Terraform Assignment on Azure 01 - Haneef Shaikh

Que 1 →

● Create two Azure AD users and one Azure AD Group using Terraform.
● Make sure you will use variables for names of Azure AD users and Group.
● Note :- Below files are required.
- main.tf
- variables.tf
- your_name_custom.tfvars

- main.tf

```
terraform {
 required_providers {
   azurerm = {
     source = "hashicorp/azurerm"
     version = "3.40.0"
   }
 }
}
provider "azurerm" {
features {}
}

data "azuread_client_config" "current" {}

resource "azuread_user" "ad_user" {
 user_principal_name = var.ad_user_principal_name
 display_name        = var.ad_user_display_name
 mail_nickname       = var.ad_mail_nickname
 password            = var.ad_password
}
resource "azuread_group" "ad_group" {
 display_name     = var.ad_group_display_name
 owners           = [data.azuread_client_config.current.object_id]
 security_enabled = true
}
resource "azuread_group_member" "ad_group_member" {
 group_object_id  = azuread_group.ad_group.object_id
 member_object_id = azuread_user.ad_user.object_id
}
```

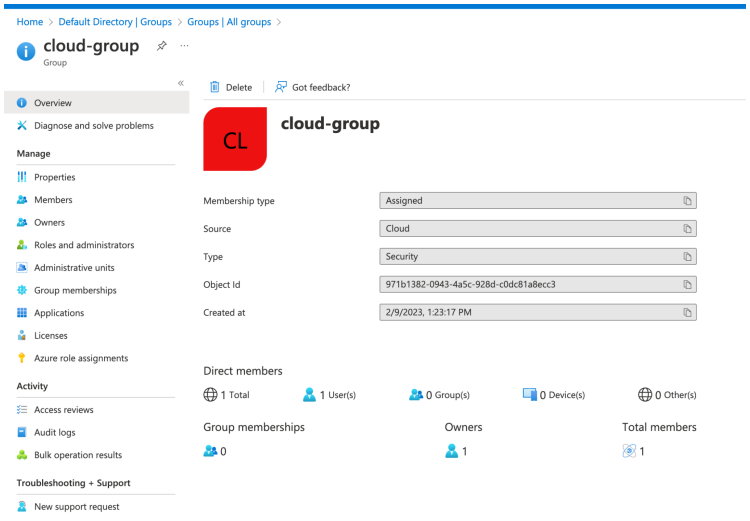
- variables.tf

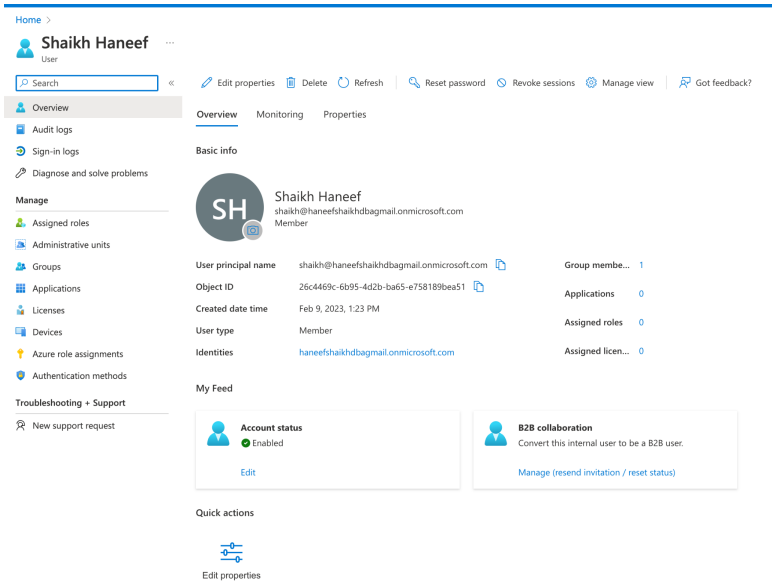
```
// AD USER
variable "ad_user_principal_name" {
   type = string
}

variable "ad_user_display_name" {
   type = string
}

variable "ad_mail_nickname" {
   type = string
}

variable "ad_password" {
   type = string
}

// AD GROUP
variable "ad_group_display_name" {
   type = string
}
```

