2025

SZABIST

Name	Umer Amir
Registration Id	2112241
Section	BSCS 8th A
Course	Introduction to DevOps
Instructor	Tariq Khan Ghouri

Lab 02 (Modules 01, 02, and 03)

Create, modify, and destroy resources.

Tariq Khan Ghouri
IT INFRASTRUCTURE, CLOUD AND DEVOPS PROFESSIONAL

Instruc(ons:

1. Log in to Azure Portal with your credentials.
2. Paste all screenshots (highlighted in red) in a single Word document in the correct order.
3. Name the document as YourName-lab02.

Lab Objec(ve:

- Create, modify, and destroy resources.

Lab Descrip(on:

- Create a single Terraform script called lab02.U containing the following: ○ Provider and Terraform blocks. ○ Code to build the required infrastructure.
- Validate, deploy, expand, analyze, and destroy infrastructure.

Part 1: Prepare for the Lab

1. Open a Command Prompt or PowerShell window.
2. Create a directory called lab02 in your home directory.
3. Change into the lab02 directory.
4. Create an empty file called lab02.U.

Part 2: Codify the Following in a Single Terraform Script

Make sure to enclose values within double quotation marks.

5. Open lab02.U in a text editor (e.g., Notepad or Visual Studio Code) and define resource blocks as follows:

Define a resource group called lab02-rg using azurerm_resource_group.

Define a virtual network called lab02-vnet using azurerm_virtual_network.

Add a subnet to the virtual network called lab02-subnet1 using azurerm_subnet.

Define a network security group called lab02-nsg1 with an inbound allow TCP rule for port 22 called rule1 with priority 100 using azurerm_network_security_group.

A^ach the network security group lab02-nsg1 to lab02-subnet1 using azurerm_subnet_network_security_group_association.

lab02.tf

```
provider "azurerm" {
  features = {}
}

# Resource Group
resource "azurerm_resource_group" "lab02_rg" {
  name     = "lab02-rg"
  location = "eastus"
}

# Virtual Network
resource "azurerm_virtual_network" "lab02_vnet" {
  name                = "lab02-vnet"
  address_space       = ["10.0.0.0/16"]
  location            = azurerm_resource_group.lab02_rg.location
  resource_group_name = azurerm_resource_group.lab02_rg.name
}

# Subnet to the Virtual Network
resource "azurerm_subnet" "lab02_subnet1" {
  name                 = "lab02-subnet1"
  resource_group_name  = azurerm_resource_group.lab02_rg.name
  virtual_network_name = azurerm_virtual_network.lab02_vnet.name
  address_prefixes     = ["10.0.1.0/24"]
}

# Network Security Group with an inbound rule
resource "azurerm_network_security_group" "lab02_nsg1" {
  name                = "lab02-nsg1"
  location            = azurerm_resource_group.lab02_rg.location
  resource_group_name = azurerm_resource_group.lab02_rg.name

  security_rule {
    name                       = "rule1"
    priority                   = 100
    direction                  = "Inbound"
    access                     = "Allow"
    protocol                   = "Tcp"
    source_port_range          = "*"
    destination_port_range     = "22"
    source_address_prefix      = "*"
    destination_address_prefix = "*"
  }
}

# Associate the NSG with the Subnet
resource "azurerm_subnet_network_security_group_association" "lab02_subnet_nsg_assoc" {
  subnet_id                 = azurerm_subnet.lab02_subnet1.id
  network_security_group_id = azurerm_network_security_group.lab02_nsg1.id
}

