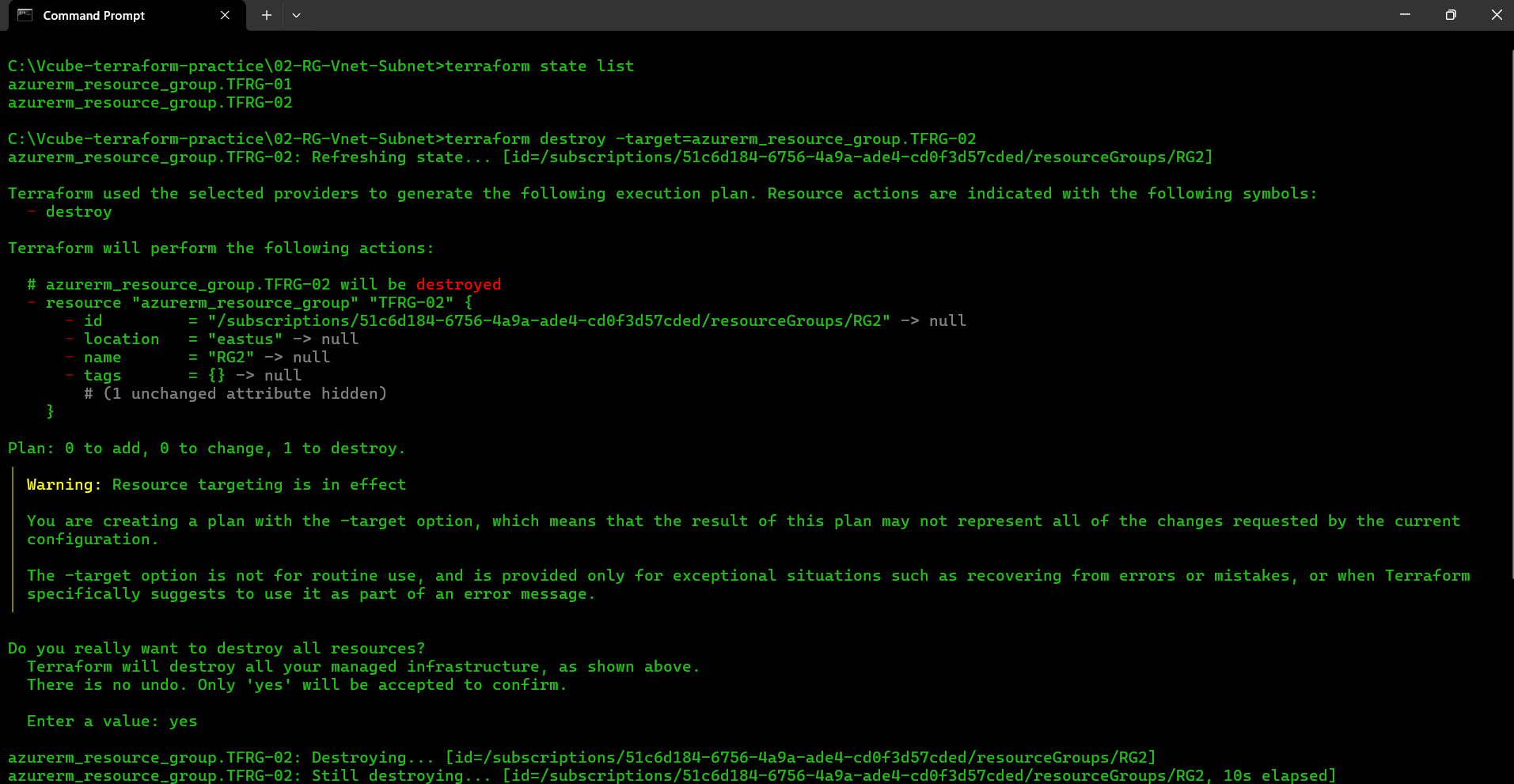


Fig: Two resource groups (RG1 & RG2) are created successfully.