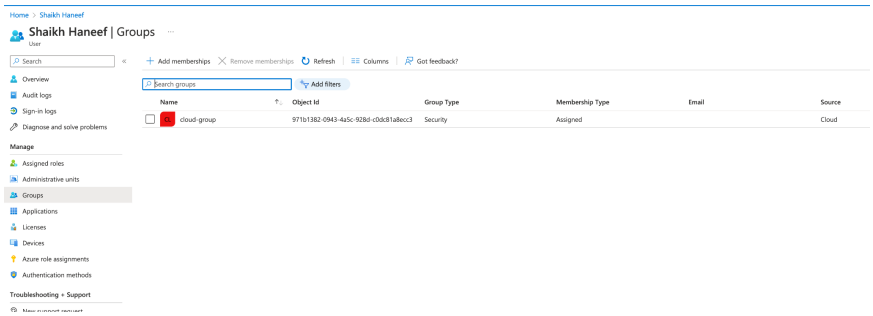
- terraform.tfvars

```
// AD USER
ad_user_principal_name  = "shaikh@haneefshaikhdbagmail.onmicrosoft.com"
ad_user_display_name    = "Shaikh Haneef"
ad_mail_nickname        = "shaikh"
ad_password             = "SecretP@sswd99!"

//AD GROUP
ad_group_display_name   = "cloud-group"
```

- **AD User and group create also assigned the created user into the created group**







Que 2 →

● Create one Azure Linux Machine , Azure public IP and Azure Network Interface using Terraform and associate public IP with Azure Linux Machine.

● Also please make sure you will use a combination of both variables in the main.tf file.

i.e. local and variable from variables.tf and custom.tfvars file.

● Also use output values to print Public IP of Azure Linux Machine.

● Note :-

Here you will require one locals in the main.tf file.

Also one output value in the main.tf file.

Main.tf

```
terraform {
 required_providers {
   azurerm = {
     source  = "hashicorp/azurerm"
     version = "3.40.0"
   }
 }
}

provider "azurerm" {
 features {}
}

// RESOURCE GROUP
resource "azurerm_resource_group" "rg" {
 name     = var.rg_name
 location = var.rg_location
}
```

compute.tf

```
locals {
 common_tags = {
     user = "devops"
   }
}
```

```
// VIRTUAL NETWORK
resource "azurerm_virtual_network" "vnet" {
 name                = var.vnet_name
 address_space       = ["10.0.0.0/16"]
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name
}

// SUBNET
resource "azurerm_subnet" "subnet" {
 name                 = var.subnet_name
 resource_group_name  = azurerm_resource_group.rg.name
 virtual_network_name = azurerm_virtual_network.vnet.name
 address_prefixes     = ["10.0.2.0/24"]
}

// NETWORK INTERFACE
resource "azurerm_network_interface" "nic" {
 name                = var.nic_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name

 ip_configuration {
   name                          = "cloud-config-01"
   subnet_id                     = azurerm_subnet.subnet.id
   private_ip_address_allocation = "Dynamic"
   public_ip_address_id          = azurerm_public_ip.public_ip.id
 }
}

// PUBLIC IP
resource "azurerm_public_ip" "public_ip" {
 name                = var.public_ip_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name
 allocation_method   = "Static"
}

// VIRTUAL MACHINE
resource "azurerm_virtual_machine" "vm" {
 name                             = var.vm_name
```

```
 location                        = azurerm_resource_group.rg.location
 resource_group_name             = azurerm_resource_group.rg.name
 network_interface_ids           = [azurerm_network_interface.nic.id]
 vm_size                         = var.vm_type
 delete_os_disk_on_termination    = true
 delete_data_disks_on_termination = true

 storage_image_reference {
   publisher = "Canonical"
   offer     = "UbuntuServer"
   sku       = "16.04-LTS"
   version   = "latest"
 }
 storage_os_disk {
   name              = "myosdisk1"
   caching           = "ReadWrite"
   create_option     = "FromImage"
   managed_disk_type = "Standard_LRS"
 }
 os_profile {
   computer_name  = var.vm_hostname
   admin_username = var.vm_username
   admin_password = var.vm_password
 }
 os_profile_linux_config {
   disable_password_authentication = false
 }
 tags = local.common_tags
}
```