```

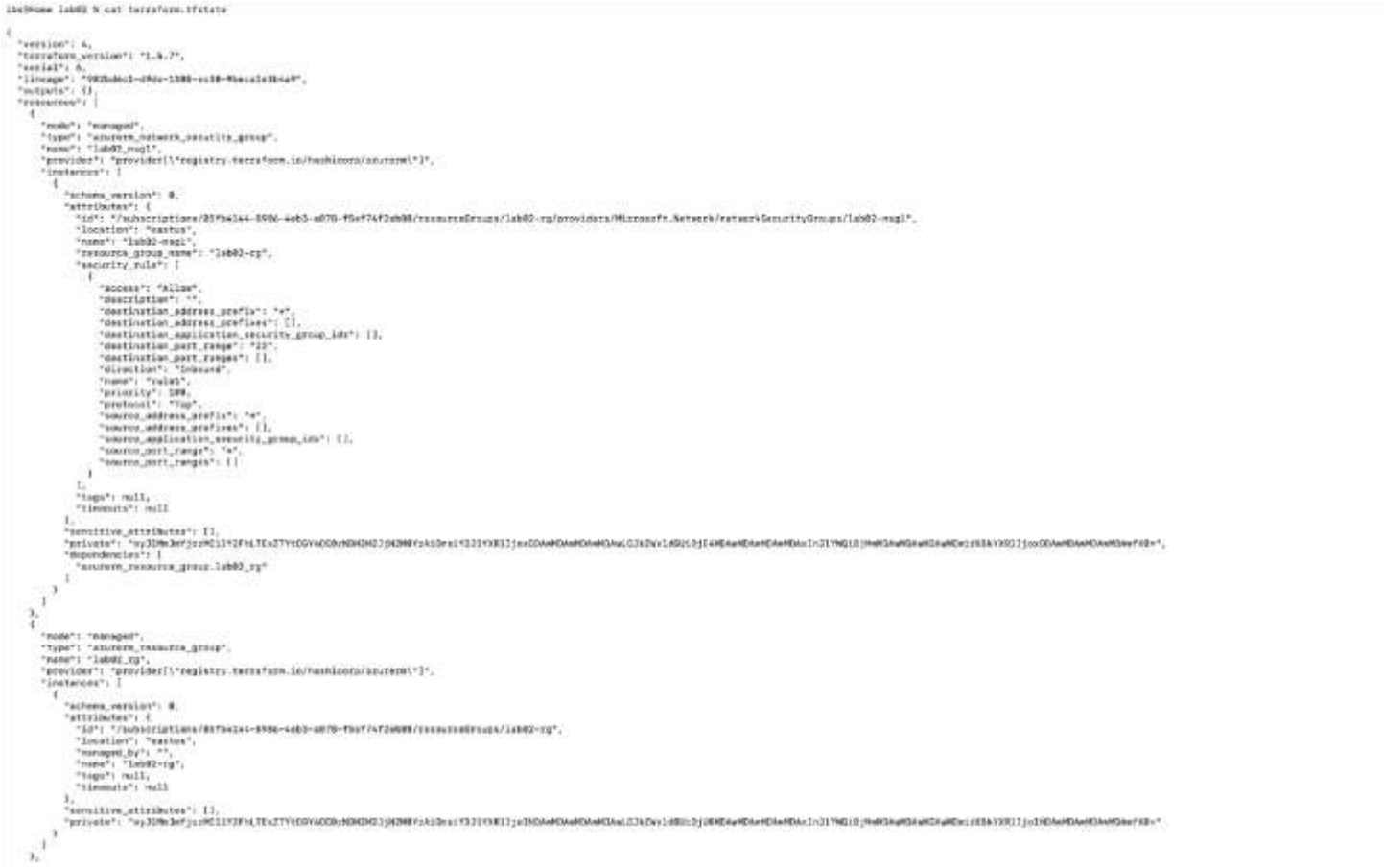
Part 3: Initialize Terraform

6. Initialize Terraform and download plug-ins as required:

```
terraform init
```

7. View the content of the terraform.Ustate file:

```
type terraform..state
```



Part 4: Validate Configuration

8. Validate the configuration to ensure there are no errors or typos in the file:

```
terraform validate
```

9. Fix any issues in the lab02.U file if reported (edit in your text editor).
10. Re-run the validation until no errors are reported.

Part 5: Run Simulation

11. Perform a dry run:

```
terraform plan
```

12. Review output and ensure all configuration is as per requirements. Observe the resources with +, -, or /+ signs.
13. Fix any issues in the lab02.U file if reported (edit in your text editor).
14. Redo the dry run until no errors are reported:

```
terraform plan
```