**Step4:** Let’s delete one of the specific Resource Group using the command as

terraform destroy -target=azurerm\_resource\_group.TFRG-02



**Fig:** Resource group RG2 with Reference name **“TFRG-02**” is deleted successfully.

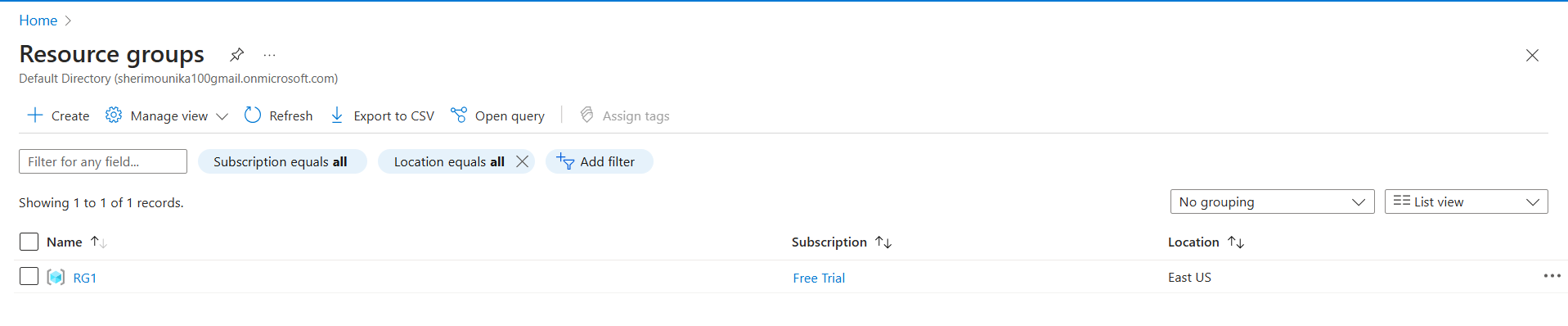


Fig: Portal View after deleting RG2.

