

Lab 03 (Modules 04 and 05)

Parametrize lab02 Terraform configuration



SZABIST

Name	Aliyan
Registration Id	2112104
Section	BSCS 8 th A
Course	Introduction to DevOps

Instructions:

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-lab03.

Lab Objectives:

There are 5 sections in this lab. Each section has a different set of objectives:

- **Section 1:** Parametrize lab02 Terraform configuration.
 - **Section 2:** Expand the parametrized Terraform configuration from Section 1 and add a Linux virtual machine to the landscape.
 - **Section 3:** Expand Section 2 and define and consume local values.
 - **Section 4:** Expand Section 3 and define output blocks.
 - **Section 5:** Expand Section 4 and add explicit lifecycle rules and dependencies.
-

Section 1

Objectives:

- Move resource values to a separate file as variable blocks.
- Update main Terraform file to use variables.
- 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 lab03s1 in your home directory.
3. Copy lab02.tf file into lab03s1 directory.
4. Change into the lab03s1 directory.
5. Rename lab02.tf as lab03s1.tf.
6. Create an empty file called vars03s1.tf.
7. Create an empty file called providers.tf.

Part 2: Update vars03s1.tf File

8. Copy provider and Terraform blocks from lab03s1.tf to providers.tf (use a text editor like Notepad or VS Code).
9. Open vars03s1.tf in a text editor and define variable blocks as follows. Enclose values in double quotation marks:
 - a. One block for resource group name.
 - b. One block for location.
 - c. One block for virtual network name.
 - d. One block for virtual network address space.
 - e. Two blocks for subnet names (one per subnet).

- f. Two blocks for subnet address spaces (one per address space).
- g. Two blocks for network security groups (one per network security group).

SCREENSHOT of vars03s1.tf (capture the full content in your editor).

```
File    Edit    View

# Random number to append to container name
resource "random_integer" "rand" {
  min = 10000
  max = 99999
}

# Resource Group
resource "azurerm_resource_group" "lab02_rg" {
  name      = var.resource_group_name
  location  = var.location
}

# Virtual Network
resource "azurerm_virtual_network" "lab02_vnet" {
  name          = var.virtual_network_name
  location      = var.location
  resource_group_name = azurerm_resource_group.lab02_rg.name
  address_space   = [var.virtual_network_address_space]
}

# Subnet 1
resource "azurerm_subnet" "lab02_subnet1" {
  name          = var.subnet1_name
  resource_group_name = azurerm_resource_group.lab02_rg.name
  virtual_network_name = azurerm_virtual_network.lab02_vnet.name
  address_prefixes  = [var.subnet1_address_space]
}

# Subnet 2
resource "azurerm_subnet" "lab02_subnet2" {
  name          = var.subnet2_name
  resource_group_name = azurerm_resource_group.lab02_rg.name
  virtual_network_name = azurerm_virtual_network.lab02_vnet.name
  address_prefixes  = [var.subnet2_address_space]
}

# NSG 1
resource "azurerm_network_security_group" "lab02_nsg1" {
  name      = var.nsg1_name
  location  = var.location
}
```

```
# NSG 1
resource "azurerm_network_security_group" "lab02_nsg1" {
    name          = var.nsg1_name
    location      = var.location
    resource_group_name = azurerm_resource_group.lab02_rg.name

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

# NSG 2
resource "azurerm_network_security_group" "lab02_nsg2" {
    name          = var.nsg2_name
    location      = var.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
    }
}
```

```
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 NSG1 with Subnet1
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc1" {
    subnet_id          = azurerm_subnet.lab02_subnet1.id
    network_security_group_id = azurerm_network_security_group.lab02_nsg1.id
}

# Associate NSG2 with Subnet2
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc2" {
    subnet_id          = azurerm_subnet.lab02_subnet2.id
    network_security_group_id = azurerm_network_security_group.lab02_nsg2.id
}

# Container Group
resource "azurerm_container_group" "lab02_container" {
    name          = "lab02-container-${random_integer.rand.result}"
    location      = var.location
    resource_group_name = azurerm_resource_group.lab02_rg.name
    os_type       = "Linux"
    ip_address_type = "Public"
    dns_name_label = "lab02-container-${random_integer.rand.result}"

    container {
        name    = "helloworld"
        image   = "mcr.microsoft.com/azuredocs/aci-helloworld"
        cpu     = 1
        memory = 1.5
    }
}
```

```
container {
    name      = "helloworld"
    image     = "mcr.microsoft.com/azuredocs/aci-helloworld"
    cpu       = 1
    memory   = 1.5

    ports {
        port      = 80
        protocol = "TCP"
    }
}

tags = {
    environment = "lab"
}
}
```

Part 3: Update lab03s1.tf File

10. Open lab03s1.tf in a text editor and update as follows:

- Remove provider and Terraform blocks.
- Update all resource blocks to source values from vars03s1.tf.

SCREENSHOT of lab03s1.tf (capture the full content in your editor).

File Edit View

```
# Random number to append to container name
resource "random_integer" "rand" {
    min = 10000
    max = 99999
}

# Resource Group
resource "azurerm_resource_group" "lab02_rg" {
    name      = var.resource_group_name
    location  = var.location
}

# Virtual Network
resource "azurerm_virtual_network" "lab02_vnet" {
    name          = var.virtual_network_name
    location      = var.location
    resource_group_name = azurerm_resource_group.lab02_rg.name
    address_space   = [var.virtual_network_address_space]
}

# Subnet 1
resource "azurerm_subnet" "lab02_subnet1" {
    name          = var.subnet1_name
    resource_group_name = azurerm_resource_group.lab02_rg.name
    virtual_network_name = azurerm_virtual_network.lab02_vnet.name
    address_prefixes = [var.subnet1_address_space]
}

# Subnet 2
resource "azurerm_subnet" "lab02_subnet2" {
    name          = var.subnet2_name
    resource_group_name = azurerm_resource_group.lab02_rg.name
    virtual_network_name = azurerm_virtual_network.lab02_vnet.name
    address_prefixes = [var.subnet2_address_space]
}

# NSG 1
resource "azurerm_network_security_group" "lab02_nsg1" {
    name          = var.nsg1_name
    location      = var.location
}
```

```
# NSG 1
resource "azurerm_network_security_group" "lab02_nsg1" {
    name          = var.nsg1_name
    location      = var.location
    resource_group_name = azurerm_resource_group.lab02_rg.name

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

# NSG 2
resource "azurerm_network_security_group" "lab02_nsg2" {
    name          = var.nsg2_name
    location      = var.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
    }
}
```

```
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 NSG1 with Subnet1
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc1" {
    subnet_id          = azurerm_subnet.lab02_subnet1.id
    network_security_group_id = azurerm_network_security_group.lab02_nsg1.id
}

# Associate NSG2 with Subnet2
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc2" {
    subnet_id          = azurerm_subnet.lab02_subnet2.id
    network_security_group_id = azurerm_network_security_group.lab02_nsg2.id
}

# Container Group
resource "azurerm_container_group" "lab02_container" {
    name          = "lab02-container-${random_integer.rand.result}"
    location      = var.location
    resource_group_name = azurerm_resource_group.lab02_rg.name
    os_type       = "Linux"
    ip_address_type = "Public"
    dns_name_label = "lab02-container-${random_integer.rand.result}"

    container {
        name    = "helloworld"
        image   = "mcr.microsoft.com/azuredocs/aci-helloworld"
        cpu     = 1
        memory = 1.5
    }
}
```

```
container {
    name      = "helloworld"
    image     = "mcr.microsoft.com/azuredocs/aci-helloworld"
    cpu       = 1
    memory   = 1.5

    ports {
        port      = 80
        protocol = "TCP"
    }
}

tags = {
    environment = "lab"
}
}
```

Part 4: Initialize Terraform

11. Initialize Terraform to download plug-ins as required:

Part 5: Validate Configuration

13. Validate the configuration to ensure no errors or typos:
14. Fix any issues in lab03s1.tf and/or vars03s1.tf files if reported (edit in your text editor).
15. Re-run validation until no errors are reported.

SCREENSHOT (capture the successful validation output).

```
C:\Users\SCS\lab03s1>terraform init
Initializing the backend...
Initializing provider plugins...
- Finding hashicorp/azurerm versions matching "~> 3.0"...
- Finding hashicorp/random versions matching "~> 3.0"...
- Installing hashicorp/azurerm v3.117.1...
- Installed hashicorp/azurerm v3.117.1 (signed by HashiCorp)
- Installing hashicorp/random v3.7.1...
- Installed hashicorp/random v3.7.1 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

```
C:\Users\SCS\lab03s1>terraform validate  
Success! The configuration is valid.
```

Part 6: Run Simulation

16. Perform a dry run:

```
terraform plan
```

17. Review output and ensure all configuration meets requirements. Observe resources with +, -, or -/+ signs.
18. Fix any issues in lab03s1.tf and/or vars03s1.tf files if reported.
19. Redo the dry run until no errors are reported.

Part 8: Deploy Infrastructure

20. Deploy the infrastructure and monitor progress.
 - o Type yes when prompted.

Part 9: Get Information from Terraform State

22. View and analyze state information:

```
terraform state list  
terraform show
```

SCREENSHOT (capture both outputs).

```
C:\Users\SCS\lab03s1>terraform state list  
azurerm_container_group.lab02_container  
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_nsg_assoc1  
azurerm_subnet_network_security_group_association.lab02_nsg_assoc2  
azurerm_virtual_network.lab02_vnet  
random_integer.rand
```

```
C:\Users\SCS\lab03s1>terraform show
# azurerm_container_group.lab02_container:
resource "azurerm_container_group" "lab02_container" {
  dns_name_label          = "lab02-container-37757"
  dns_name_label_reuse_policy = "Unsecure"
  exposed_port            = [
    {
      port      = 80
      protocol = "TCP"
    },
    {
      port      = 80
      protocol = "TCP"
    }
  ]
  fqdn                   = "lab02-container-37757.eastus.azurecontainer.io"
  id                     = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/containers/labGroups/lab02-container-37757"
  ip_address             = "52.226.110.71"
  ip_address_type        = "Public"
  location               = "eastus"
  name                   = "lab02-container-37757"
  os_type                = "Linux"
  priority               = null
  resource_group_name    = "lab02-rg"
  restart_policy          = "Always"
  sku                    = "Standard"
  tags                   = {
    "environment" = "lab"
  }

  container {
    commands      = []
    cpu           = 1
    cpu_limit     = 0
    image          = "mcr.microsoft.com/azuredocs/aci-helloworld"
    memory         = 1.5
    memory_limit   = 0
    name           = "helloworld"

    ports {
      port      = 80
      protocol = "TCP"
    }
  }
}

# azurerm_network_security_group.lab02_nsg1:
resource "azurerm_network_security_group" "lab02_nsg1" {
  id           = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
  location     = "eastus"
  name         = "lab02-nsg1"
  resource_group_name = "lab02-rg"
  security_rule = [
    {
      access          = "Allow"
      description     = null
      destination_address_prefix = "*"
      destination_address_prefixes = []
      destination_application_security_group_ids = []
      destination_port_range = "22"
      destination_port_ranges = []
      direction        = "Inbound"
      name             = "SSH"
      priority         = 100
      protocol         = "Tcp"
      source_address_prefix = "*"
      source_address_prefixes = []
      source_application_security_group_ids = []
      source_port_range = "*"
      source_port_ranges = []
    },
  ],
}