Part 6: Deploy Infrastructure

15. Deploy the infrastructure and monitor progress:

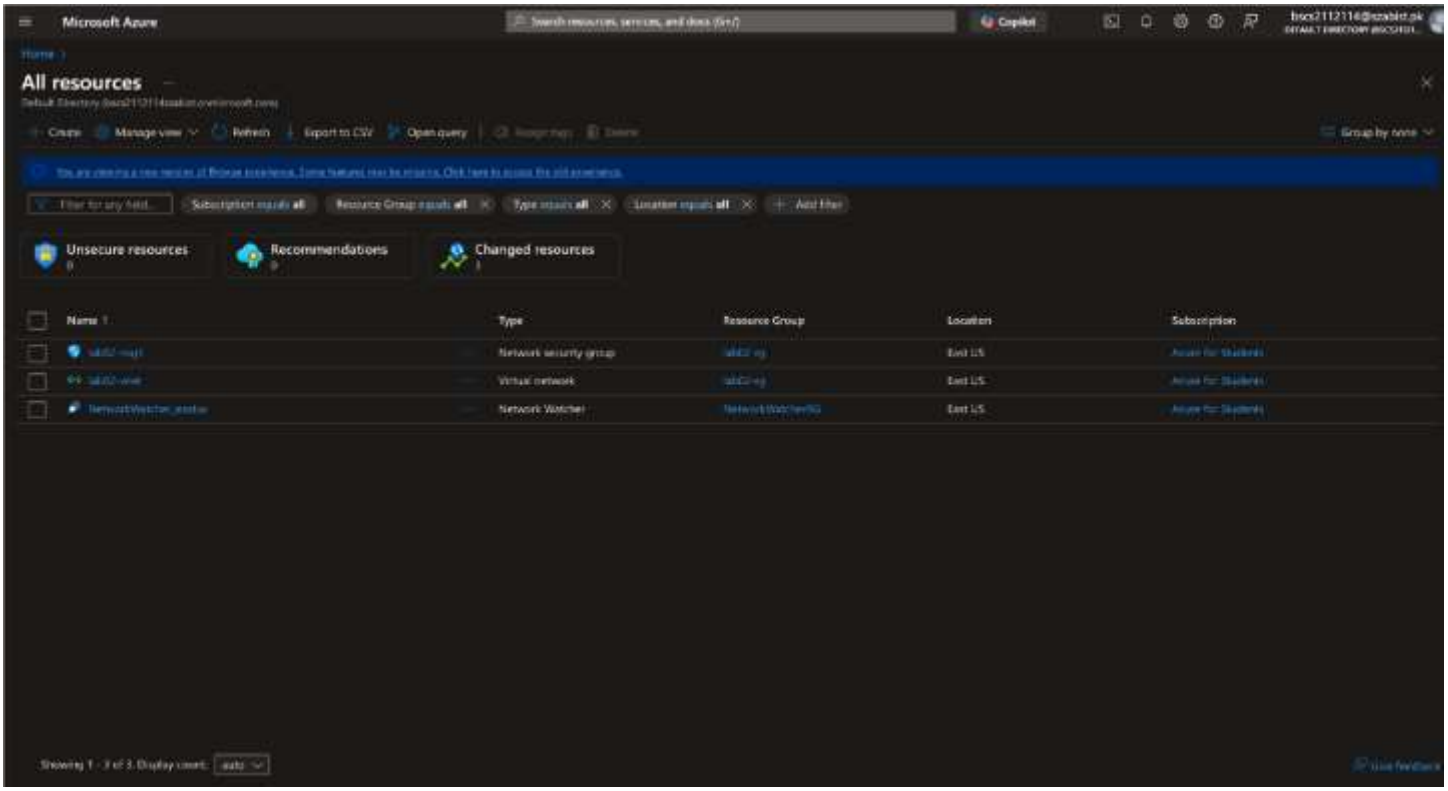
```
terraform apply
```

    - Type yes when prompted to confirm.