**Virtual Network and Subnet Creation**

**Step5:** Create the virtual Network in the Resource group RG1.

#Resource Blocke to create Virtual Network

resource "azurerm\_virtual\_network" "TFVnet" {

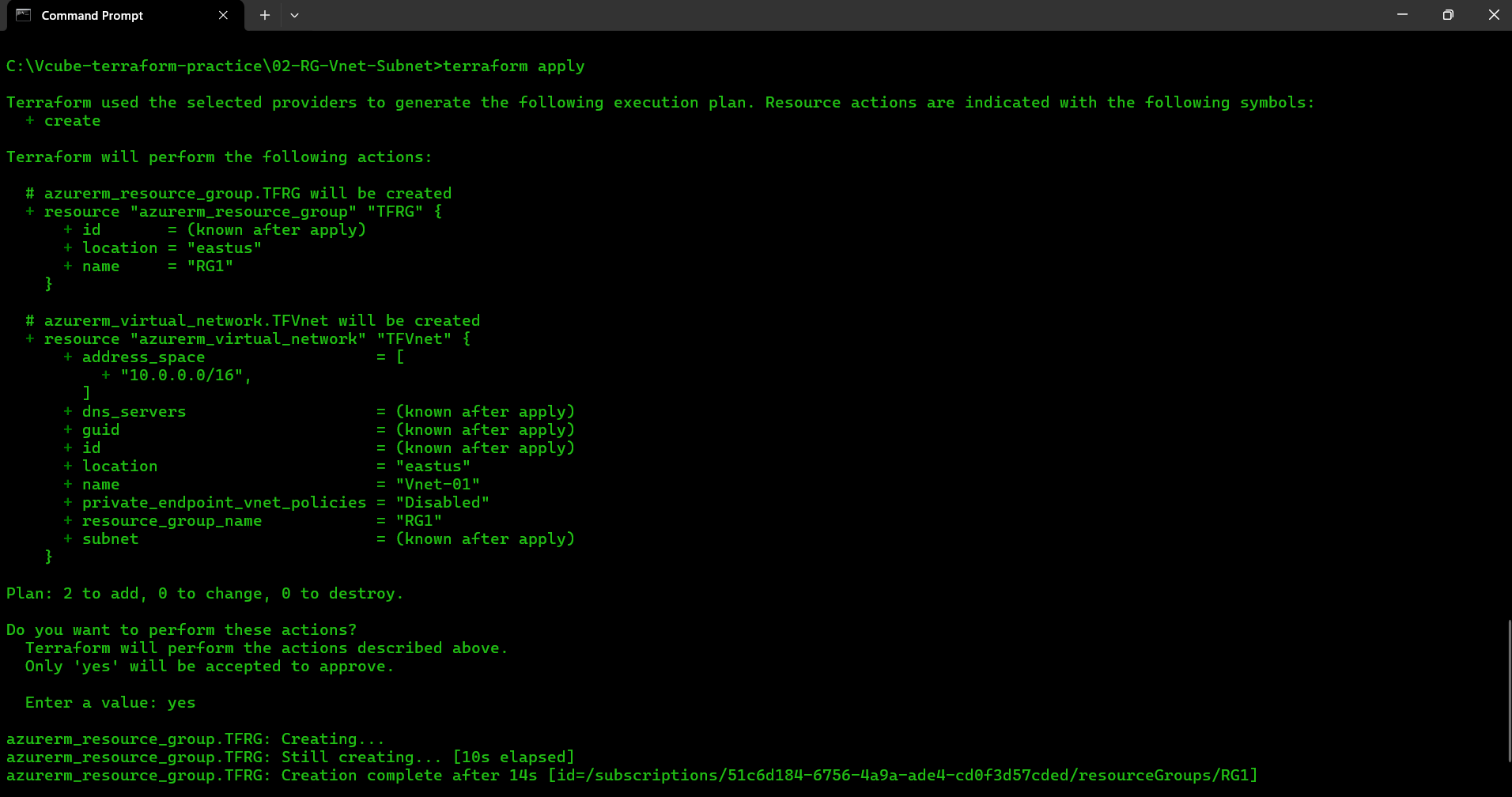
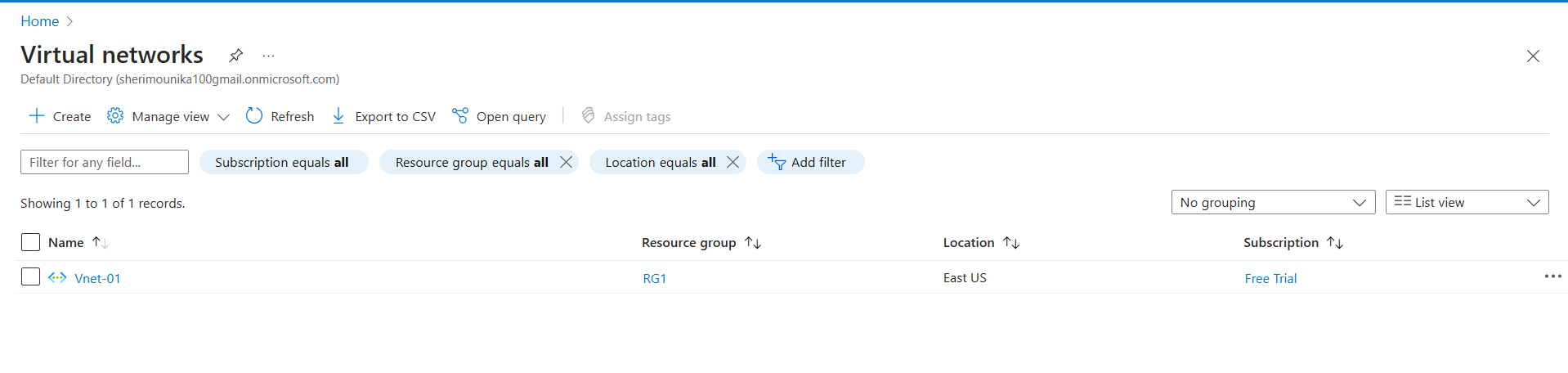
    name = "Vnet-01"

    location = azurerm\_resource\_group.TFRG.location

    resource\_group\_name = azurerm\_resource\_group.TFRG.name

    address\_space = ["10.0.0.0/16"]

}



**Fig:** Virtual Network is created.

**Step6:** Create the Subnets with in the Vnet-01.

#Resource Block to create the Subnet Within the Vnet

resource "azurerm\_subnet" "TFSubnet-01" {

  name = "Subnet-01"

  resource\_group\_name  = azurerm\_resource\_group.TFRG.name

  virtual\_network\_name = azurerm\_virtual\_network.TFVnet.name

  address\_prefixes     = ["10.0.1.0/24"]