# azurerm_network_security_group.lab02_nsg2:
resource "azurerm_network_security_group" "lab02_nsg2" {
  id           = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2"
  location     = "eastus"
  name         = "lab02-nsg2"
  resource_group_name = "lab02-rg"
}
```

```
    protocol = "TCP"
  }
}

# azurerm_network_security_group.lab02_nsg1:
resource "azurerm_network_security_group" "lab02_nsg1" {
  id           = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
  location     = "eastus"
  name         = "lab02-nsg1"
  resource_group_name = "lab02-rg"
  security_rule = [
    {
      access          = "Allow"
      description     = null
      destination_address_prefix = "*"
      destination_address_prefixes = []
      destination_application_security_group_ids = []
      destination_port_range = "22"
      destination_port_ranges = []
      direction        = "Inbound"
      name             = "SSH"
      priority         = 100
      protocol         = "Tcp"
      source_address_prefix = "*"
      source_address_prefixes = []
      source_application_security_group_ids = []
      source_port_range = "*"
      source_port_ranges = []
    },
  ],
}

# azurerm_network_security_group.lab02_nsg2:
resource "azurerm_network_security_group" "lab02_nsg2" {
  id           = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2"
  location     = "eastus"
  name         = "lab02-nsg2"
  resource_group_name = "lab02-rg"
}
```

```

name          = "lab02-nsg2"
resource_group_name = "lab02-rg"
security_rule      = [
    {
        access           = "Allow"
        description      = null
        destination_address_prefix = "*"
        destination_address_prefixes = []
        destination_application_security_group_ids = []
        destination_port_range      = "3389"
        destination_port_ranges     = []
        direction           = "Inbound"
        name               = "rule1"
        priority           = 100
        protocol           = "Tcp"
        source_address_prefix = "*"
        source_address_prefixes = []
        source_application_security_group_ids = []
        source_port_range      = "*"
        source_port_ranges     = []
    },
    {
        access           = "Allow"
        description      = null
        destination_address_prefix = "*"
        destination_address_prefixes = []
        destination_application_security_group_ids = []
        destination_port_range      = "5985"
        destination_port_ranges     = []
        direction           = "Inbound"
        name               = "rule2"
        priority           = 200
        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/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg"
    location  = "eastus"
    managed_by = null
    name      = "lab02-rg"
    tags      = {}
}

# azurerm_subnet.lab02_subnet1:
resource "azurerm_subnet" "lab02_subnet1" {
    address_prefixes      = [
        "10.0.1.0/24",
    ]
    default_outbound_access_enabled = true
    enforce_private_link_endpoint_network_policies = false
    enforce_private_link_service_network_policies = false
    id                   = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
    name                = "lab02-subnet1"
    private_endpoint_network_policies = "Enabled"
    private_endpoint_network_policies_enabled = true
    private_link_service_network_policies = true
    resource_group_name   = "lab02-rg"
    virtual_network_name = "lab02-vnet"
}

# azurerm_subnet.lab02_subnet2:
resource "azurerm_subnet" "lab02_subnet2" {
    address_prefixes      = [
        "10.0.2.0/24",
    ]
    default_outbound_access_enabled = true
    enforce_private_link_endpoint_network_policies = false
    enforce_private_link_service_network_policies = false
    id                   = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2"
    name                = "lab02-subnet2"
}

```

```

k/virtualNetworks/lab02-vnet/subnets/lab02-subnet2"
  name                  = "lab02-subnet2"
  private_endpoint_network_policies      = "Enabled"
  private_endpoint_network_policies_enabled = true
  private_link_service_network_policies_enabled = true
  resource_group_name        = "lab02-rg"
  virtual_network_name       = "lab02-vnet"
}

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc1:
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc1" {
  id          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
  network_security_group_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
  subnet_id    = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
}

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc2:
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc2" {
  id          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2"
  network_security_group_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2"
  subnet_id    = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/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",
  ]
  bgp_community     = null
  dns_servers       = []
  edge_zone         = null
  flow_timeout_in_minutes = 0
  guid              = "dbdb4a10-5015-4626-b13e-52aab9dfb46"
  id                = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet"
}

# azurerm_virtual_network.lab02_vnet:
resource "azurerm_virtual_network" "lab02_vnet" {
  address_space     = [
    "10.0.0.0/16",
  ]
  bgp_community     = null
  dns_servers       = []
  edge_zone         = null
  flow_timeout_in_minutes = 0
  guid              = "dbdb4a10-5015-4626-b13e-52aab9dfb46"
  id                = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet"
  location          = "eastus"
  name              = "lab02-vnet"
  resource_group_name = "lab02-rg"
  subnet            = []
}

# random_integer.rand:
resource "random_integer" "rand" {
  id      = "37757"
  max    = 99999
  min    = 10000
  result = 37757
}

```

C:\Users\SCS\lab03s1>

Part 10: Confirm Resource Creation in Azure

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

SCREENSHOT (capture the Azure Portal showing resources).

Microsoft Azure

Search resources, services, and docs (G+)

Copilot

bscs2112104@szabist.pk

Home > Resource groups

You are viewing a new version of Browse experience. Some features may be missing. Click here to access the old experience.

Create Manage view Refresh Export to CSV Open query Assign tags

Group by none

Filter for any field... Subscription equals all Location equals all Add filter

Name	Subscription	Location
lab02-rg	Azure for Students	East US
NetworkWatcherRG	Azure for Students	East US

Showing 1 - 2 of 2. Display count: 10

Microsoft Azure

Search resources, services, and docs (G+)

Copilot

bscs2112104@szabist.pk

Home > Resource groups > lab02-rg

You are viewing a new version of Browse experience. Some features may be missing. Click here to access the old experience.

Create Group by none

Overview Essentials

Resources Recommendations

Filter for any field... Type equals all Location equals all Add filter

Showing 1 to 4 of 4 records. Show hidden types

Name	Type	Location
lab02-container-37757	Container instances	East US
lab02-nsg1	Network security group	East US
lab02-nsg2	Network security group	East US
lab02-vnet	Virtual network	East US

Showing 1 - 2 of 2. Display count: 10

< Previous Page 1 of 1 Next >

Give feedback

Microsoft Azure

Home > Resource groups > lab02-rg > lab02-vnet

lab02-vnet | Subnets

Virtual network

Search Subnets

Name	IPv4	IPv6	Available IPs	Delegated to	Security group	Route table
lab02-subnet2	10.0.2.0/24	-	251	-	lab02-nsg2	-
lab02-subnet1	10.0.1.0/24	-	251	-	lab02-nsg1	-

Showing 2 subnets

<https://portal.azure.com/#@bsc2112104@szabist.onmicrosoft.com/resource/subscriptions/eaf6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets>

Microsoft Azure

Home > Resource groups > lab02-rg

lab02-container-37757

Container instances

Search Start Restart Stop Delete Refresh Give feedback

Overview

Essentials

Resource group (move) : lab02-rg	SKU : Standard
Status : Running	OS type : Linux
Location : East US	IP address (Public) : 52.226.110.71
Subscription (move) : Azure for Students	FQDN : lab02-container-37757.eastus.azurecontainer.io
Subscription ID : eaf6fc9c7-fe32-4565-b42a-e336c72040d9	Container count : 1
Tags (edit) : environment : lab	

CPU (millicores)

Memory

JSON View

Microsoft Azure

All services >

Network security groups

Default Directory (bsc2112104@szabist.onmicrosoft.com)

Create Manage view Refresh Export to CSV Open query Assign tags

Filter for any field... Subscription equals all Resource group equals all Location equals all Add filter

No grouping List view

Showing 1 to 2 of 2 records.

Name	Resource group	Location	Subscription	Flow log
lab02-nsg1	lab02-rg	East US	Azure for Students	***
lab02-nsg2	lab02-rg	East US	Azure for Students	***

< Previous Page 1 of 1 Next >

Give feedback

Part 11: Destroy All Resources and Verify

24. Destroy all resources:

```
terraform destroy
```

- Type yes when prompted.

25. Verify deletion:

```
terraform state list
```

```
terraform show
```

SCREENSHOT (capture the verification output).

```
Destroy complete! Resources: 10 destroyed.
```

```
C:\Users\SCS\lab03s1>terraform state list
```

```
C:\Users\SCS\lab03s1>terraform show
```

```
The state file is empty. No resources are represented.
```

```
C:\Users\SCS\lab03s1>
```

Section 2

Objectives:

- Use configuration from Section 1.
- Define resource and variable blocks for virtual machine resources.
- Validate, deploy, expand, analyze, and destroy infrastructure.

Part 1: Prepare for the Lab

1. Create a directory called lab03s2 in your home directory:
2. Copy lab03s1.tf, vars03s1.tf, and providers.tf into lab03s2 directory:
3. Change into the lab03s2 directory:
4. Create two empty files called lab03s2.tf and vars03s2.tf:

Part 2: Update vars03s2.tf File

5. Open vars03s2.tf in a text editor and define variable blocks for a Linux virtual machine. Enclose values in double quotation marks:
 - a. Name (e.g., linux_name): "lab03s2-db1-u-vm1".
 - b. Size: "Standard_B1s".
 - c. Admin user name: "<firstname-yourHumberID>" [from Lab 01].
 - d. Public key: "C:\Users\<YourWindowsUsername>\.ssh\id_rsa.pub" (adjust path to your SSH public key location on Windows).
 - e. OS disk attributes:
 - i. Storage account type: "Premium_LRS".
 - ii. Disk size: "32".

iii. Caching: "ReadWrite".

f. Ubuntu Linux OS information:

i. Publisher: "Canonical".

ii. Offer: "UbuntuServer".

iii. Sku: "19.04".

iv. Version: "latest".

SCREENSHOT of vars03s2.tf (capture the full content in your editor).