Part 7: Get Information from Terraform State

16. View and analyze state information:

terraform state list terraform



Part 9: Expand the lab02.S Script

```
cisco@Home lab02 % terraform state list
azurerm_network_security_group.lab02_nsg1
azurerm_resource_group.lab02_rg
azurerm_subnet.lab02_subnet1
azurerm_subnet_network_security_group_association.lab02_subnet_nsg_assoc
azurerm_virtual_network.lab02_vnet
cisco@Home lab02 % terraform show
# azurerm_network_security_group.lab02_nsg1:
resource "azurerm_network_security_group" "lab02_nsg1" {
    id                  = "/subscriptions/007fb41a4-0986-4eb3-a870-f5ef71f2eb88/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
    location            = "eastus"
    name                = "lab02-nsg1"
    resource_group_name = "lab02-rg"
    security_rule       = [
        {
            access                                     = "Allow"
            description                                = ""
            destination_address_prefix                 = "*"
            destination_address_prefixes               = []
            destination_application_security_group_ids = []
            destination_port_range                     = "22"
            destination_port_ranges                    = []
            direction                                  = "Inbound"
            name                                       = "rule1"
            priority                                   = 300
            protocol                                   = "Tcp"
            source_address_prefix                      = "*"
            source_address_prefixes                    = []
            source_application_security_group_ids      = []
            source_port_range                          = "*"
            source_port_ranges                         = []
        },
    ]
}

# azurerm_resource_group.lab02_rg:
resource "azurerm_resource_group" "lab02_rg" {
    id       = "/subscriptions/007fb41a4-0986-4eb3-a870-f5ef71f2eb88/resourcegroups/lab02-rg"
    location = "eastus"
    name     = "lab02-rg"
}

# azurerm_subnet.lab02_subnet1:
resource "azurerm_subnet" "lab02_subnet1" {
    address_prefixes                              = [
        "10.0.1.0/24",
    ]
    default_outbound_access_enabled               = true
    id                                            = "/subscriptions/007fb41a4-0986-4eb3-a870-f5ef71f2eb88/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
    name                                          = "lab02-subnet1"
    private_endpoint_network_policies             = "Disabled"
    private_link_service_network_policies_enabled = true
    resource_group_name                           = "lab02-rg"
    virtual_network_name                          = "lab02-vnet"
}

# azurerm_subnet_network_security_group_association.lab02_subnet_nsg_assoc:
resource "azurerm_subnet_network_security_group_association" "lab02_subnet_nsg_assoc" {
    id                        = "/subscriptions/007fb41a4-0986-4eb3-a870-f5ef71f2eb88/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
    network_security_group_id = "/subscriptions/007fb41a4-0986-4eb3-a870-f5ef71f2eb88/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
    subnet_id                 = "/subscriptions/007fb41a4-0986-4eb3-a870-f5ef71f2eb88/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2"
}

# azurerm_virtual_network.lab02_vnet:
resource "azurerm_virtual_network" "lab02_vnet" {
    address_space       = [
        "10.0.0.0/16",
    ]
    dns_servers         = []
    flow_timeout_in_minutes = 0
    guid                = "0b51bf1b-d08e-4ceb-a2ca-8f97a8fcc2f"
    id                  = "/subscriptions/007fb41a4-0986-4eb3-a870-f5ef71f2eb88/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet"
    location            = "eastus"
    name                = "lab02-vnet"
    private_endpoint_vnet_policies = "Disabled"
    resource_group_name = "lab02-rg"
    subnet              = []
}
cisco@Home lab02 %
```

Part 8: Confirm Resource Creation in Azure

17. Log in to the Azure Portal. Navigate to the resource group and confirm all resources exist as per the specifications.
    SCREENSHOT (capture the Azure Portal showing the resource group and resources).