Variable.tf

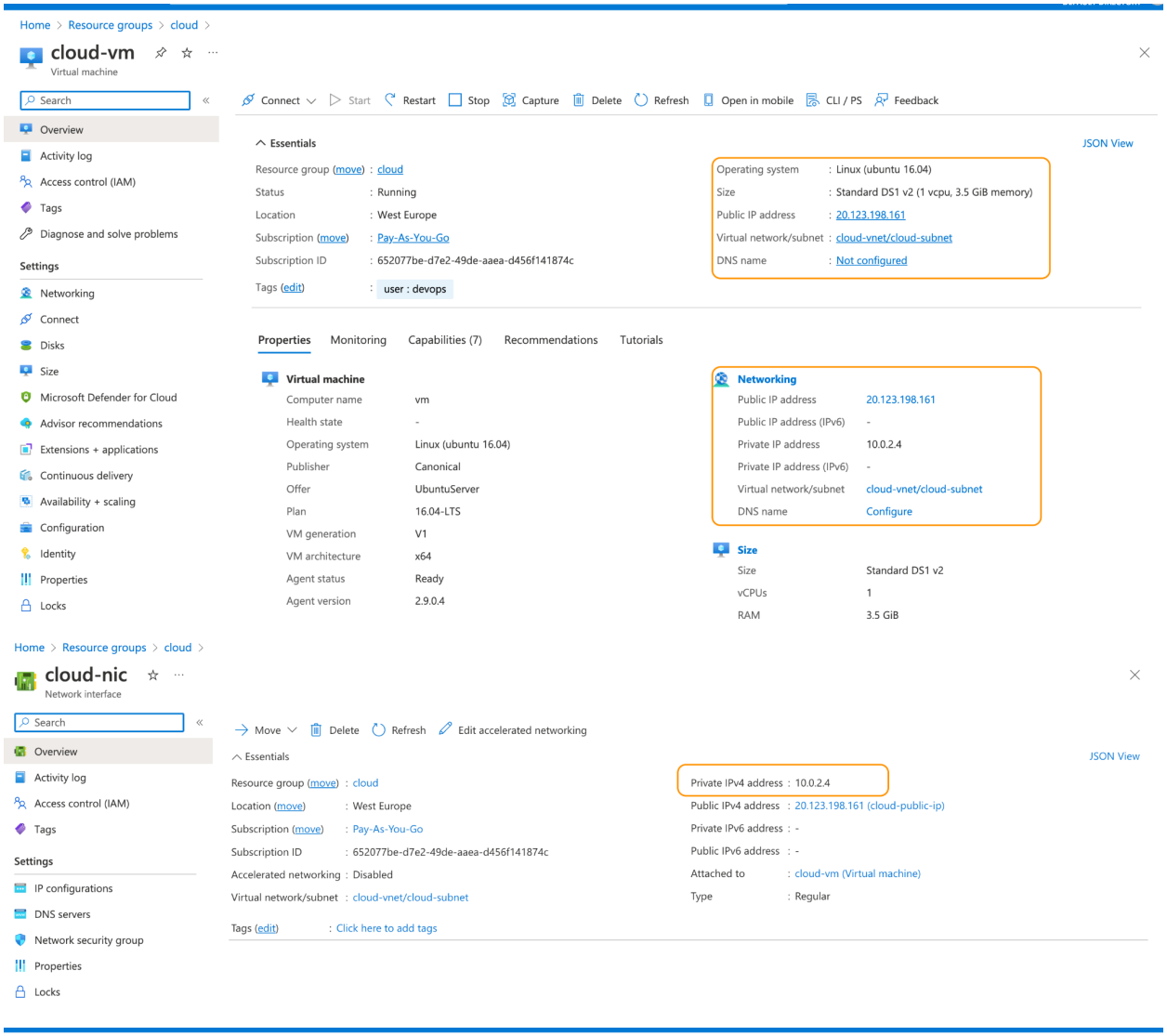
```
// RESOURCE GROUP
variable "rg_name" {
 type = string
}

variable "rg_location" {
 type = string
}
```

```
// PUBLIC IP
variable "public_ip_name" {
   type = string
}

// VIRTUAL NET
variable "vnet_name" {
 type = string
}

// SUBNET
variable "subnet_name" {
 type = string
}

// NETWORK INTERFACE
variable "nic_name" {
 type = string
}

// SECURITY GROUP
variable "sg_name" {
 type = string
}

// VM

variable "vm_name" {
 type = string
}

variable "vm_type" {
 type = string
}

variable "vm_hostname" {
 type = string
}

variable "vm_username" {
 type = string
}
```

```
variable "vm_password" {
 type = string
}
```