```
File Edit View

variable "linux_name" {
  description = "Name of the Linux VM"
  type        = string
  default     = "lab03s2-db1-u-vm1"
}

variable "vm_size" {
  description = "The size of the virtual machine"
  type        = string
  default     = "Standard_B1s"
}

variable "admin_username" {
  description = "Admin user name for the VM"
  type        = string
  default     = "scs-yourHumberID" # Replace with your actual username (e.g., "scs-12345")
}

variable "public_key" {
  description = "The path to the public key for SSH authentication"
  type        = string
  default     = "C:/Users/scs/.ssh/id_rsa.pub" # Updated to point to the new RSA public key
}

variable "os_disk_storage_account_type" {
  description = "Storage account type for the OS disk"
  type        = string
  default     = "Premium_LRS"
}

variable "os_disk_size" {
  description = "Size of the OS disk in GB"
  type        = number
  default     = 32
}

variable "os_disk_caching" {
  description = "Caching mode for the OS disk"
  type        = string
  default     = "ReadWrite"
```

```

variable "os_disk_caching" {
  description = "Caching mode for the OS disk"
  type        = string
  default     = "ReadWrite"
}

variable "ubuntu_publisher" {
  description = "Publisher for the Ubuntu image"
  type        = string
  default     = "Canonical"
}

variable "ubuntu_offer" {
  description = "Offer for the Ubuntu image"
  type        = string
  default     = "UbuntuServer"
}

variable "ubuntu_sku" {
  description = "SKU for the Ubuntu image"
  type        = string
  default     = "19.04"
}

variable "ubuntu_version" {
  description = "Version for the Ubuntu image"
  type        = string
  default     = "latest"
}

```

Part 3: Update lab03s2.tf File

6. Open lab03s2.tf in a text editor and define resource blocks as follows:
 - a. Define network interface called \${var.linux_name}-nic with IP configuration name \${var.linux_name}-ipconfig1 using azurerm_network_interface. Use Dynamic IP address allocation.
 - b. Define public IP address called \${var.linux_name}-pip using azurerm_public_ip. Use Dynamic IP address allocation method.
 - c. Define virtual machine using azurerm_linux_virtual_machine. Use \${var.linux_name}-osdisk as the OS disk name.

SCREENTHOT of lab03s2.tf (capture the full content in your editor).

Note: At this point, you should have 5 Terraform files—providers.tf, lab03s1.tf, vars03s1.tf, lab03s2.tf, and vars03s2.tf—under the lab03s2 directory.

File Edit View

```
# Public IP Address
resource "azurerm_public_ip" "linux_pip" {
    name          = "${var.linux_name}-pip"
    location      = var.location
    resource_group_name = var.resource_group_name
    allocation_method = "Dynamic"
}

# Network Interface
resource "azurerm_network_interface" "linux_nic" {
    name          = "${var.linux_name}-nic"
    location      = var.location
    resource_group_name = var.resource_group_name

    ip_configuration {
        name          = "${var.linux_name}-ipconfig1"
        subnet_id     = azurerm_subnet.lab02_subnet1.id
        private_ip_address_allocation = "Dynamic"
        public_ip_address_id       = azurerm_public_ip.linux_pip.id
    }
}

# Linux Virtual Machine
resource "azurerm_linux_virtual_machine" "linux_vm" {
    name          = var.linux_name
    location      = var.location
    resource_group_name = var.resource_group_name
    size          = var.vm_size
    admin_username = var.admin_username

    network_interface_ids = [
        azurerm_network_interface.linux_nic.id
    ]

    admin_ssh_key {
        username    = var.admin_username
        public_key  = file(var.public_key)
    }

    os_disk {
```

```

admin_ssh_key {
  username    = var.admin_username
  public_key  = file(var.public_key)
}

os_disk {
  name          = "${var.linux_name}-osdisk"
  caching       = var.os_disk_caching
  storage_account_type = var.os_disk_storage_account_type
  disk_size_gb   = var.os_disk_size
}

#  Updated image configuration for eastus
source_image_reference {
  publisher = "Canonical"
  offer     = "UbuntuServer"
  sku       = "18_04-lts-gen2"
  version   = "latest"
}

tags = {
  environment = "lab"
}
}

```

```

C:\Users\SCS\lab03s2>dir
Volume in drive C has no label.
Volume Serial Number is E8EA-1946

Directory of C:\Users\SCS\lab03s2

09/04/2025  01:56 pm    <DIR>          .
09/04/2025  01:53 pm    <DIR>          ..
09/04/2025  01:18 pm            3,853 lab03s1.tf
09/04/2025  02:03 pm            1,709 lab03s2.tf
09/04/2025  01:06 pm            342 providers.tf
09/04/2025  01:10 pm            1,680 vars03s1.tf
09/04/2025  02:00 pm            1,593 vars03s2.tf
                           5 File(s)      9,177 bytes
                           2 Dir(s)  17,776,865,280 bytes free

C:\Users\SCS\lab03s2>

```

Part 4: Initialize Terraform

7. Initialize Terraform to download plug-ins as required:

Part 5: Validate Configuration

8. Validate the configuration to ensure no errors or typos:

SCREENSHOT (capture the successful validation output).

```
C:\Users\SCS\lab03s2>terraform init
Initializing the backend...
Initializing provider plugins...
- Finding hashicorp/azurerm versions matching "~> 3.0"...
- Finding hashicorp/random versions matching "~> 3.0"...
- Installing hashicorp/random v3.7.1...
- Installed hashicorp/random v3.7.1 (signed by HashiCorp)
- Installing hashicorp/azurerm v3.117.1...
- Installed hashicorp/azurerm v3.117.1 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

C:\Users\SCS\lab03s2>terraform validate
Success! The configuration is valid.
```

Part 6: Run Simulation

11. Perform a dry run:

```
terraform plan
```

12. Review output and ensure all configuration meets requirements. Observe resources with +, -, or -/+ signs.
13. Fix any issues in the Terraform files if reported.
14. Redo the dry run until no errors are reported:

Part 7: Deploy Infrastructure

15. Deploy the infrastructure and monitor progress:

Part 8: Get Information from Terraform State

17. View and analyze state information:

```
terraform state list  
terraform show
```

SCRENSHOT (capture both outputs).

```
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.  
C:\Users\SCS\lab03s2>terraform state list  
azurerm_container_group.lab02_container  
azurerm_linux_virtual_machine.linux_vm  
azurerm_network_interface.linux_nic  
azurerm_network_security_group.lab02_nsg1  
azurerm_network_security_group.lab02_nsg2  
azurerm_public_ip.linux_pip  
azurerm_resource_group.lab02_rg  
azurerm_subnet.lab02_subnet1  
azurerm_subnet.lab02_subnet2  
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1  
azurerm_subnet_network_security_group_association.lab02_nsg_assoc2  
azurerm_virtual_network.lab02_vnet  
random_integer.rand  
  
C:\Users\SCS\lab03s2>terraform show  
# azurerm_container_group.lab02_container:  
resource "azurerm_container_group" "lab02_container" {  
  dns_name_label          = "lab02-container-72390"  
  dns_name_label_reuse_policy = "Unsecure"  
  exposed_port           = [  
    {  
      port      = 80  
      protocol = "TCP"  
    },  
  ]  
  fqdn                  = "lab02-container-72390.eastus.azurecontainer.io"  
  id                    = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/Lab02-rg/providers/Microsoft.ContainerInstance/containers/lab02-container-72390"  
  ip_address            = "4.255.32.213"  
  ip_address_type       = "Public"  
  location              = "eastus"  
  name                  = "lab02-container-72390"  
  os_type                = "Linux"  
  priority              = "null"  
  resource_group_name   = "lab02-rg"  
  restart_policy         = "Always"  
  sku                   = "Standard"
```

```

restart_policy          = "Always"
sku                   = "Standard"
subnet_ids            = []
tags                 = {}
  "environment" = "lab"
}
zones                = []

container {
  commands           = []
  cpu                = 1
  cpu_limit          = 0
  environment_variables = {}
  image              = "mcr.microsoft.com/azuredocs/aci-helloworld"
  memory             = 1.5
  memory_limit       = 0
  name               = "helloworld"
  secure_environment_variables = (sensitive value)

  ports {
    port      = 80
    protocol = "TCP"
  }
}
}

# azurerm_linux_virtual_machine.linux_vm:
resource "azurerm_linux_virtual_machine" "linux_vm" {
  admin_username          = "scs-yourHumberID"
  allow_extension_operations = true
  availability_set_id     = null
  bypass_platform_safety_checks_on_user_schedule_enabled = false
  capacity_reservation_group_id = null
  computer_name           = "lab03s2-db1-u-vm1"
  dedicated_host_group_id = null
  dedicated_host_id        = null
  disable_password_authentication = true
  disk_controller_type     = "SCSI"
  edge_zone                = null
  encryption_at_host_enabled = false
  eviction_policy          = null
}

```

```

encryption_at_host_enabled          = false
eviction_policy                    = null
extensions_time_budget            = "PT1H30M"
id                                = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Compute/virtualMachines/lab03s2-db1-u-vm1"
license_type                       = null
location                          = "eastus"
max_bid_price                     = -1
name                             = "lab03s2-db1-u-vm1"
network_interface_ids            = [
  "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkInterfaces/lab03s2-db1-u-vm1-nic",
]
patch_assessment_mode            = "ImageDefault"
patch_mode                         = "ImageDefault"
platform_fault_domain            = -1
priority                          = "Regular"
private_ip_address                = "10.0.1.4"
private_ip_addresses              = [
  "10.0.1.4",
]
provision_vm_agent               = true
proximity_placement_group_id     = null
public_ip_address                = "172.191.206.6"
public_ip_addresses              = [
  "172.191.206.6",
]
reboot_setting                   = null
resource_group_name              = "lab02-rg"
secure_boot_enabled              = false
size                             = "Standard_B1s"
source_image_id                  = null
tags                 = {}
  "environment" = "lab"
}
user_data                         = null
virtual_machine_id                = "91e3031c-a6f7-48fb-b4b2-e51d1db06d26"
virtual_machine_scale_set_id      = null
vm_agent_platform_updates_enabled = false
vtppm_enabled                     = false
zone                            = null

```

```

admin_ssh_key {
    public_key = <<-EOT
        ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQADQIPRPH9LNbULUqkQgEu3jA1EDjIIUVRA4tHIMe3ZsGi2K0bpivT0f0/081yrtSy19uGW+Xs9wk6XogP4yz54GCF85QyTD2ta3uY1j1/g6
9LfnU5py0sTSdUsP/F+BLEX0vRPNmaC2QmEt4pFhmNVFLJmFj3fx2WRHgBgaYB0CTK6tyUDLa50T7BgDQ0qHE2ASrxwaxSGX3MzDMEQYD04GqvB3m6mfSwUHTWHz2ut9i3G1SRe4u4VeN
UBP91Qnx4oCSH2j8jFRSN5wOK8Rlvj87xHffwx1SXQe0792Dec71Ty5v/09Rfu13vYalrL9bR7McCsZnMGtwHFFPM5MhL61g+VTI89+1M5MFldqvMdH9SLw+rhG+HDVmh9nC8knkNY00Ay/U1ynRML2
hBuWfLbfw1cP37vqzbXaih+8gk3nGw550fbilNS6qjcCmvr1Fu7i+5EtFOKxzbyvt0kypv1453n4DjM1YzpG3susx7kxxU7RLBwInbuAH0lzbobfRNg0w5wYMMTz1N3J9X6fWB00lu/esdkS0Ag
VBa4rCNPcOSk2HNj4gv1pf+kRX6F0v+uMnUbh2ifYAxrGJHochHY1PSvgN8zq+VWAS5JBkj7i9DrHJY+nWBzE6V8Dk2d7LZugIub04HnXyw== sscs@DESKTOP-MM9IRH
    EOT
    username  = "scs-yourHumberID"
}

os_disk {
    caching          = "ReadWrite"
    disk_encryption_set_id = null
    disk_size_gb    = 32
    name            = "lab03s2-dbl-u-vm1-osdisk"
    secure_vm_disk_encryption_set_id = null
    security_encryption_type = null
    storage_account_type = "Premium_LRS"
    write_accelerator_enabled = false
}

source_image_reference {
    offer      = "UbuntuServer"
    publisher  = "Canonical"
    sku        = "18.04-lts-gen2"
    version    = "latest"
}

}

# azurerm_network_interface.linux_nic:
resource "azurerm_network_interface" "linux_nic" {
    accelerated_networking_enabled = false
    applied_dns_servers          = []
    auxiliary_mode                = null
    auxiliary_sku                 = null
    dns_servers                   = []
    edge_zone                     = null
    enable_accelerated_networking = false
}

```