20. Open lab02.U in your text editor and add the following:
    a. Add another subnet to the virtual network called lab02-subnet2 using azurerm_subnet.
    b. Define a network security group called lab02-nsg2 with two inbound allow TCP rules: o Port 3389 called rule1 with priority 100. o Port 5985 called rule2 with priority 200 using azurerm_network_security_group.
    c. A^ach the network security group lab02-nsg2 to lab02-subnet2 using azurerm_subnet_network_security_group_association.

```hcl
provider "azurerm" {
  subscription_id = "05fb4144-0806-4eb3-a870-f5ef7af2eb08"
  features {}
}

# Resource Group
resource "azurerm_resource_group" "lab02_rg" {
  name     = "lab02-rg"
  location = "eastus"
}

# Virtual Network
resource "azurerm_virtual_network" "lab02_vnet" {
  name                = "lab02-vnet"
  address_space       = ["10.0.0.0/16"]
  location            = azurerm_resource_group.lab02_rg.location
  resource_group_name = azurerm_resource_group.lab02_rg.name
}

# Subnet in the Virtual Network
resource "azurerm_subnet" "lab02_subnet1" {
  name                 = "lab02-subnet1"
  resource_group_name  = azurerm_resource_group.lab02_rg.name
  virtual_network_name = azurerm_virtual_network.lab02_vnet.name
  address_prefixes     = ["10.0.1.0/24"]
}

# Network Security Group with an inbound rule
resource "azurerm_network_security_group" "lab02_nsg1" {
  name                = "lab02-nsg1"
  location            = azurerm_resource_group.lab02_rg.location
  resource_group_name = azurerm_resource_group.lab02_rg.name

  security_rule {
    name                       = "rule1"
    priority                   = 100
    direction                  = "Inbound"
    access                     = "Allow"
    protocol                   = "Tcp"
    source_port_range          = "*"
    destination_port_range     = "22"
    source_address_prefix      = "*"
    destination_address_prefix = "*"
  }
}

# Associate the NSG with the Subnet
resource "azurerm_subnet_network_security_group_association" "lab02_subnet_nsg_assoc" {
  subnet_id                 = azurerm_subnet.lab02_subnet1.id
  network_security_group_id = azurerm_network_security_group.lab02_nsg1.id
}
```

```hcl
# Second Subnet
resource "azurerm_subnet" "lab02_subnet2" {
  name                 = "lab02-subnet2"
  resource_group_name  = azurerm_resource_group.lab02_rg.name
  virtual_network_name = azurerm_virtual_network.lab02_vnet.name
  address_prefixes     = ["10.0.2.0/24"]
}

# Second NSG with two inbound rules
resource "azurerm_network_security_group" "lab02_nsg2" {
  name                = "lab02-nsg2"
  location            = azurerm_resource_group.lab02_rg.location
  resource_group_name = azurerm_resource_group.lab02_rg.name

  security_rule {
    name                       = "rule1"
    priority                   = 100
    direction                  = "Inbound"
    access                     = "Allow"
    protocol                   = "Tcp"
    source_port_range          = "*"
    destination_port_range     = "3389"
    source_address_prefix      = "*"
    destination_address_prefix = "*"
  }

  security_rule {
    name                       = "rule2"
    priority                   = 200
    direction                  = "Inbound"
    access                     = "Allow"
    protocol                   = "Tcp"
    source_port_range          = "*"
    destination_port_range     = "5985"
    source_address_prefix      = "*"
    destination_address_prefix = "*"
  }
}

# Associate lab02-nsg2 with lab02-subnet2
resource "azurerm_subnet_network_security_group_association" "lab02_subnet2_nsg_assoc" {
  subnet_id                 = azurerm_subnet.lab02_subnet2.id
  network_security_group_id = azurerm_network_security_group.lab02_nsg2.id
}
```

Part 10: Validate Configuration

21. Validate configuration to ensure there are no errors or typos:
22. Fix any issues in the lab02.U file if reported.
23. Re-run the validation until no errors are reported:

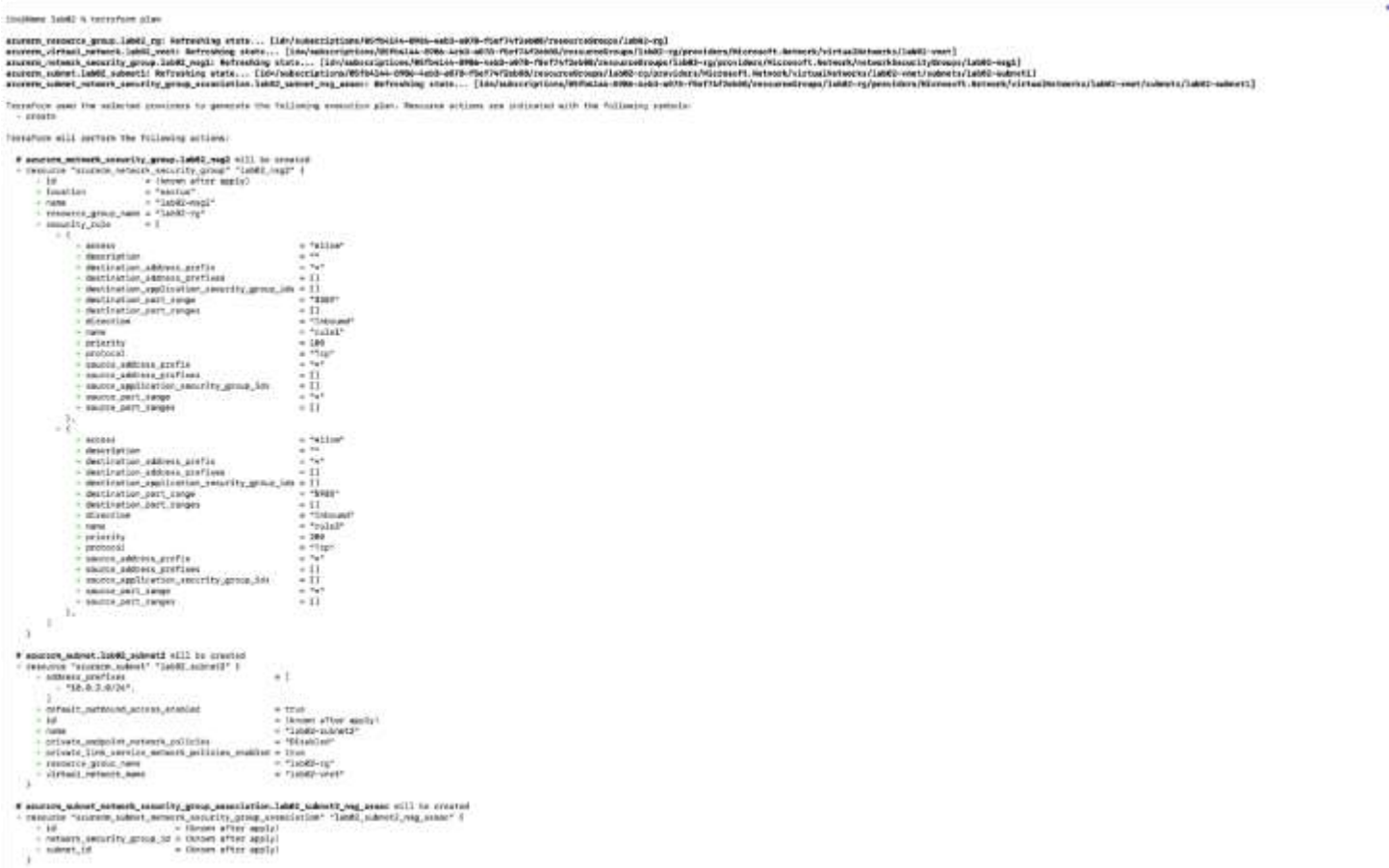
```
ibs@Home lab02 % terraform validate

Success! The configuration is valid.

ibs@Home lab02 % terraform plan
```