}

resource "azurerm\_subnet" "TFSubnet-02" {

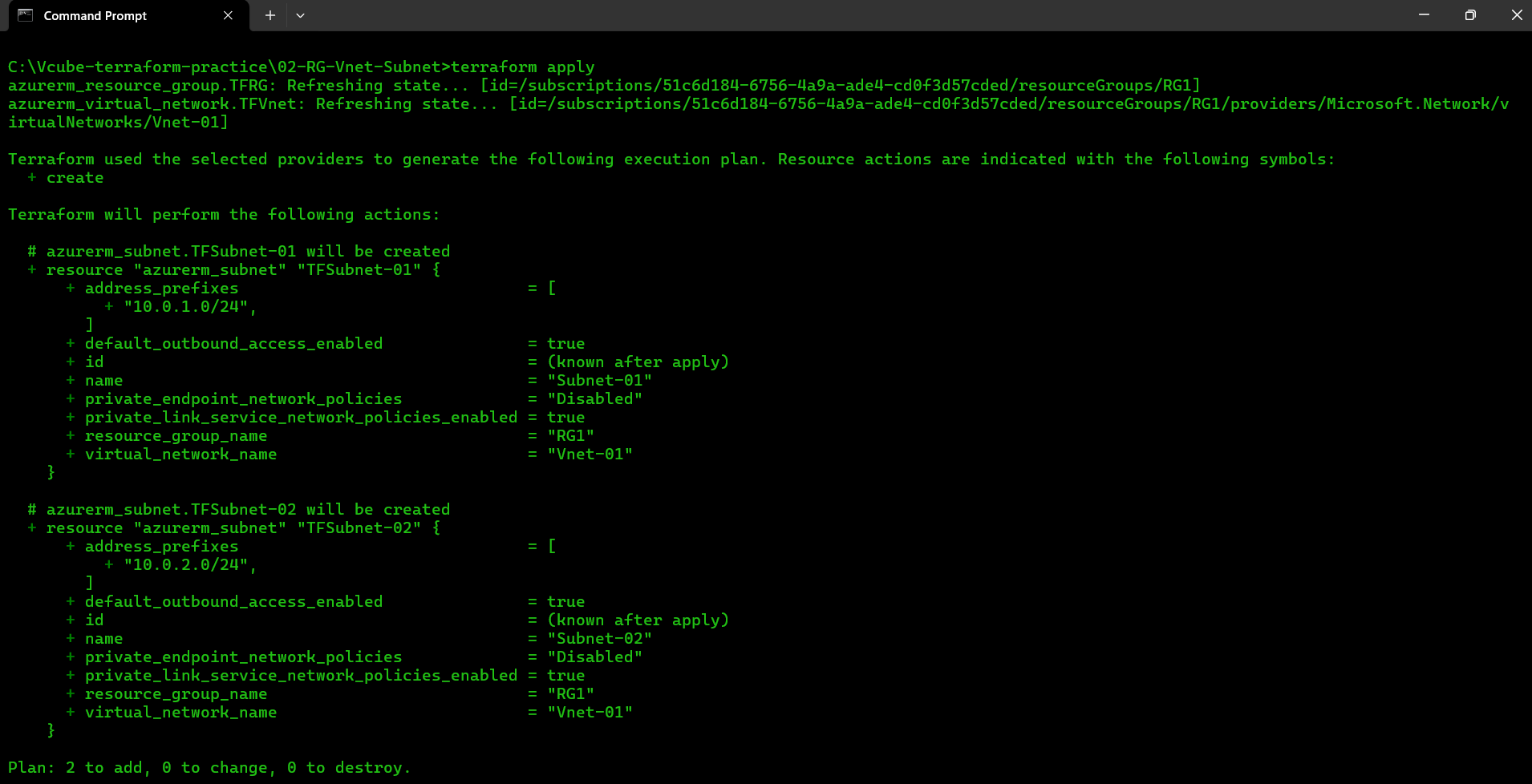
    name = "Subnet-02"