```

edge_zone          = null
enable_accelerated_networking = false
enable_ip_forwarding = false
id                = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkInterfaces/lab03s2-dbl-u-vm1-nic"
internal_dns_name_label = null
internal_domain_name_suffix = "qp3lboqndmepcq4h2hq4nvfkdb.internal.cloudapp.net"
ip_forwarding_enabled = false
location          = "eastus"
mac_address       = null
name              = "lab03s2-dbl-u-vm1-nic"
private_ip_address = "10.0.1.4"
private_ip_addresses = [
    "10.0.1.4",
]
resource_group_name = "lab02-rg"
tags               = {}
virtual_machine_id = null

ip_configuration {
    gateway_load_balancer_frontend_ip_configuration_id = null
    name          = "Lab03s2-dbl-u-vm1-ipconfig1"
    primary       = true
    private_ip_address = "10.0.1.4"
    private_ip_address_allocation = "Dynamic"
    private_ip_address_version = "IPv4"
    public_ip_address_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/publicIPAddresses/lab03s2-dbl-u-vm1-pip"
    subnet_id     = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
}
}

# azurerm_network_security_group.lab02_ns1:
resource "azurerm_network_security_group" "lab02_ns1" {
    id                = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-ns1"
    location          = "eastus"
    name              = "lab02-ns1"
    resource_group_name = "lab02-rg"
    security_rule     = [
        {
            access          = "Allow"
            description     = null
            destination_address_prefix = "*"
            destination_address_prefixes = []
            destination_application_security_group_ids = []
            destination_port_range = "22"
            destination_port_ranges = []
            direction       = "Inbound"
            name            = "SSH"
            priority        = 100
            protocol        = "Tcp"
            source_address_prefix = "*"
            source_address_prefixes = []
            source_application_security_group_ids = []
            source_port_range = "*"
            source_port_ranges = []
        },
        {
            tags           = {}
        }
    ]
}

# azurerm_network_security_group.lab02_ns2:
resource "azurerm_network_security_group" "lab02_ns2" {
    id                = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-ns2"
    location          = "eastus"
    name              = "lab02-ns2"
    resource_group_name = "lab02-rg"
    security_rule     = [
        {
            access          = "Allow"
            description     = null
            destination_address_prefix = "*"
            destination_address_prefixes = []
            destination_application_security_group_ids = []
            destination_port_range = "3389"
            destination_port_ranges = []
            direction       = "Inbound"
        }
    ]
}

```

```

resource_group_name = "lab02-rg"
security_rule      = [
    {
        access          = "Allow"
        description     = null
        destination_address_prefix = "*"
        destination_address_prefixes = []
        destination_application_security_group_ids = []
        destination_port_range = "22"
        destination_port_ranges = []
        direction       = "Inbound"
        name            = "SSH"
        priority        = 100
        protocol        = "Tcp"
        source_address_prefix = "*"
        source_address_prefixes = []
        source_application_security_group_ids = []
        source_port_range = "*"
        source_port_ranges = []
    },
    {
        tags           = {}
    }
]

# azurerm_network_security_group.lab02_ns2:
resource "azurerm_network_security_group" "lab02_ns2" {
    id                = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-ns2"
    location          = "eastus"
    name              = "lab02-ns2"
    resource_group_name = "lab02-rg"
    security_rule     = [
        {
            access          = "Allow"
            description     = null
            destination_address_prefix = "*"
            destination_address_prefixes = []
            destination_application_security_group_ids = []
            destination_port_range = "3389"
            destination_port_ranges = []
            direction       = "Inbound"
        }
    ]
}

```

```

        destination_port_ranges = []
        direction = "Inbound"
        name = "rule1"
        priority = 100
        protocol = "Tcp"
        source_address_prefix = "*"
        source_address_prefixes = []
        source_application_security_group_ids = []
        source_port_range = "*"
        source_port_ranges = []

    },
    {
        access = "Allow"
        description = null
        destination_address_prefix = "*"
        destination_address_prefixes = []
        destination_application_security_group_ids = []
        destination_port_range = "5985"
        destination_port_ranges = []
        direction = "Inbound"
        name = "rule2"
        priority = 200
        protocol = "Tcp"
        source_address_prefix = "*"
        source_address_prefixes = []
        source_application_security_group_ids = []
        source_port_range = "*"
        source_port_ranges = []
    },
],
tags = {}
}

# azurerm_public_ip.linux_pip:
resource "azurerm_public_ip" "linux_pip" {
    allocation_method = "Dynamic"
    ddos_protection_mode = "VirtualNetworkInherited"
    edge_zone = null
    id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/publicIPAddresses/lab03s2-dbl-u-vm1-pip"
    idle_timeout_in_minutes = 4
}

```

```

03s2-dbl-u-vm1-pip"
idle_timeout_in_minutes = 4
ip_address = null
ip_tags = {}
ip_version = "IPv4"
location = "eastus"
name = "lab03s2-dbl-u-vm1-pip"
resource_group_name = "lab02-rg"
sku = "Basic"
sku_tier = "Regional"
tags = {}
zones = []

}

# azurerm_resource_group.lab02_rg:
resource "azurerm_resource_group" "lab02_rg" {
    id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg"
    location = "eastus"
    managed_by = null
    name = "lab02-rg"
    tags = {}
}

# azurerm_subnet.lab02_subnet1:
resource "azurerm_subnet" "lab02_subnet1" {
    address_prefixes = [
        "10.0.1.0/24",
    ]
    default_outbound_access_enabled = true
    enforce_private_link_endpoint_network_policies = false
    enforce_private_link_service_network_policies = false
    id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
    name = "lab02-subnet1"
    private_endpoint_network_policies = "Enabled"
    private_endpoint_network_policies_enabled = true
    private_link_service_network_policies_enabled = true
    resource_group_name = "lab02-rg"
    service_endpoint_policy_ids = []
    service_endpoints = []
    virtual_network_name = "lab02-vnet"
}

```

```

    service_endpoints = []
    virtual_network_name = "lab02-vnet"
}

# azurerm_subnet.lab02_subnet2:
resource "azurerm_subnet" "lab02_subnet2" {
    address_prefixes = [
        "10.0.2.0/24",
    ]
    default_outbound_access_enabled = true
    enforce_private_link_endpoint_network_policies = false
    enforce_private_link_service_network_policies = false
    id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2"
    name = "lab02-subnet2"
    private_endpoint_network_policies = "Enabled"
    private_endpoint_network_policies_enabled = true
    private_link_service_network_policies_enabled = true
    resource_group_name = "lab02-rg"
    service_endpoint_policy_ids = []
    service_endpoints = []
    virtual_network_name = "lab02-vnet"
}

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc1:
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc1" {
    id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
    network_security_group_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
    subnet_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
}

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc2:
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc2" {
    id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2"
    network_security_group_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2"
    subnet_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
}
```

```

        ],
        "bgp_community": null,
        "dns_servers": [],
        "edge_zone": null,
        "flow_timeout_in_minutes": 0,
        "guid": "bab0fa83-1b0d-4708-8a1e-3f0f0f36a553",
        "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet",
        "location": "eastus",
        "name": "lab02-vnet",
        "resource_group_name": "lab02-rg",
        "subnet": [
            {
                "address_prefix": "10.0.1.0/24",
                "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1",
                "name": "lab02-subnet1",
                "security_group": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
            },
            {
                "address_prefix": "10.0.2.0/24",
                "id": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2",
                "name": "lab02-subnet2",
                "security_group": "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2"
            }
        ],
        "tags": {}
    }

    # random_integer.rand:
    resource "random_integer" "rand" {
        id      = "72390"
        max    = 99999
        min    = 10000
        result = 72390
    }
}

C:\Users\SCS\lab03s2>

```

Part 9: Confirm Resource Creation in Azure

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

SCREENSHOT (capture the Azure Portal showing resources).

Name	Subscription	Location
lab02-rg	Azure for Students	East US
NetworkWatcherRG	Azure for Students	East US

Name	Type	Location
lab02-container-72390	Container instances	East US
lab02-nsg1	Network security group	East US
lab02-nsg2	Network security group	East US
lab02-vnet	Virtual network	East US
lab03s2-db1-u-vm1	Virtual machine	East US
lab03s2-db1-u-vm1-nic	Network Interface	East US
lab03s2-db1-u-vm1-osdisk	Disk	East US
lab03s2-db1-u-vm1-pip	Public IP address	East US

The screenshot shows the Microsoft Azure portal interface for managing subnets. The left sidebar navigation includes 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', 'Resource visualizer', 'Settings' (with 'Address space' and 'Connected devices' options), and 'Subnets' (which is currently selected). The main content area displays a table of subnets with columns: Name, IPv4, IPv6, Available IPs, Delegated to, Security group, and Route table. Two subnets are listed: 'lab02-subnet1' (IPv4 10.0.1.0/24, Delegated to 'lab02-nsg1') and 'lab02-subnet2' (IPv4 10.0.2.0/24, Delegated to 'lab02-nsg2'). A search bar at the top allows for searching subnets.

The screenshot shows the Microsoft Azure portal interface for managing container instances. The left sidebar navigation includes 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Resource visualizer', 'Settings' (with 'Containers' option selected), 'Identity', 'Properties', 'Locks', 'Monitoring', 'Automation', and 'Help'. The main content area shows a container instance named 'lab02-container-72390'. It provides details such as Resource group (lab02-rg), Status (Running), Location (East US), Subscription (Azure for Students), Subscription ID, and Tags (environment: lab). It also displays performance metrics for CPU (millicores) and Memory over time. A warning message at the top notes a recent pull rate limit on Docker Hub images.

Part 10: Destroy All Resources and Verify

19. Destroy all resources:

```
terraform destroy
```

- Type yes when prompted.

20. Verify deletion:

```
terraform state list
terraform show
```

SCREENSHOT (capture both outputs).

```
Destroy complete! Resources: 13 destroyed.

C:\Users\SCS\lab03s2>terraform state list

C:\Users\SCS\lab03s2>terraform show
The state file is empty. No resources are represented.

C:\Users\SCS\lab03s2>
```

Section 3

Objectives:

- Use configuration from Section 2.
- Define a locals block to assign tags to resources.
- Validate, deploy, expand, analyze, and destroy infrastructure.

Part 1: Prepare for the Lab

1. Create a directory called lab03s3 in your home directory:
2. Copy lab03s1.tf, vars03s1.tf, lab03s2.tf, vars03s2.tf, and providers.tf into lab03s3 directory:
3. Change into the lab03s3 directory:
4. Rename lab03s2.tf as lab03s3.tf and vars03s2.tf as vars03s3.tf:

Part 2: Update vars03s3.tf File

5. Open vars03s3.tf in a text editor and define a locals block at the beginning with the following values. Enclose values in double quotation marks:

- a. Name: "Terraform-Class".
- b. Project: "Learning".
- c. ContactEmail: "youremailaddress".
- d. Environment: "Lab".

SCREENSHOT of vars03s3.tf (capture the full content in your editor).