Part 11: Run Simulation

24. Perform a dry run:

```
terraform plan
```

25. Observe output closely. Note resources with +, -, or -/+ signs.
26. Fix any issues in the lab02.U file if reported.



27. Redo the dry run until no errors are reported:

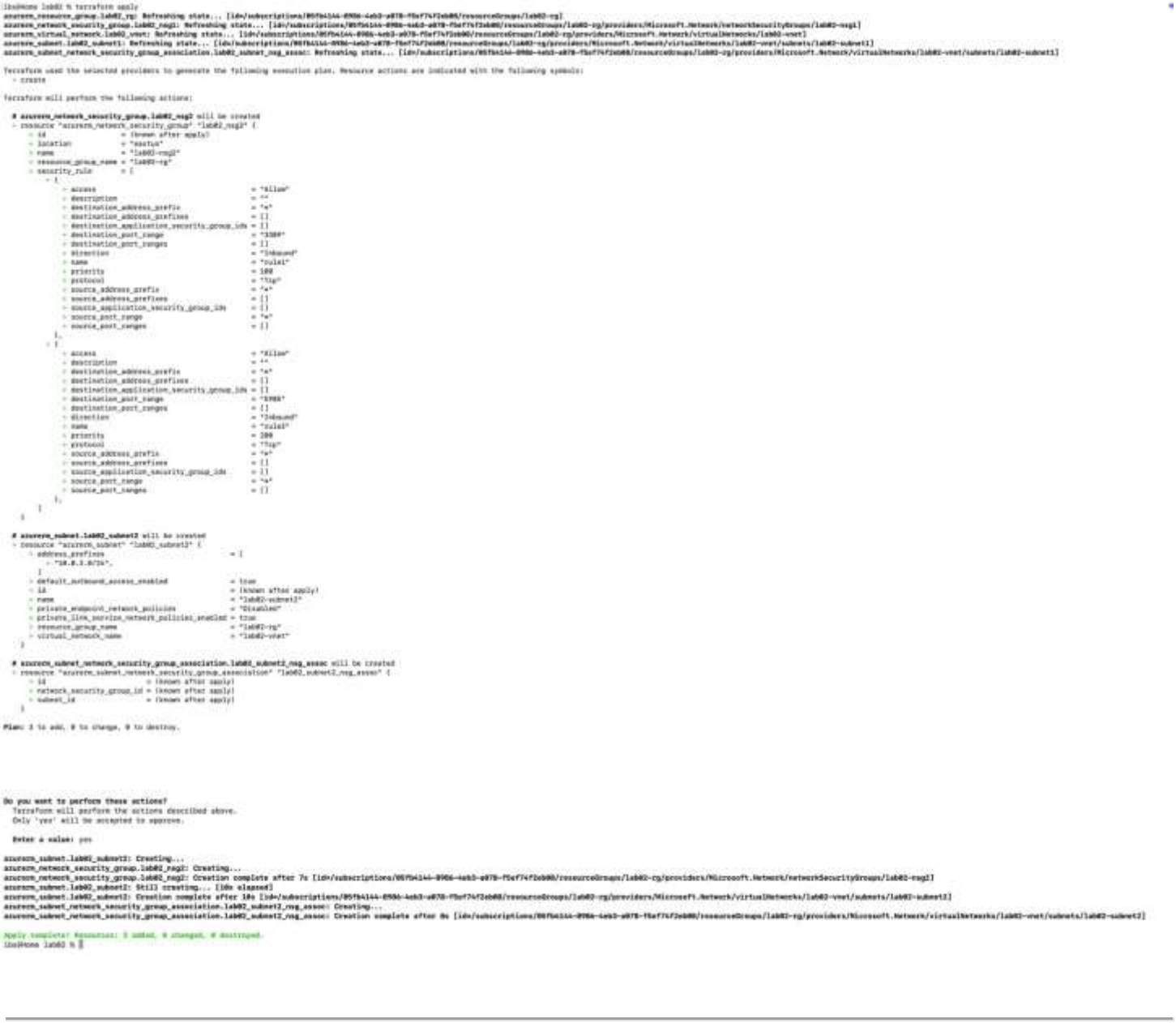
```
  # azurerm_subnet.lab02_subnet2 will be created
  + resource "azurerm_subnet" "lab02_subnet2" {
      + address_prefixes                              = [
          + "10.0.2.0/24",
        ]
      + default_outbound_access_enabled               = true
      + id                                            = (known after apply)
      + name                                          = "lab02-subnet2"
      + private_endpoint_network_policies             = "Disabled"
      + private_link_service_network_policies_enabled = true
      + resource_group_name                           = "lab02-rg"
      + virtual_network_name                          = "lab02-vnet"
    }

  # azurerm_subnet_network_security_group_association.lab02_subnet2_nsg_assoc will be created
  + resource "azurerm_subnet_network_security_group_association" "lab02_subnet2_nsg_assoc" {
      + id                        = (known after apply)
      + network_security_group_id = (known after apply)
      + subnet_id                 = (known after apply)
    }

Plan: 3 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
ibs@Home lab02 %
```

Part 12: Deploy Infrastructure

28. Deploy the infrastructure and monitor progress:

```
terraform apply
```

- Type yes when prompted.