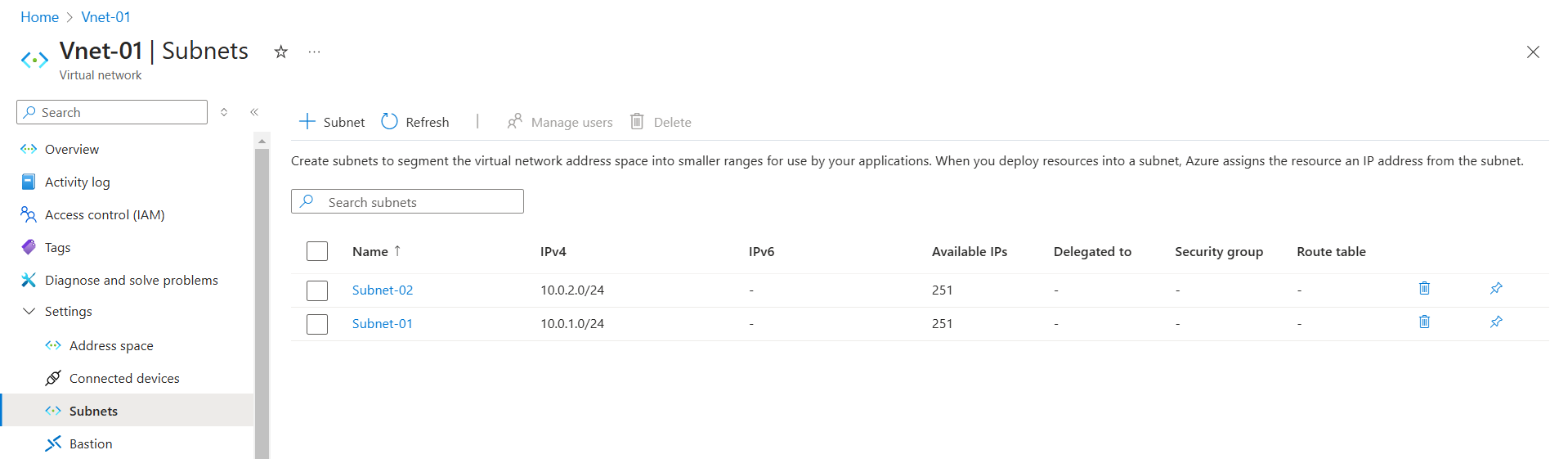
    resource\_group\_name = azurerm\_resource\_group.TFRG.name

    virtual\_network\_name = azurerm\_virtual\_network.TFVnet.name

    address\_prefixes = ["10.0.2.0/24"]

}





**Fig:** Two Subnets (Subnet-01 & Subnet-02) are created successfully.

**Let’s Create another Virtual Network “Vnet-02” and Perform the Peering**

**Step7:** Create another Virtual Network (Vnet-02).

#resource block to create another Virtual network (Vnet-02)

resource "azurerm\_virtual\_network" "TFVnet-02" {

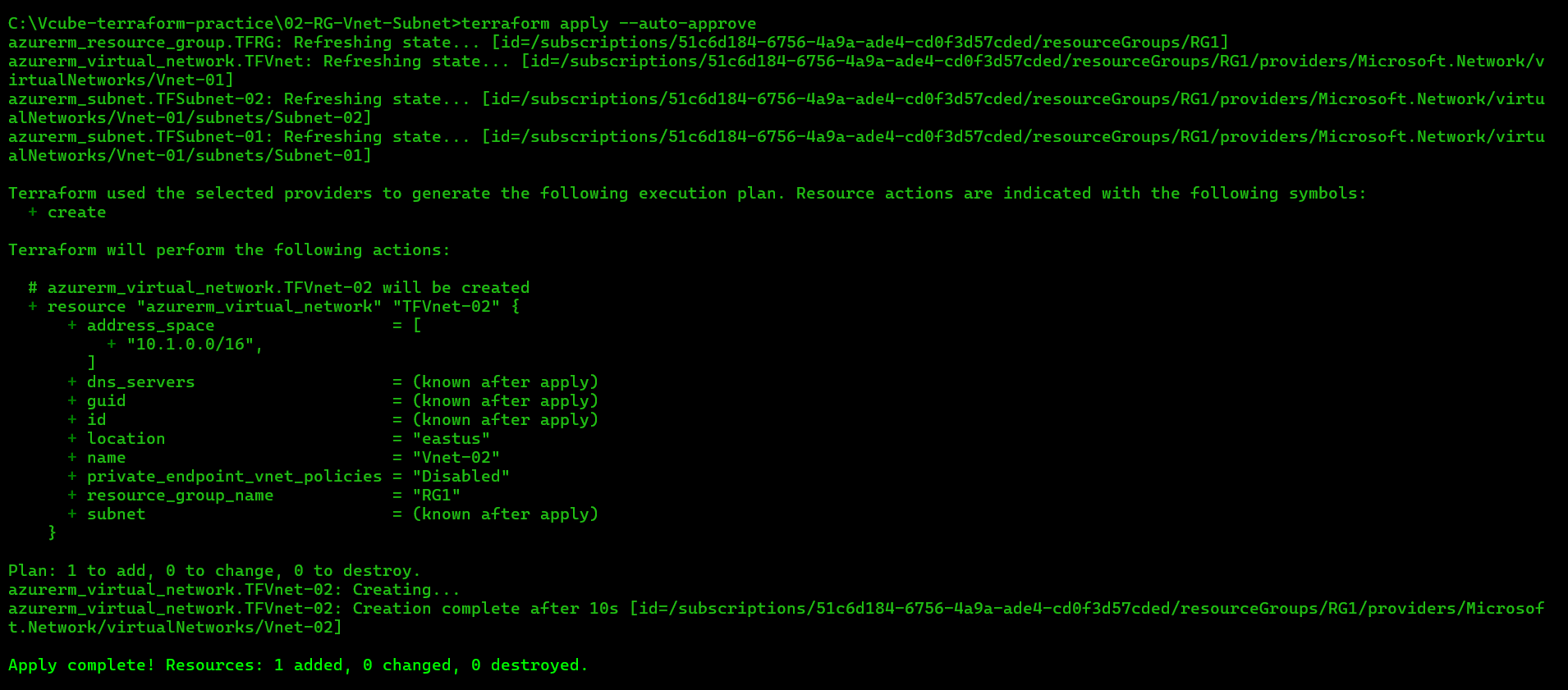
    name = "Vnet-02"