```

File Edit View

locals {
  Name      = "Terraform-Class"
  Project   = "Learning"
  ContactEmail = "youremailaddress"
  Environment = "Lab"
}

variable "linux_name" {
  description = "Name of the Linux VM"
  type        = string
  default     = "lab03s2-db1-u-vm1"
}

variable "vm_size" {
  description = "The size of the virtual machine"
  type        = string
  default     = "Standard_B1s"
}

variable "admin_username" {
  description = "Admin user name for the VM"
  type        = string
  default     = "scs-yourHumberID" # Replace with your actual username (e.g., "scs-12345")
}

variable "public_key" {
  description = "The path to the public key for SSH authentication"
  type        = string
  default     = "C:/Users/SCS/.ssh/id_rsa.pub"
}

variable "os_disk_storage_account_type" {
  description = "Storage account type for the OS disk"
  type        = string
  default     = "Premium_LRS"
}

variable "os_disk_size" {
  description = "Size of the OS disk in GB"
  type        = number
}

```

```

variable "os_disk_size" {
  description = "Size of the OS disk in GB"
  type        = number
  default     = 32
}

variable "os_disk_caching" {
  description = "Caching mode for the OS disk"
  type        = string
  default     = "ReadWrite"
}

variable "ubuntu_publisher" {
  description = "Publisher for the Ubuntu image"
  type        = string
  default     = "Canonical"
}

variable "ubuntu_offer" {
  description = "Offer for the Ubuntu image"
  type        = string
  default     = "UbuntuServer"
}

variable "ubuntu_sku" {
  description = "SKU for the Ubuntu image"
  type        = string
  default     = "19.04"
}

variable "ubuntu_version" {
  description = "Version for the Ubuntu image"
  type        = string
  default     = "latest"
}

```

Part 3: Update lab03s3.tf File

6. Open lab03s3.tf in a text editor and add tags to network interface, public IP, and virtual machine resource blocks using the locals values.

SCREENSHOT of lab03s3.tf (capture the full content in your editor).

File Edit View

```
# Public IP Address
resource "azurerm_public_ip" "linux_pip" {
    name          = "${var.linux_name}-pip"
    location      = var.location
    resource_group_name = var.resource_group_name
    allocation_method = "Dynamic"

    tags = {
        Name      = local.Name
        Project   = local.Project
        ContactEmail = local.ContactEmail
        Environment = local.Environment
    }
}

# Network Interface
resource "azurerm_network_interface" "linux_nic" {
    name          = "${var.linux_name}-nic"
    location      = var.location
    resource_group_name = var.resource_group_name

    ip_configuration {
        name          = "${var.linux_name}-ipconfig1"
        subnet_id     = azurerm_subnet.lab02_subnet1.id
        private_ip_address_allocation = "Dynamic"
        public_ip_address_id       = azurerm_public_ip.linux_pip.id
    }

    tags = {
        Name      = local.Name
        Project   = local.Project
        ContactEmail = local.ContactEmail
        Environment = local.Environment
    }
}

# Linux Virtual Machine
resource "azurerm_linux_virtual_machine" "linux_vm" {
    name          = var.linux_name
    location      = var.location
```

```

# Linux Virtual Machine
resource "azurerm_linux_virtual_machine" "linux_vm" {
    name          = var.linux_name
    location      = var.location
    resource_group_name = var.resource_group_name
    size          = var.vm_size
    admin_username = var.admin_username

    network_interface_ids = [
        azurerm_network_interface.linux_nic.id
    ]

    admin_ssh_key {
        username    = var.admin_username
        public_key = file(var.public_key)
    }

    os_disk {
        name          = "${var.linux_name}-osdisk"
        caching       = var.os_disk_caching
        storage_account_type = var.os_disk_storage_account_type
        disk_size_gb   = var.os_disk_size
    }

    #  Updated image configuration for eastus
    source_image_reference {
        publisher = "Canonical"
        offer     = "UbuntuServer"
        sku       = "18_04-lts-gen2"
        version   = "latest"
    }

    tags = {
        Name          = local.Name
        Project       = local.Project
        ContactEmail = local.ContactEmail
        Environment   = local.Environment
    }
}

```

Part 4: Initialize Terraform

7. Initialize Terraform to download plug-ins as required:

Part 5: Validate Configuration

8. Validate the configuration to ensure no errors or typos:

9. Fix any issues in the Terraform files if reported.

10. Re-run validation until no errors are reported:

SCREENSHOT (capture the successful validation output).

```
C:\Users\SCS\lab03s3>terraform init
Initializing the backend...
Initializing provider plugins...
- Finding hashicorp/random versions matching "~> 3.0"...
- Finding hashicorp/azurerm versions matching "~> 3.0"...
- Installing hashicorp/random v3.7.1...
- Installed hashicorp/random v3.7.1 (signed by HashiCorp)
- Installing hashicorp/azurerm v3.117.1...
- Installed hashicorp/azurerm v3.117.1 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

C:\Users\SCS\lab03s3>terraform validate
Success! The configuration is valid.
```

Part 6: Run Simulation

11. Perform a dry run:

```
terraform plan
```

12. Review output and ensure all configuration meets requirements. Observe resources with +, -, or -/+ signs.

13. Fix any issues in the Terraform files if reported.

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

Part 7: Deploy Infrastructure

15. Deploy the infrastructure and monitor progress:

```
terraform apply
```

- Type yes when prompted.

Part 8: Get Information from Terraform State

17. View and analyze state information:

```
terraform state list
```

```
terraform show
```

SCREENSHOT (capture both outputs).

```
Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
```

```
C:\Users\SCS\lab03s3>terraform state list
azurerm_container_group.lab02_container
azurerm_linux_virtual_machine.linux_vm
azurerm_network_interface.linux_nic
azurerm_network_security_group.lab02_nsg1
azurerm_network_security_group.lab02_nsg2
azurerm_public_ip.linux_pip
azurerm_resource_group.lab02_rg
azurerm_subnet.lab02_subnet1
azurerm_subnet.lab02_subnet2
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1
azurerm_subnet_network_security_group_association.lab02_nsg_assoc2
azurerm_virtual_network.lab02_vnet
random_integer.rand
```

```
C:\Users\SCS\lab03s3>terraform show
# azurerm_container_group.lab02_container:
resource "azurerm_container_group" "lab02_container" {
  dns_name_label          = "lab02-container-75002"
  dns_name_label_reuse_policy = "Unsecure"
  exposed_port {
    port      = 80
    protocol = "TCP"
  },
  fqdn                  = "lab02-container-75002.eastus.azurecontainer.io"
  id                    = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/contain
erGroups/lab02-container-75002"
  ip_address            = "20.242.220.80"
  ip_address_type       = "Public"
  location              = "eastus"
  name                  = "lab02-container-75002"
  os_type                = "Linux"
  priority              = null
  resource_group_name   = "lab02-rg"
  restart_policy         = "Always"
  sku                   = "Standard"
  subnet_ids            = []
  tags {
    "environment" = "lab"
  }
  zones                = []
}

container {
  commands            = []
  cpu                 = 1
  cpu_limit           = 0
  environment_variables = {}
  image               = "mcr.microsoft.com/azuredocs/aci-helloworld"
  memory              = 1.5
  memory_limit         = 0
  name                = "helloworld"
}
```

```
environment_variables      = {}
image                     = "mcr.microsoft.com/azuredocs/aci-helloworld"
memory                   = 1.5
memory_limit              = 0
name                      = "helloworld"
secure_environment_variables = (sensitive value)

ports {
  port      = 80
  protocol = "TCP"
}
}

# azurerm_linux_virtual_machine.linux_vm:
resource "azurerm_linux_virtual_machine" "linux_vm" {
  admin_username          = "scs-yourHumberID"
  allow_extension_operations = true
  availability_set_id     = null
  bypass_platform_safety_checks_on_user_schedule_enabled = false
  capacity_reservation_group_id = null
  computer_name            = "lab03s2-db1-u-vm1"
  dedicated_host_group_id = null
  dedicated_host_id        = null
  disable_password_authentication = true
  disk_controller_type     = "SCSI"
  edge_zone                = null
  encryption_at_host_enabled = false
  eviction_policy           = null
  extensions_time_budget   = "PT1H30M"
  id                       = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft
Compute/virtualMachines/lab03s2-db1-u-vm1"
  license_type              = null
  location                  = "eastus"
  max_bid_price             = -1
  name                      = "lab03s2-db1-u-vm1"
  network_interface_ids     = [
    "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkInterfaces/lab03s2-db1-u-vm1-nic",
  ]
  patch_assessment_mode     = "ImageDefault"
  patch_mode                = "ImageDefault"
```

```

network_interface_ids = [
    "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkInterfaces/lab03s2-db1-u-vml-nic",
]
patch_assessment_mode = "ImageDefault"
patch_mode = "ImageDefault"
platform_fault_domain = -1
priority = "Regular"
private_ip_address = "10.0.1.4"
private_ip_addresses = [
    "10.0.1.4",
]
provision_vm_agent = true
proximity_placement_group_id = null
public_ip_address = "172.190.144.44"
public_ip_addresses = [
    "172.190.144.44",
]
reboot_setting = null
resource_group_name = "lab02-rg"
secure_boot_enabled = false
size = "Standard_B1s"
source_image_id = null
tags = {
    "ContactEmail" = "youremailaddress"
    "Environment" = "Lab"
    "Name" = "Terraform-Class"
    "Project" = "Learning"
}
user_data = "#!/bin/bash
ssh rsa AAAAB3NzaC1yc2EAAAQABAAQADIPRPH9lNbUvUoKgEu3jA1EdjIIUVRA4tHIMe3ZsG12K0bpIV70F0/081yrtS19uGw+Xs9wK6XogPhy5HGCf85QyTD2ta3uY1j1/g6
9LFLnu5py0sUTsD5sRL/+fbLE80VNPMPaCQmEt4pFhmNVFLjmF3pf2eWRHgBgaYB0CTK6htyUDs50T7BgpdQ0Qqhe2ASxwASGX3M2dHEQYD04QvgB3m6DmFSwUHTWTh5zut91L3G1SRE4u4vEN
UBP9lQngnXoC8L5s5RvJ87kHffwvIyA7L9bRG7MsZnMgtw66792Dec717y5kW+7gW+h+HfDVMn0c8nLkY008Ay/U1ynRMLe
BuWfLbfwicP37qvqzbxAik+8gk3GwWSB2FB1lNS6gRjcCmvrIu7Ui0+5cETfB0Kzbvvt0kypv1453N4DM1YzpG35us7hxU7LBWmibuaHB0Lzobf+mRWg0sWYMMTz1N3J9xFWBmLu/esdk50Ag
admin_ssh_key {
    public_key = <<-EOT
        ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQADIPRPH9lNbUvUoKgEu3jA1EdjIIUVRA4tHIMe3ZsG12K0bpIV70F0/081yrtS19uGw+Xs9wK6XogPhy5HGCf85QyTD2ta3uY1j1/g6
9LFLnu5py0sUTsD5sRL/+fbLE80VNPMPaCQmEt4pFhmNVFLjmF3pf2eWRHgBgaYB0CTK6htyUDs50T7BgpdQ0Qqhe2ASxwASGX3M2dHEQYD04QvgB3m6DmFSwUHTWTh5zut91L3G1SRE4u4vEN
UBP9lQngnXoC8L5s5RvJ87kHffwvIyA7L9bRG7MsZnMgtw66792Dec717y5kW+7gW+h+HfDVMn0c8nLkY008Ay/U1ynRMLe
BuWfLbfwicP37qvqzbxAik+8gk3GwWSB2FB1lNS6gRjcCmvrIu7Ui0+5cETfB0Kzbvvt0kypv1453N4DM1YzpG35us7hxU7LBWmibuaHB0Lzobf+mRWg0sWYMMTz1N3J9xFWBmLu/esdk50Ag

```