Part 13: Get Information from Terraform State

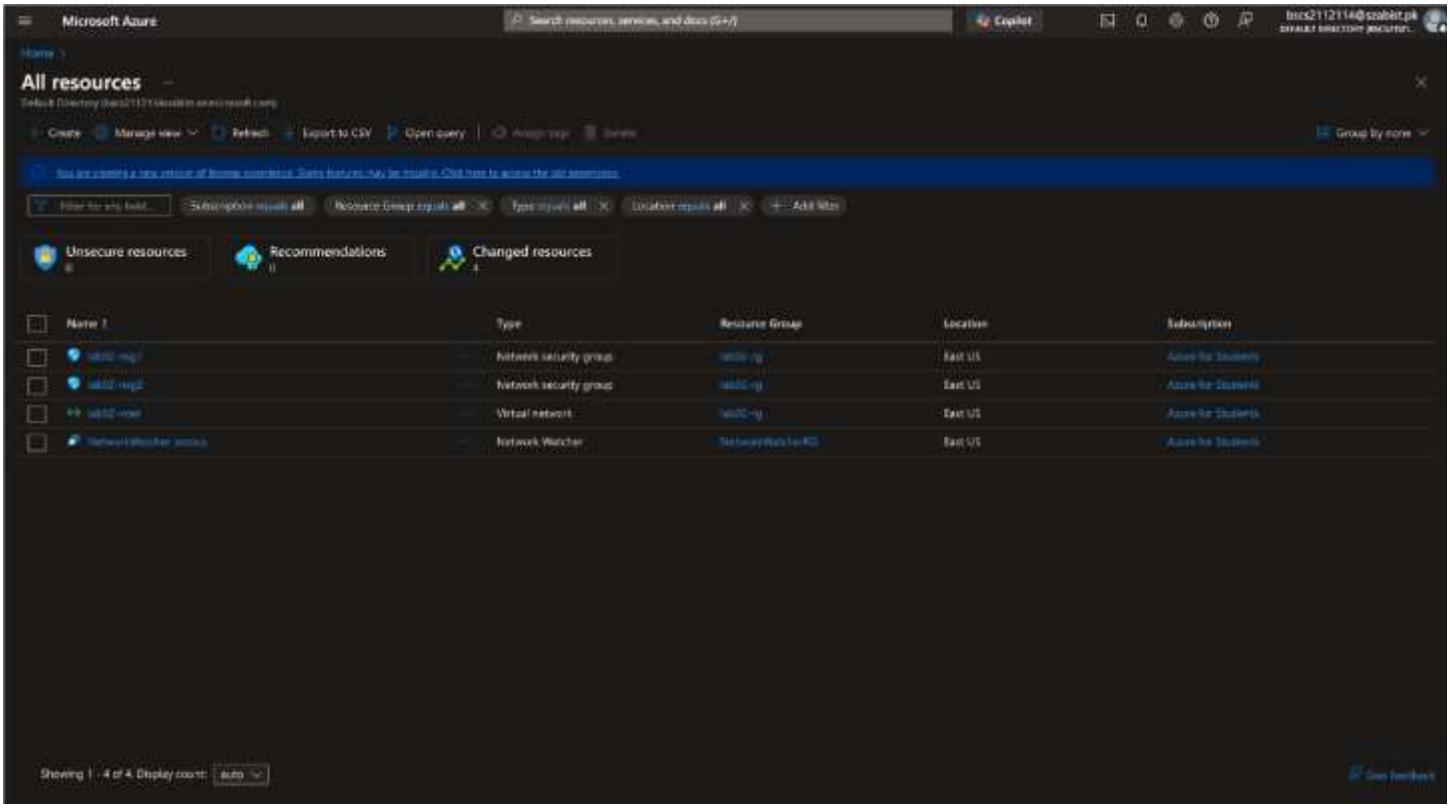
31. View and analyze state information:

```
terraform state list
```

```
ibs@Home lab02 % terraform state list
azurerm_network_security_group.lab02_nsg1
azurerm_network_security_group.lab02_nsg2
azurerm_resource_group.lab02_rg
azurerm_subnet.lab02_subnet1
azurerm_subnet.lab02_subnet2
azurerm_subnet_network_security_group_association.lab02_subnet2_nsg_assoc
azurerm_subnet_network_security_group_association.lab02_subnet_nsg_assoc
azurerm_virtual_network.lab02_vnet
ibs@Home lab02 %
```

Part 14: Confirm Resource Creation in Azure

32. Log in to the Azure Portal. Navigate to the resource group and confirm all resources exist as per the specifications.



Part 15: Destroy All Resources and Verify

33. Destroy all the resources:

```
terraform destroy
```

    - Type yes when prompted.

34. Verify deletion:

```
terraform state list
```

terraform show

```
ibs@Home lab02 % terraform state list
ibs@Home lab02 % terraform show
The state file is empty. No resources are represented.
ibs@Home lab02 %
```