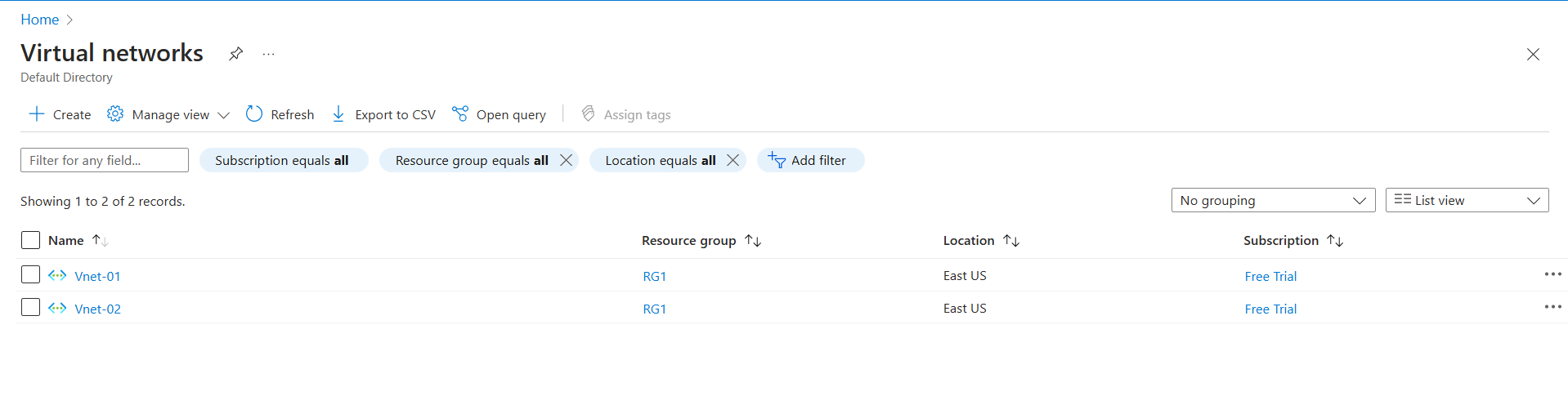
    location = azurerm\_resource\_group.TFRG.location

    resource\_group\_name = azurerm\_resource\_group.TFRG.name

    address\_space = ["10.1.0.0/16"]

}





**Fig:** another Virtual Network (Vnet-02) is created successfully.

**Note**: Whatever changes are done they can be uploaded to “**Terraform.tfstate”** file only after doing “**terraform apply**”.

**Step8:** Perform the Peering between Two Virtual networks (Vnet-01 & Vnet-02).

#resource block to crerate the Virtual Network peering

#peering form 1 to 2

resource "azurerm\_virtual\_network\_peering" "TFpeering1-2" {

  name                      = "peering-1-2"

  resource\_group\_name       = azurerm\_resource\_group.TFRG.name

  virtual\_network\_name      = azurerm\_virtual\_network.TFVnet-01.name

  remote\_virtual\_network\_id = azurerm\_virtual\_network.TFVnet-02.id

}

#peering from 2 to 1

resource "azurerm\_virtual\_network\_peering" "example-2" {

  name                      = "peer-2-1"