```

    EOT
    username      = "scs-yourHuberID"
}

os_disk {
    caching           = "ReadWrite"
    disk_encryption_set_id = null
    disk_size_gb     = 32
    name              = "lab03s2-db1-u-vm1-osdisk"
    secure_vm_disk_encryption_set_id = null
    security_encryption_type   = null
    storage_account_type       = "Premium_LRS"
    write_accelerator_enabled = false
}

source_image_reference {
    offer          = "UbuntuServer"
    publisher      = "Canonical"
    sku            = "18.04-lts-gen2"
    version        = "latest"
}

# azurerm_network_interface.linux_nic:
resource "azurerm_network_interface" "linux_nic" {
    accelerated_networking_enabled = false
    applied_dns_servers          = []
    auxiliary_mode                = null
    auxiliary_sku                 = null
    dns_servers                   = []
    edge_zone                     = null
    enable_accelerated_networking = false
    enable_ip_forwarding          = false
    id                            = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkInterfaces/lab03s2-db1-u-vm1-nic"
    internal_dns_name_label       = null
    internal_domain_name_suffix  = "1ff3edlhig3uhid15gtixxddoa.bx.internal.cloudapp.net"
    ip_forwarding_enabled         = false
}

```

```

ces/lab03s2-db1-u-vml-nic"
  internal_dns_name_label      = null
  internal_domain_name_suffix  = "1ff3edlhig3uhid15gtixxddoa.bx.internal.cloudapp.net"
  ip_forwarding_enabled        = false
  location                     = "eastus"
  mac_address                  = null
  name                         = "lab03s2-db1-u-vml-nic"
  private_ip_address           = "10.0.1.4"
  private_ip_addresses          = [
    "10.0.1.4",
  ]
  resource_group_name          = "lab02-rg"
  tags                         = {
    "ContactEmail" = "youremailaddress"
    "Environment"  = "Lab"
    "Name"         = "Terraform-Class"
    "Project"      = "Learning"
  }
  virtual_machine_id           = null

  ip_configuration {
    gateway_load_balancer_frontend_ip_configuration_id = null
    name                      = "lab03s2-db1-u-vml-ipconfig1"
    primary                   = true
    private_ip_address         = "10.0.1.4"
    private_ip_address_allocation = "Dynamic"
    private_ip_address_version = "IPv4"
    public_ip_address_id       = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft
t.Network/publicIPAddresses/lab03s2-db1-u-vml-pip"
    subnet_id                 = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft
t.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
  }
}

# azurerm_network_security_group.lab02_nsg1:
resource "azurerm_network_security_group" "lab02_nsg1" {
  id              = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab
02-nsg1"
  location        = "eastus"
  name            = "lab02-nsg1"
}

```

```

location      = "eastus"
name         = "lab02-nsg1"
resource_group_name = "lab02-rg"
security_rule   = [
  {
    access           = "Allow"
    description     = null
    destination_address_prefix = "*"
    destination_address_prefixes = []
    destination_application_security_group_ids = []
    destination_port_range = "22"
    destination_port_ranges = []
    direction        = "Inbound"
    name            = "SSH"
    priority        = 100
    protocol        = "Tcp"
    source_address_prefix = "*"
    source_address_prefixes = []
    source_application_security_group_ids = []
    source_port_range = "*"
    source_port_ranges = []
  },
]
tags          = {}

# azurerm_network_security_group.lab02_nsg2:
resource "azurerm_network_security_group" "lab02_nsg2" {
  id          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2"
  location      = "eastus"
  name         = "lab02-nsg2"
  resource_group_name = "lab02-rg"
  security_rule   = [
    {
      access           = "Allow"
      description     = null
      destination_address_prefix = "*"
      destination_address_prefixes = []
      destination_application_security_group_ids = []
      destination_port_range = "3389"
      destination_port_ranges = []
      direction        = "Inbound"
      name            = "rule1"
      priority        = 100
      protocol        = "Tcp"
      source_address_prefix = "*"
      source_address_prefixes = []
      source_application_security_group_ids = []
      source_port_range = "*"
      source_port_ranges = []
    },
    {
      access           = "Allow"
      description     = null
      destination_address_prefix = "*"
      destination_address_prefixes = []
      destination_application_security_group_ids = []
      destination_port_range = "5985"
      destination_port_ranges = []
      direction        = "Inbound"
      name            = "rule2"
      priority        = 200
      protocol        = "Tcp"
      source_address_prefix = "*"
      source_address_prefixes = []
      source_application_security_group_ids = []
      source_port_range = "*"
      source_port_ranges = []
    },
  ]
}

```

```

security_rule      = [
  {
    access           = "Allow"
    description     = null
    destination_address_prefix = "*"
    destination_address_prefixes = []
    destination_application_security_group_ids = []
    destination_port_range = "3389"
    destination_port_ranges = []
    direction        = "Inbound"
    name            = "rule1"
    priority        = 100
    protocol        = "Tcp"
    source_address_prefix = "*"
    source_address_prefixes = []
    source_application_security_group_ids = []
    source_port_range = "*"
    source_port_ranges = []
  },
  {
    access           = "Allow"
    description     = null
    destination_address_prefix = "*"
    destination_address_prefixes = []
    destination_application_security_group_ids = []
    destination_port_range = "5985"
    destination_port_ranges = []
    direction        = "Inbound"
    name            = "rule2"
    priority        = 200
    protocol        = "Tcp"
    source_address_prefix = "*"
    source_address_prefixes = []
    source_application_security_group_ids = []
    source_port_range = "*"
    source_port_ranges = []
  },
]
tags          = {}


```

```

        ],
        tags          = {}
    }

# azurerm_public_ip.azure_ip:
resource "azurerm_public_ip" "linux_pip" {
    allocation_method      = "Dynamic"
    ddos_protection_mode  = "VirtualNetworkInherited"
    edge_zone              = null
    id                     = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/publicIPAddresses/lab03s2-dbl1-u-vm1-pip"
    idle_timeout_in_minutes = 4
    ip_address             = null
    ip_version              = "IPv4"
    location                = "eastus"
    name                    = "lab03s2-dbl1-u-vm1-pip"
    resource_group_name     = "lab02-rg"
    sku                     = "Basic"
    sku_tier                = "Regional"
    tags {
        "ContactEmail" = "youremailaddress"
        "Environment"  = "Lab"
        "Name"         = "Terraform-Class"
        "Project"      = "Learning"
    }
}

# azurerm_resource_group.lab02_rg:
resource "azurerm_resource_group" "lab02_rg" {
    id           = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg"
    location     = "eastus"
    managed_by   = null
    name         = "lab02-rg"
    tags          = {}
}

# azurerm_subnet.lab02_subnet1:
resource "azurerm_subnet" "lab02_subnet1" {
    address_prefixes      = [
        "10.0.1.0/24",
    ]
}

# azurerm_subnet.lab02_subnet1:
resource "azurerm_subnet" "lab02_subnet1" {
    address_prefixes      = [
        "10.0.1.0/24",
    ]
    default_outbound_access_enabled = true
    enforce_private_link_endpoint_network_policies = false
    enforce_private_link_service_network_policies = false
    id                     = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
    name                  = "lab02-subnet1"
    private_endpoint_network_policies = "Enabled"
    private_endpoint_network_policies_enabled = true
    private_link_service_network_policies = true
    private_link_service_network_policies_enabled = true
    resource_group_name     = "lab02-rg"
    service_endpoint_policy_ids = []
    service_endpoints       = []
    virtual_network_name     = "lab02-vnet"
}

# azurerm_subnet.lab02_subnet2:
resource "azurerm_subnet" "lab02_subnet2" {
    address_prefixes      = [
        "10.0.2.0/24",
    ]
    default_outbound_access_enabled = true
    enforce_private_link_endpoint_network_policies = false
    enforce_private_link_service_network_policies = false
    id                     = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2"
    name                  = "lab02-subnet2"
    private_endpoint_network_policies = "Enabled"
    private_endpoint_network_policies_enabled = true
    private_link_service_network_policies = true
    private_link_service_network_policies_enabled = true
    resource_group_name     = "lab02-rg"
    service_endpoint_policy_ids = []
    service_endpoints       = []
    virtual_network_name     = "lab02-vnet"
}

```

```

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc1:
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc1" {
    id           = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
    network_security_group_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
    subnet_id      = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
}

# azurerm_subnet_network_security_group_association.lab02_nsg_assoc2:
resource "azurerm_subnet_network_security_group_association" "lab02_nsg_assoc2" {
    id           = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2"
    network_security_group_id = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2"
    subnet_id      = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/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",
    ]
    bgp_community       = null
    dns_servers         = []
    edge_zone           = null
    flow_timeout_in_minutes = 0
    guid                = "0dd24bd9-4167-43bb-a07b-f9a68bdc6370"
    id                 = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet"
    location            = "eastus"
    name                = "lab02-vnet"
    resource_group_name = "lab02-rg"
    subnet {
        {
            address_prefix = "10.0.1.0/24"
            id           = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet"
        }
    }
}

```

```

]
  bgp_community      = null
  dns_servers        = []
  edge_zone          = null
  flow_timeout_in_minutes = 0
  guid               = "0dd24bd9-4167-43bb-a07b-f9a68bcd6370"
  id                 = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet"
  location           = "eastus"
  name               = "lab02-vnet"
  resource_group_name = "lab02-rg"
  subnet              = [
    {
      address_prefix = "10.0.1.0/24"
      id            = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1"
      name           = "lab02-subnet1"
      security_group = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg1"
    },
    {
      address_prefix = "10.0.2.0/24"
      id            = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2"
      name           = "lab02-subnet2"
      security_group = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkSecurityGroups/lab02-nsg2"
    }
  ],
  tags                = {}
}

# random_integer.rand:
resource "random_integer" "rand" {
  id      = "75002"
  max    = 99999
  min    = 10000
  result = 75002
}

```

C:\Users\SCS\lab03s3>

Part 9: Confirm Resource Creation in Azure

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

SCREENSHOT (capture the Azure Portal showing resources and tags).

Name	Subscription	Location
lab02-rg	... Azure for Students	East US
NetworkWatcherRG	... Azure for Students	East US

Showing 1 - 2 of 2. Display count: 10

You are viewing a new version of Browse experience. Some features may be missing. Click here to access the old experience.

Name ↑

- lab02-rg
- NetworkWatcherRG

Showing 1 to 8 of 8 records. □ Show hidden types ⓘ

Name	Type	Location
lab02-container-75002	Container instances	East US
lab02-nsg	Network security group	East US
lab02-nsg2	Network security group	East US
lab02-vnet	Virtual network	East US
lab03s2-db1-u-vm1-nic	Virtual machine	East US
lab03s2-db1-u-vm1-osdisk	Network Interface	East US
lab03s2-db1-u-vm1-pip	Disk	East US
	Public IP address	East US

< Previous Page 1 of 1 Next >

Give feedback

Part 10: Destroy All Resources and Verify

19. Destroy all resources:

```
terraform destroy
```

- Type yes when prompted.

20. Verify deletion:

```
terraform state list
terraform show
```

SCREENSHOT (capture both outputs).

```
Destroy complete! Resources: 13 destroyed.

C:\Users\SCS\lab03s3>terraform state list

C:\Users\SCS\lab03s3>terraform show
The state file is empty. No resources are represented.