Terraform.tfvars

```
// NETWORK
subnet_name = "cloud-subnet"
vnet_name   = "cloud-vnet"
nic_name    = "cloud-nic"
sg_name     = "cloud-sg"

// VM
vm_name     = "cloud-vm"
vm_type     = "Standard_DS1_v2"
vm_hostname = "vm.cloud.local"
vm_username = "adminvm"
vm_password = "admin@123"

// RESOURCE GROUP
rg_location    = "West Europe"
rg_name        = "cloud"
public_ip_name = "cloud-public-ip"
```


Home > Resource groups > cloud >
cloud-vm
Virtual machine
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Settings
Networking
Connect
Disks
Size
Microsoft Defender for Cloud
Advisor recommendations
Extensions + applications
Continuous delivery
Availability + scaling
Configuration
Identity
Properties
Locks
Connect
Start
Restart
Stop
Capture
Delete
Refresh
Open in mobile
CLI / PS
Feedback
Essentials
JSON View
Resource group (move) : cloud
Status : Running
Location : West Europe
Subscription (move) : Pay-As-You-Go
Subscription ID : 652077be-d7e2-49de-aaea-d456f141874c
Tags (edit) : user : devops
Operating system : Linux (ubuntu 16.04)
Size : Standard DS1 v2 (1 vcpu, 3.5 GiB memory)
Public IP address : 20.123.198.161
Virtual network/subnet : cloud-vnet/cloud-subnet
DNS name : Not configured
Properties
Monitoring
Capabilities (7)
Recommendations
Tutorials
Virtual machine
Computer name vm
Health state -
Operating system Linux (ubuntu 16.04)
Publisher Canonical
Offer UbuntuServer
Plan 16.04-LTS
VM generation V1
VM architecture x64
Agent status Ready
Agent version 2.9.0.4
Networking
Public IP address 20.123.198.161
Public IP address (IPv6) -
Private IP address 10.0.2.4
Private IP address (IPv6) -
Virtual network/subnet cloud-vnet/cloud-subnet
DNS name Configure
Size
Size Standard DS1 v2
vCPUs 1
RAM 3.5 GiB
Home > Resource groups > cloud >
cloud-nic
Network interface
Search
Overview
Activity log
Access control (IAM)
Tags
Settings
IP configurations
DNS servers
Network security group
Properties
Locks
Move
Delete
Refresh
Edit accelerated networking
Essentials
JSON View
Resource group (move) : cloud
Location (move) : West Europe
Subscription (move) : Pay-As-You-Go
Subscription ID : 652077be-d7e2-49de-aaea-d456f141874c
Accelerated networking : Disabled
Virtual network/subnet : cloud-vnet/cloud-subnet
Tags (edit) : Click here to add tags
Private IPv4 address : 10.0.2.4
Public IPv4 address : 20.123.198.161 (cloud-public-ip)
Private IPv6 address : -
Public IPv6 address : -
Attached to : cloud-vm (Virtual machine)
Type : Regular
Home > cloud >
cloud-public-ip
Public IP address
Search
Overview
Activity log
Access control (IAM)
Tags
Settings
Configuration
Properties
Locks
Monitoring
Associate
Dissociate
Move
Delete
Refresh
Upgrade to Standard SKU - Microsoft recommends Standard SKU public IP address for production workloads →
Essentials
JSON View
Resource group (move) : cloud
Location : West Europe
Subscription (move) : Pay-As-You-Go
Subscription ID : 652077be-d7e2-49de-aaea-d456f141874c
Tags (edit) : Click here to add tags
See more
SKU : Basic
Tier : Regional
IP address : 20.123.198.161
DNS name : -
Associated to : cloud-nic

Que 3 →

● Create Azure Virtual Network with Terraform.

1. Create Resource G
2. Create Azure Virtual Network
3. Create Web Subnet
4. Create App Subnet
5. Create DB Subnet
6. Create Azure Network SG for WEb to allow all traffic on port 80
7. Create Azure Network SG for APP to allow all traffic on port 8080.
8. Create Azure Network SG for DB to allow all traffic on port 3306.
9. Associate
    a. Web NSG to WEB subnet
    b. APP NSG to APP Subnet &
    c. DB NSG to DB Subnet.
10. Create a NAT Gateway in Same resource Group
11. Associate NAT Gateway to APP subnet.