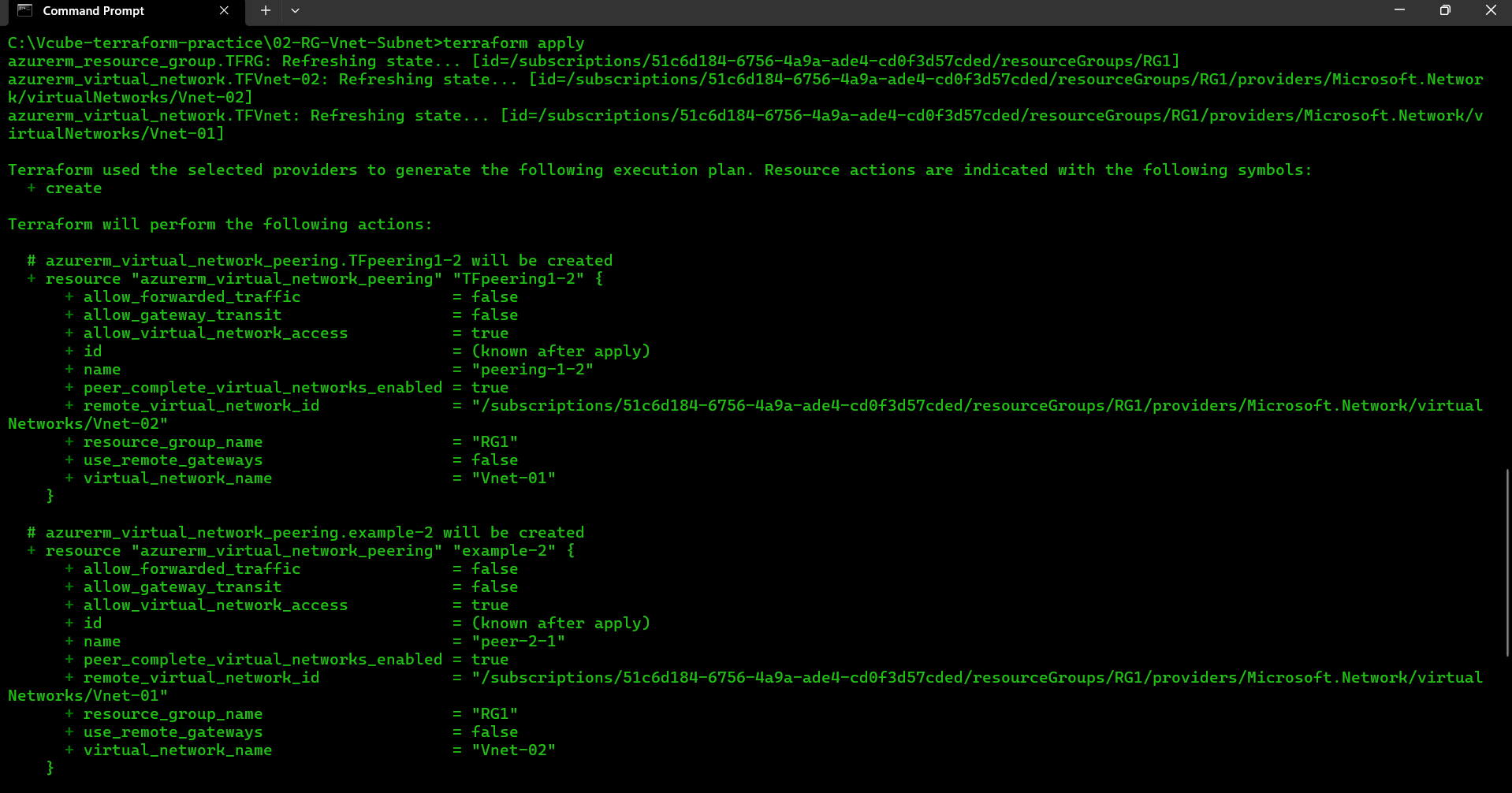
  resource\_group\_name       = azurerm\_resource\_group.TFRG.name