C:\Users\SCS\lab03s3>
```

Section 4

Objectives:

- Use configuration from Section 3.

- Define output blocks to display values.
- Validate, deploy, expand, analyze, and destroy infrastructure.

Part 1: Prepare for the Lab

1. Copy the lab03s3 directory as lab03s4:
2. Change into the lab03s4 directory:
3. Create an empty file called outputs03s4.tf:

Part 2: Update outputs03s4.tf File

4. Open outputs03s4.tf in a text editor and define output blocks to display:
 - a. VM hostname (1 block).
 - b. Private IP address (1 block) and Public IP address (1 block).
 - c. Virtual network name (1 block) and address space (1 block).
 - d. Subnet names (2 blocks) and address spaces (2 blocks).

SCREENSHOT of outputs03s4.tf (capture the full content in your editor).

File Edit View

```
output "vm_hostname" {
    value = azurerm_linux_virtual_machine.linux_vm.name
}

output "private_ip_address" {
    value = azurerm_network_interface.linux_nic.private_ip_address
}

output "public_ip_address" {
    value = azurerm_public_ip.linux_pip.ip_address
}

output "vnet_name" {
    value = azurerm_virtual_network.myvnet.name
}

output "vnet_address_space" {
    value = azurerm_virtual_network.myvnet.address_space
}

output "subnet1_name" {
    value = azurerm_subnet.lab02_subnet1.name
}

output "subnet1_address_prefix" {
    value = azurerm_subnet.lab02_subnet1.address_prefixes
}

output "subnet2_name" {
    value = azurerm_subnet.lab02_subnet2.name
}

output "subnet2_address_prefix" {
    value = azurerm_subnet.lab02_subnet2.address_prefixes
}
```

Part 3: Validate Configuration

5. Validate the configuration to ensure no errors or typos:
6. Fix any issues in the Terraform files if reported.
7. Re-run validation until no errors are reported:

SCREENSHOT (capture the successful validation output).

```
C:\Users\SCS\lab03s4>terraform validate  
Success! The configuration is valid.
```

```
C:\Users\SCS\lab03s4>
```

Part 4: Run Simulation

8. Perform a dry run:

```
terraform plan
```

9. Review output and ensure all configuration meets requirements. Observe resources with +, -, or -/+ signs.
10. Fix any issues in the Terraform files if reported.
11. Redo the dry run until no errors are reported:

Part 5: Deploy Infrastructure

12. Deploy the infrastructure and monitor progress:

```
terraform apply
```

- Type yes when prompted.
13. Confirm output values displayed on the screen at the end of deployment.

Part 6: Get Information from Terraform State

14. View and analyze state information:

```
terraform state list  
terraform show
```

SCREENSHOT (capture both outputs).

```
Apply complete! Resources: 4 added, 0 changed, 0 destroyed.
```

Outputs:

```
private_ip_address = "10.0.1.4"
public_ip_address = ""
subnet1_address_prefix = tolist([
    "10.0.1.0/24",
])
subnet1_name = "lab02-subnet1"
subnet2_address_prefix = tolist([
    "10.0.2.0/24",
])
subnet2_name = "lab02-subnet2"
vm_hostname = "lab03s2-db1-u-vm1"
vnet_address_space = tolist([
    "10.0.0.0/16",
])
vnet_name = "lab03s2-db1-u-vm1-vnet"

C:\Users\SCS\lab03s4>terraform state list
azurerm_container_group.lab02_container
azurerm_linux_virtual_machine.linux_vm
azurerm_network_interface.linux_nic
azurerm_network_security_group.lab02_nsg1
azurerm_network_security_group.lab02_nsg2
azurerm_public_ip.linux_pip
azurerm_resource_group.lab02_rg
azurerm_subnet.lab02_subnet1
azurerm_subnet.lab02_subnet2
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1
azurerm_subnet_network_security_group_association.lab02_nsg_assoc2
azurerm_virtual_network.lab02_vnet
azurerm_virtual_network.myvnet
random_integer.rand
```

```
C:\Users\SCS\lab03s4>terraform show
# azurerm_container_group.lab02_container:
resource "azurerm_container_group" "lab02_container" {
  dns_name_label          = "lab02-container-90197"
  dns_name_label_reuse_policy = "Unsecure"
  exposed_port {
    {
      port      = 80
      protocol = "TCP"
    },
  }
  fqdn                   = "lab02-container-90197.eastus.azurecontainer.io"
  id                     = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/contain
erGroups/lab02-container-90197"
  ip_address              = "20.241.238.233"
  ip_address_type         = "Public"
  location                = "eastus"
  name                   = "lab02-container-90197"
  os_type                 = "Linux"
  priority                = null
  resource_group_name     = "lab02-rg"
  restart_policy          = "Always"
  sku                     = "Standard"
  subnet_ids              = []
  tags {
    "environment" = "lab"
  }
  zones                  = []

  container {
    commands           = []
    cpu                = 1
    cpu_limit          = 0
    environment_variables = {}
    image               = "mcr.microsoft.com/azuredocs/aci-helloworld"
    memory              = 1.5
    memory_limit        = 0
    name                = "helloworld"
    secure_environment_variables = (sensitive value)
  }

  ports {
```

```

    ports {
        port      = 80
        protocol = "TCP"
    }
}

# azurerm_linux_virtual_machine.linux_vm:
resource "azurerm_linux_virtual_machine" "linux_vm" {
    admin_username          = "scs-yourHumberID"
    allow_extension_operations = true
    availability_set_id     = null
    bypass_platform_safety_checks_on_user_schedule_enabled = false
    capacity_reservation_group_id = null
    computer_name           = "lab03s2-db1-u-vm1"
    dedicated_host_group_id = null
    dedicated_host_id       = null
    disable_password_authentication = true
    disk_controller_type    = "SCSI"
    edge_zone               = null
    encryption_at_host_enabled = false
    eviction_policy         = null
    extensions_time_budget = "PT1H30M"
    id                      = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Compute/virtualMachines/lab03s2-db1-u-vm1"
    license_type             = null
    location                = "eastus"
    max_bid_price            = -1
    name                    = "lab03s2-db1-u-vm1"
    network_interface_ids   = [
        "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkInterfaces/lab03s2-db1-u-vm1-nic",
    ]
    patch_assessment_mode   = "ImageDefault"
    patch_mode               = "ImageDefault"
    platform_fault_domain   = "-1"
    priority                = "Regular"
    private_ip_address       = "10.0.1.4"
    private_ip_addresses     = [
        "10.0.1.4",
    ]
}

proximity_placement_group_id          = null
public_ip_address                   = "172.190.195.5"
public_ip_addresses                 = [
    "172.190.195.5",
]
reboot_setting                       = null
resource_group_name                  = "lab02-rg"
secure_boot_enabled                  = false
size                                = "Standard_B1s"
source_image_id                      = null
tags {
    "ContactEmail" = "youremailaddress"
    "Environment"  = "Lab"
    "Name"         = "Terraform-Class"
    "Project"      = "Learning"
}
user_data {
    virtual_machine_id      = "34e11c17-5396-47ef-bedd-22e5aa1beb8"
    virtual_machine_scale_set_id = null
    vm_agent_platform_updates_enabled = false
    vtpm_enabled            = null
    zone                   = null
}

admin_ssh_key {
    public_key = <<-EOT
        ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQACQDIPRPH9LNbULUqkQgEu3jA1EDjIIUVRAUtHIMe3zsGi2K0bpiV70f0/081yrtSy19uGw+Xs9wk6XogP4yz54GCF850yTD2ta3uY1j1/g6
9LfNu5py0sUTSdUs5R/F+8lEX00vRPNPmacC0mEt4pFhmNVfJmF3p3f2wRHgBpxYB OCTh6htyUDLa5077PbgDGQQqhE2ASrxwxaSGX3NMzDMEQYD04GQvpB3m6DmfSwUHTThSz2ut9i3G1SRRe4uUVeH
UBP910gnx4qCSH2j;j8FRSN5w0K8Rlvj87xKHfFwxLSQe60792Dec71Ty5V/09RFU13yArlL9bRG7Mcs2nMgtwhFFPM5MhL6i5+VTI89+lMSMfldqyMdH9StLwVrhG+HDVmhb8nC8knlkNY00Ay/U1ynRML2
hBuWFbfwIcp37/vqzfbXA1h+8gK3nG6WS502FB1n56gRjcCmvr1fU7i0+sEtF0Kxzbyt0Kypv1453n4DjM1YzpG35usx/kxrU7RLBmW1buHB01zobfRWg0w5wMMTz1N3J9X6fWBNoLu/esdk50Ag
Vba4rCNPcOSk2HNj4gvIp4f+kRX6F0v+uMnUbh2iFyAxrGJHochHYfPSvgN8zq+VwASJBkjU7i9DfHJVy+nWBzE6V8DK2d7LZUgiub04HHnXy== scs@DESKTOP-MM91RHU
    EOT
    username   = "scs-yourHumberID"
}

os_disk {
    caching          = "ReadWrite"
    disk_encryption_set_id = null
    disk_size_gb    = 32
    name             = "lab03s2-db1-u-vm1-osdisk"
    secure_vm_disk_encryption_set_id = null
    security_encryption_type = null
}

```

```

s2-db1-u-vm1-vnet"
  location          = "eastus"
  name             = "lab03s2-db1-u-vm1-vnet"
  resource_group_name = "lab02-rg"
  subnet           = []
  tags             = {
    "ContactEmail" = "youremailaddress"
    "Environment"  = "Lab"
    "Name"         = "Terraform-Class"
    "Project"      = "Learning"
  }
}

# random_integer.rand:
resource "random_integer" "rand" {
  id      = "90197"
  max    = 99999
  min    = 10000
  result = 90197
}

```

Outputs:

```

private_ip_address = "10.0.1.4"
public_ip_address = null
subnet1_address_prefix = [
  "10.0.1.0/24",
]
subnet1_name = "lab02-subnet1"
subnet2_address_prefix = [
  "10.0.2.0/24",
]
subnet2_name = "lab02-subnet2"
vm_hostname = "lab03s2-db1-u-vm1"
vnet_address_space = [
  "10.0.0.0/16",
]
vnet_name = "lab03s2-db1-u-vm1-vnet"

```

C:\Users\SCS\lab03s4>

Part 7: Display Output Information

15. Display output information:

terraform output

SCREENSHOT (capture the output).

```
C:\Users\SCS\lab03s4>terraform output
private_ip_address = "10.0.1.4"
public_ip_address = ""
subnet1_address_prefix = tolist([
  "10.0.1.0/24",
])
subnet1_name = "lab02-subnet1"
subnet2_address_prefix = tolist([
  "10.0.2.0/24",
])
subnet2_name = "lab02-subnet2"
vm_hostname = "lab03s2-db1-u-vm1"
vnet_address_space = tolist([
  "10.0.0.0/16",
])
vnet_name = "lab03s2-db1-u-vm1-vnet"
C:\Users\SCS\lab03s4>
```

Part 8: Destroy All Resources and Verify

16. Destroy all resources:

```
terraform destroy
```

- Type yes when prompted.
17. Verify deletion:

```
terraform state list  
terraform show
```

SCREENSHOT (capture both outputs).

```
Destroy complete! Resources: 14 destroyed.  
  
C:\Users\SCS\lab03s4>terraform state list  
  
C:\Users\SCS\lab03s4>terraform show  
The state file is empty. No resources are represented.  
  