Main.tf

```
terraform {
 required_providers {
   azurerm = {
     source  = "hashicorp/azurerm"
     version = "3.40.0"
   }
 }
}

provider "azurerm" {
 features {}
}

// RESOURCE GROUP
resource "azurerm_resource_group" "rg" {
 name     = var.rg_name
 location = var.rg_location
}
```

Network.tf

```
// VIRTUAL NETWORK
resource "azurerm_virtual_network" "vnet" {
 name                = var.vnet_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name
 address_space       = ["10.0.0.0/16"]
}


// SUBNET
resource "azurerm_subnet" "web_subnet" {
 name                 = var.web_subnet_name
 resource_group_name  = azurerm_resource_group.rg.name
 virtual_network_name = azurerm_virtual_network.vnet.name
 address_prefixes     = ["10.0.1.0/24"]
}

resource "azurerm_subnet" "db_subnet" {
 name                 = var.db_subnet_name
 resource_group_name  = azurerm_resource_group.rg.name
 virtual_network_name = azurerm_virtual_network.vnet.name
 address_prefixes     = ["10.0.2.0/24"]
}

resource "azurerm_subnet" "app_subnet" {
 name                 = var.app_subnet_name
 resource_group_name  = azurerm_resource_group.rg.name
 virtual_network_name = azurerm_virtual_network.vnet.name
 address_prefixes     = ["10.0.3.0/24"]
}


// SECURITY GROUP WEB
resource "azurerm_network_security_group" "web_sg" {
 name                = var.web_sg_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name

 security_rule {
   name                       = "AllowWebTraffic"
```

```
   priority                   = 101
   direction                  = "Inbound"
   access                     = "Allow"
   protocol                   = "Tcp"
   source_port_range          = "*"
   destination_port_range     = "80"
   source_address_prefix      = "*"
   destination_address_prefix = "*"
 }
}

resource "azurerm_subnet_network_security_group_association" "web_sg_ass" {
 subnet_id                 = azurerm_subnet.web_subnet.id
 network_security_group_id = azurerm_network_security_group.web_sg.id
}

// SECURITY GROUP APP
resource "azurerm_network_security_group" "app_sg" {
 name                = var.app_sg_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name

 security_rule {
   name                       = "AllowAppTraffic"
   priority                   = 201
   direction                  = "Inbound"
   access                     = "Allow"
   protocol                   = "Tcp"
   source_port_range          = "*"
   destination_port_range     = "8080"
   source_address_prefix      = "*"
   destination_address_prefix = "*"
 }
}

resource "azurerm_subnet_network_security_group_association" "app_sg_ass" {
 subnet_id                 = azurerm_subnet.app_subnet.id
 network_security_group_id = azurerm_network_security_group.app_sg.id
}

// SECURITY GROUP DB
resource "azurerm_network_security_group" "db_sg" {
```

```
 name                = var.db_sg_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name

 security_rule {
   name                       = "AllowDBTraffic"
   priority                   = 301
   direction                  = "Inbound"
   access                     = "Allow"
   protocol                   = "Tcp"
   source_port_range          = "*"
   destination_port_range     = "3306"
   source_address_prefix      = "*"
   destination_address_prefix = "*"
 }
}

resource "azurerm_subnet_network_security_group_association" "db_sg_ass" {
 subnet_id                 = azurerm_subnet.db_subnet.id
 network_security_group_id = azurerm_network_security_group.db_sg.id
}

//NAT GATEWAY
resource "azurerm_public_ip" "nat_public_ip" {
 name                = var.public_ip_name
 resource_group_name = azurerm_resource_group.rg.name
 location            = azurerm_resource_group.rg.location
 allocation_method   = "Static"
}

resource "azurerm_nat_gateway" "nat_gateway" {
 name                = var.nat_gateway_name
 resource_group_name = azurerm_resource_group.rg.name
 location            = azurerm_resource_group.rg.location
}

resource "azurerm_subnet_nat_gateway_association" "nat_gateway_subnet_ass" {
 subnet_id      = azurerm_subnet.app_subnet.id
 nat_gateway_id = azurerm_nat_gateway.nat_gateway.id
}
```