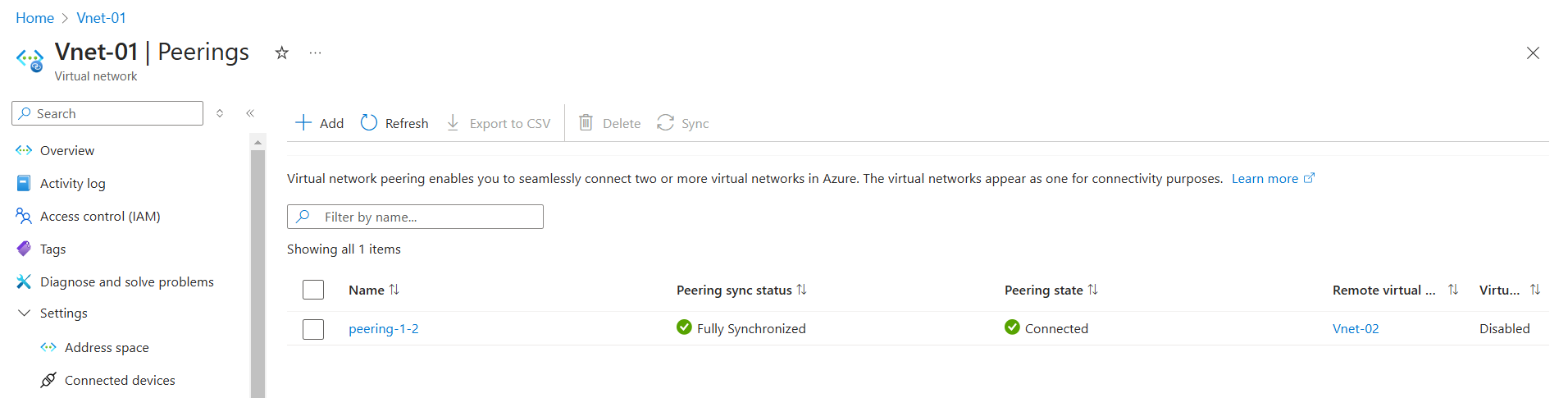
  virtual\_network\_name      = azurerm\_virtual\_network.TFVnet-02.name

  remote\_virtual\_network\_id = azurerm\_virtual\_network.TFVnet-01.id

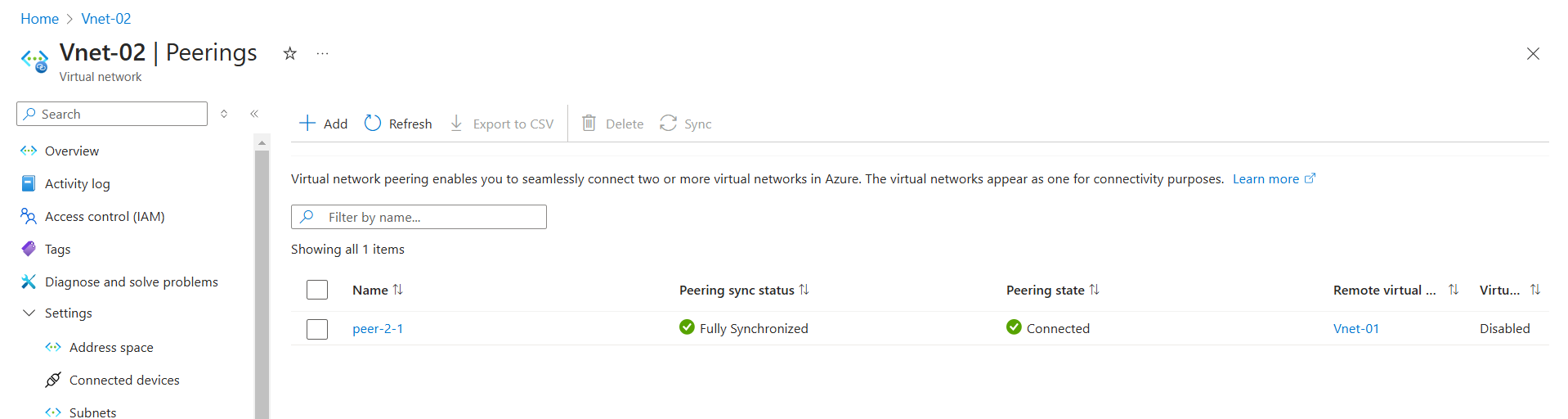
}

**Note:** In terraform coding **peering can be configured at two sides** as show above figure.





**Fig:** Peering at Vnet-01 side.



**Fig:** Peering at Vnet-02 side.

**Note:** If we delete or destroy all the resource by using command **“terraform destroy”** it will delete all resource from both **provider platform** as well as in **Terraform.tfstate file.**