C:\Users\SCS\lab03s4>
```

Section 5

Objectives:

- Use configuration from Section 4.
- Define lifecycle rules to prevent updates and resource deletions.
- Define explicit dependency.
- Validate, deploy, expand, analyze, and destroy infrastructure.

Part 1: Prepare for the Lab

1. Copy the lab03s4 directory as lab03s5:
2. Change into the lab03s5 directory:
3. Rename lab03s3.tf as lab03s5.tf:

Part 2: Update lab03s5.tf File

4. Open lab03s5.tf in a text editor and define an explicit dependency rule for the virtual machine to wait for the creation of the resource group.

SCREENSHOT of lab03s5.tf (capture the full content in your editor).

File Edit View

```
# Virtual Network
resource "azurerm_virtual_network" "myvnet" {
    name          = "${var.linux_name}-vnet"
    location      = var.location
    resource_group_name = var.resource_group_name
    address_space     = ["10.0.0.0/16"]

    tags = {
        Name      = local.Name
        Project   = local.Project
        ContactEmail = local.ContactEmail
        Environment = local.Environment
    }
}

# Public IP Address
resource "azurerm_public_ip" "linux_pip" {
    name          = "${var.linux_name}-pip"
    location      = var.location
    resource_group_name = var.resource_group_name
    allocation_method = "Dynamic"

    tags = {
        Name      = local.Name
        Project   = local.Project
        ContactEmail = local.ContactEmail
        Environment = local.Environment
    }
}

# Network Interface
resource "azurerm_network_interface" "linux_nic" {
    name          = "${var.linux_name}-nic"
    location      = var.location
    resource_group_name = var.resource_group_name

    ip_configuration {
        name          = "${var.linux_name}-ipconfig1"
        subnet_id      = azurerm_subnet.lab02_subnet1.id
        private_ip_address_allocation = "Dynamic"
```

```

ip_configuration {
  name                  = "${var.linux_name}-ipconfig1"
  subnet_id             = azurerm_subnet.lab02_subnet1.id
  private_ip_address_allocation = "Dynamic"
  public_ip_address_id      = azurerm_public_ip.linux_pip.id
}

tags = {
  Name      = local.Name
  Project   = local.Project
  ContactEmail = local.ContactEmail
  Environment = local.Environment
}
}

# Linux Virtual Machine
resource "azurerm_linux_virtual_machine" "linux_vm" {
  name          = var.linux_name
  location      = var.location
  resource_group_name = var.resource_group_name
  size          = var.vm_size
  admin_username = var.admin_username

  network_interface_ids = [
    azurerm_network_interface.linux_nic.id
  ]

  admin_ssh_key {
    username  = var.admin_username
    public_key = file(var.public_key)
  }

  os_disk {
    name          = "${var.linux_name}-osdisk"
    caching       = var.os_disk_caching
    storage_account_type = var.os_disk_storage_account_type
    disk_size_gb  = var.os_disk_size
  }
}

```

```

# ✓ Updated image configuration for eastus
source_image_reference {
  publisher = "Canonical"
  offer     = "UbuntuServer"
  sku       = "18_04-lts-gen2"
  version   = "latest"
}

tags = {
  Name      = local.Name
  Project   = local.Project
  ContactEmail = local.ContactEmail
  Environment = local.Environment
}

# Define explicit dependency to ensure VM is created after the resource group
depends_on = [
  azurerm_resource_group.my_resource_group
]
}

# Resource Group
resource "azurerm_resource_group" "my_resource_group" {
  name      = var.resource_group_name
  location  = var.location
}

```

Part 3: Validate Configuration

5. Validate the configuration to ensure no errors or typos:
6. Fix any issues in the Terraform files if reported.
7. Re-run validation until no errors are reported:

SCREENSHOT (capture the successful validation output).

```
C:\Users\SCS\lab03s5>terraform validate
Success! The configuration is valid.
```

Part 4: Run Simulation

8. Perform a dry run:

```
terraform plan
```

9. Review output and ensure all configuration meets requirements. Observe resources with +, -, or -/+ signs.
10. Fix any issues in the Terraform files if reported.
11. Redo the dry run until no errors are reported:

Part 5: Deploy Infrastructure

12. Deploy the infrastructure and monitor progress:

```
terraform apply
```

- Type yes when prompted.

13. Confirm output values displayed on the screen at the end of deployment.

Part 6: Add and Test a Lifecycle Deletion Rule

16. Edit lab03s5.tf in a text editor and add a rule to prevent virtual machine, public IP, and network interface resources from removal (use lifecycle { prevent_destroy = true }).

17. Run:

```
terraform destroy
```

- Observe the error message generated.
SCREENSHOT (capture the error output).

18. Edit lab03s5.tf again and remove the deletion rules.

Do not destroy the infrastructure yet.

```
Error: Instance cannot be destroyed
on lab03s5.tf line 2:
  2: resource "azurerm_virtual_network" "myvnet" {

Resource azurerm_virtual_network.myvnet has lifecycle.prevent_destroy set, but the plan calls for this resource to be destroyed. To avoid this error and
continue with the plan, either disable lifecycle.prevent_destroy or reduce the scope of the plan using the -target option.

Error: Instance cannot be destroyed
on lab03s5.tf line 65:
  65: resource "azurerm_linux_virtual_machine" "linux_vm" {

Resource azurerm_linux_virtual_machine.linux_vm has lifecycle.prevent_destroy set, but the plan calls for this resource to be destroyed. To avoid this
error and continue with the plan, either disable lifecycle.prevent_destroy or reduce the scope of the plan using the -target option.

C:\Users\SCS\Lab03s5>
```

Part 7: Add and Test a Lifecycle Update Rule

19. Go to the Azure Portal and change some tag values for the virtual machine.

20. Edit lab03s5.tf in a text editor and add a rule to prevent tag updates to the virtual machine (use lifecycle { ignore_changes = [tags] }).

21. Run:

```
terraform plan
```

- Observe the dry run output.
SCREENSHOT (capture the output).

22. Edit lab03s5.tf again and remove the update rule.

```

rossoft.Network/networkSecurityGroups/lab02-nsg2]
azurerm_container_group.lab02_container: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.ContainerInstance/containerGroups/lab02-container-14551]
azurerm_subnet.lab02_subnet1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]
azurerm_subnet.lab02_subnet2: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc2: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet2]
azurerm_subnet_network_security_group_association.lab02_nsg_assoc1: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/virtualNetworks/lab02-vnet/subnets/lab02-subnet1]
azurerm_network_interface.linux_nic: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Network/networkInterfaces/lab03s2-dbl-u-vm1-nic]
azurerm_linux_virtual_machine.linux_vm: Refreshing state... [id=/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg/providers/Microsoft.Compute/virtualMachines/lab03s2-dbl-u-vm1]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
~ update in-place

Terraform will perform the following actions:

# azurerm_resource_group.lab02_rg will be updated in-place
~ resource "azurerm_resource_group" "lab02_rg" {
  id          = "/subscriptions/ea6fc9c7-fe32-4565-b42a-e336c72040d9/resourceGroups/lab02-rg"
  name        = "lab02-rg"
  ~ tags       = {
      - "ContactEmail" = "youremailaddress" -> null
      - "Environment" = "Lab" -> null
      - "Name"        = "Terraform-Class" -> null
      - "Project"     = "Learning" -> null
    }
  # (2 unchanged attributes hidden)
}

Plan: 0 to add, 1 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.

C:\Users\SCS\lab03s5>

```

Part 8: Display Dependency Graph

23. Show dependency tree:

```
terraform graph
```

SCREENSHOT (capture the output).

```

C:\Users\SCS\lab03s5>terraform graph
graph TD
rankdir = "RL";
node [shape = rect, fontname = "sans-serif"];
"azurerm_container_group.lab02_container" [label="azurerm_container_group.lab02_container"];
"azurerm_linux_virtual_machine.linux_vm" [label="azurerm_linux_virtual_machine.linux_vm"];
"azurerm_network_interface.linux_nic" [label="azurerm_network_interface.linux_nic"];
"azurerm_network_security_group.lab02_nsg1" [label="azurerm_network_security_group.lab02_nsg1"];
"azurerm_network_security_group.lab02_nsg2" [label="azurerm_network_security_group.lab02_nsg2"];
"azurerm_public_ip.linux_ip" [label="azurerm_public_ip.linux_ip"];
"azurerm_resource_group.lab02_rg" [label="azurerm_resource_group.lab02_rg"];
"azurerm_resource_group.my_resource_group" [label="azurerm_resource_group.my_resource_group"];
"azurerm_subnet.lab02_subnet1" [label="azurerm_subnet.lab02_subnet1"];
"azurerm_subnet.lab02_subnet2" [label="azurerm_subnet.lab02_subnet2"];
"azurerm_subnet_network_security_group_association.lab02_nsg_assoc1" [label="azurerm_subnet_network_security_group_association.lab02_nsg_assoc1"];
"azurerm_subnet_network_security_group_association.lab02_nsg_assoc2" [label="azurerm_subnet_network_security_group_association.lab02_nsg_assoc2"];
"azurerm_virtual_network.lab02_vnet" [label="azurerm_virtual_network.lab02_vnet"];
"azurerm_virtual_network.myvnet" [label="azurerm_virtual_network.myvnet"];
"random_integer.rand" [label="random_integer.rand"];
"azurerm_container_group.lab02_container" --> "azurerm_resource_group.lab02_rg";
"azurerm_container_group.lab02_container" --> "random_integer.rand";
"azurerm_linux_virtual_machine.linux_vm" --> "azurerm_network_interface.linux_nic";
"azurerm_network_interface.linux_nic" --> "azurerm_public_ip.linux_ip";
"azurerm_network_interface.linux_nic" --> "azurerm_subnet.lab02_subnet1";
"azurerm_network_security_group.lab02_nsg1" --> "azurerm_resource_group.lab02_rg";
"azurerm_network_security_group.lab02_nsg2" --> "azurerm_resource_group.lab02_rg";
"azurerm_public_ip.linux_ip" --> "azurerm_resource_group.my_resource_group";
"azurerm_subnet.lab02_subnet1" --> "azurerm_virtual_network.lab02_vnet";
"azurerm_subnet.lab02_subnet2" --> "azurerm_virtual_network.lab02_vnet";
"azurerm_subnet_network_security_group_association.lab02_nsg_assoc1" --> "azurerm_network_security_group.lab02_nsg1";
"azurerm_subnet_network_security_group_association.lab02_nsg_assoc1" --> "azurerm_subnet.lab02_subnet1";
"azurerm_subnet_network_security_group_association.lab02_nsg_assoc2" --> "azurerm_network_security_group.lab02_nsg2";
"azurerm_subnet_network_security_group_association.lab02_nsg_assoc2" --> "azurerm_subnet.lab02_subnet2";
"azurerm_virtual_network.lab02_vnet" --> "azurerm_resource_group.lab02_rg";
"azurerm_virtual_network.myvnet" --> "azurerm_resource_group.my_resource_group";
}
C:\Users\SCS\lab03s5>

```

Part 9: Destroy All Resources and Verify

24. Destroy all resources:

```
terraform destroy
```

- Type yes when prompted.
25. Verify deletion:

```
terraform state list  
terraform show
```

SCREENSHOT (capture both outputs).

```
Destroy complete! Resources: 2 destroyed.  
  
C:\Users\SCS\lab03s5>terraform state list  
  
C:\Users\SCS\lab03s5>terraform show  
The state file is empty. No resources are represented.  
  
C:\Users\SCS\lab03s5>
```