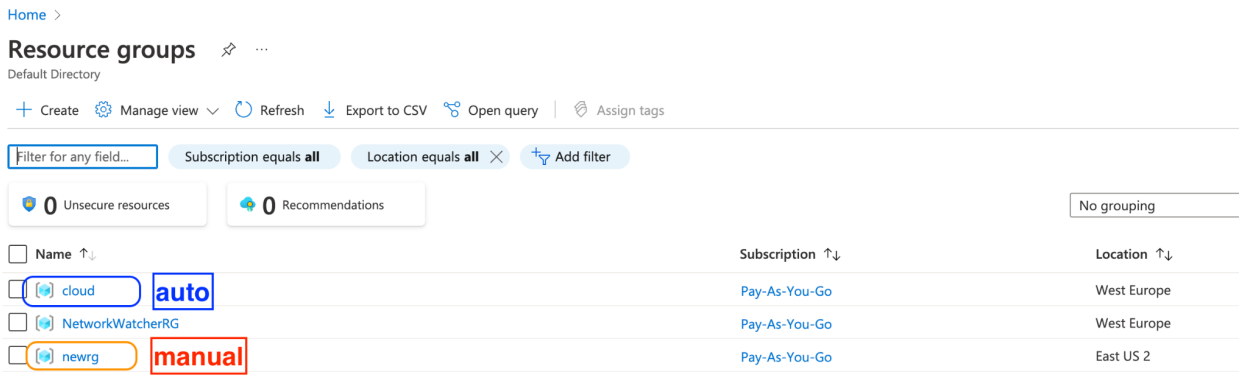
Variable.tf

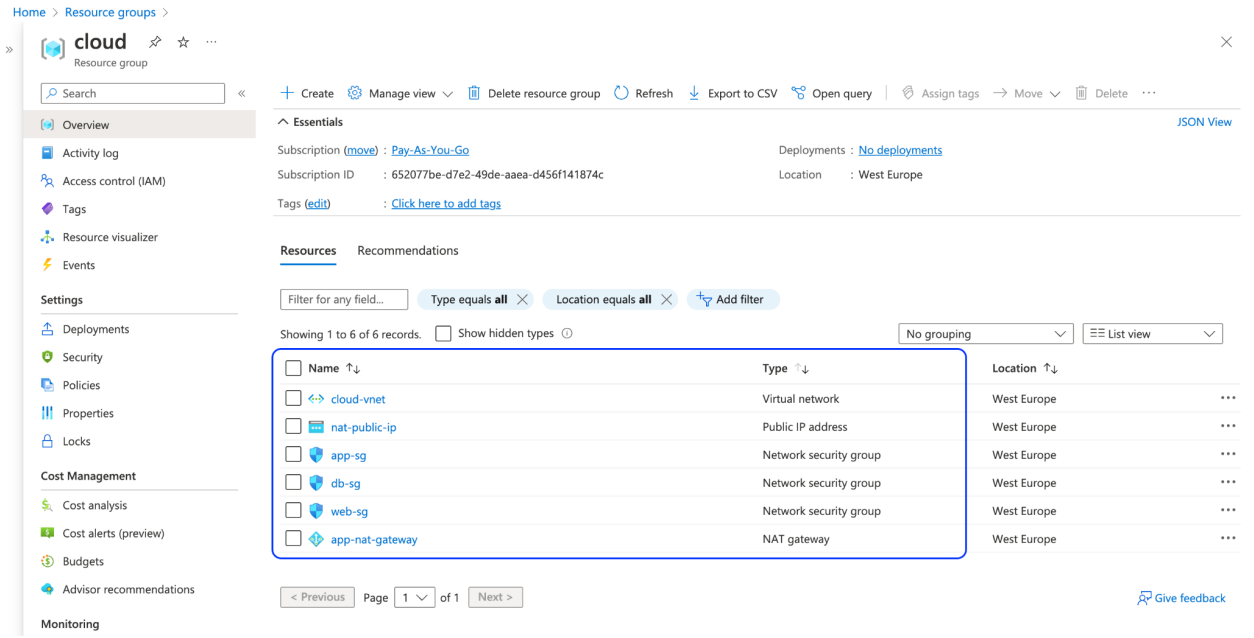
```
// RESOURCE GROUP
variable "rg_name" {
 type = string
}

variable "rg_location" {
 type = string
}

// VIRTUAL NET
variable "vnet_name" {
 type = string
}

// SUBNET
variable "web_subnet_name" {
 type = string
}
variable "db_subnet_name" {
 type = string
}
variable "app_subnet_name" {
 type = string
}

// SECURITY GROUP
variable "web_sg_name" {
 type = string
}
variable "db_sg_name" {
 type = string
}
variable "app_sg_name" {
 type = string
}

// PUBLIC IP
variable "public_ip_name" {
 type = string
```

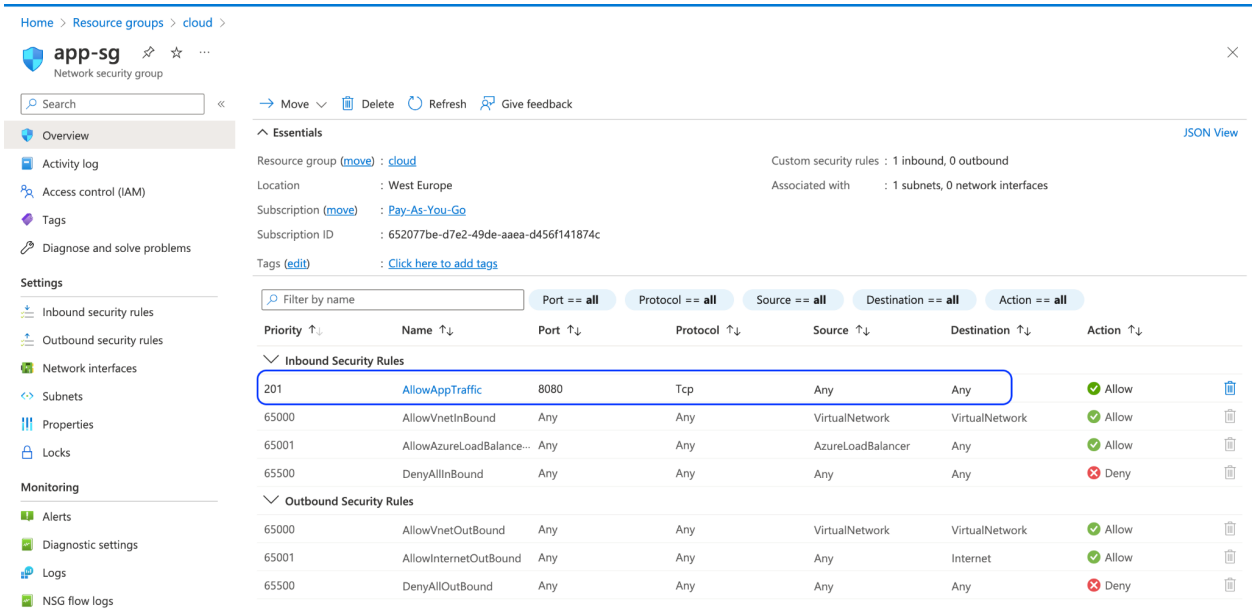
```
}
variable "nat_gateway_name" {
 type = string
}
```

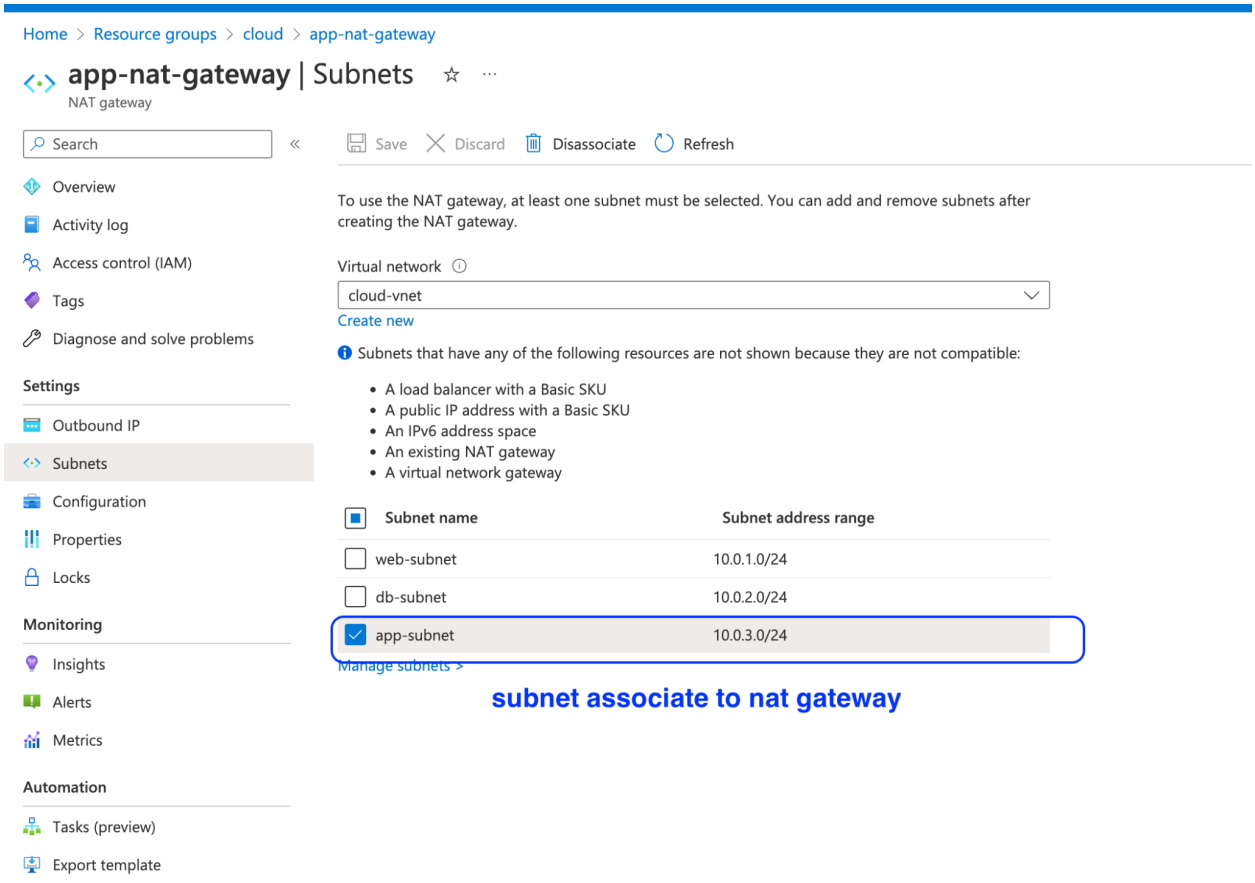
Terraform.tfvars

```
// RESOURCE GROUP
rg_location = "West Europe"
rg_name     = "cloud"

// NETWORK
vnet_name        = "cloud-vnet"
web_subnet_name = "web-subnet"
db_subnet_name   = "db-subnet"
app_subnet_name = "app-subnet"

// SECURITY GROUP
web_sg_name = "web-sg"
app_sg_name = "app-sg"
db_sg_name   = "db-sg"

//NAT
public_ip_name    = "nat-public-ip"
nat_gateway_name = "app-nat-gateway"
```




Home > Resource groups >
cloud
Resource group
Search
Overview
Activity log
Access control (IAM)
Tags
Resource visualizer
Events
Settings
Deployments
Security
Policies
Properties
Locks
Cost Management
Cost analysis
Cost alerts (preview)
Budgets
Advisor recommendations
Monitoring
Create
Manage view
Delete resource group
Refresh
Export to CSV
Open query
Assign tags
Move
Delete
Essentials
JSON View
Subscription (move) : Pay-As-You-Go
Subscription ID : 652077be-d7e2-49de-aaea-d456f141874c
Tags (edit) : Click here to add tags
Deployments : No deployments
Location : West Europe
Resources
Recommendations
Filter for any field...
Type equals all
Location equals all
Add filter
Showing 1 to 6 of 6 records.
Show hidden types
No grouping
List view
Name
Type
Location
cloud-vnet
Virtual network
West Europe
nat-public-ip
Public IP address
West Europe
app-sg
Network security group
West Europe
db-sg
Network security group
West Europe
web-sg
Network security group
West Europe
app-nat-gateway
NAT gateway
West Europe
< Previous
Page 1 of 1
Next >
Give feedback


Home > Resource groups > cloud >
app-sg
Network security group
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Settings
Inbound security rules
Outbound security rules
Network interfaces
Subnets
Properties
Locks
Monitoring
Alerts
Diagnostic settings
Logs
NSG flow logs
Move
Delete
Refresh
Give feedback
Essentials
JSON View
Resource group (move) : cloud
Location : West Europe
Subscription (move) : Pay-As-You-Go
Subscription ID : 652077be-d7e2-49de-aaea-d456f141874c
Tags (edit) : Click here to add tags
Custom security rules : 1 inbound, 0 outbound
Associated with : 1 subnets, 0 network interfaces
Filter by name
Port == all
Protocol == all
Source == all
Destination == all
Action == all
Priority
Name
Port
Protocol
Source
Destination
Action
Inbound Security Rules
201 AllowAppTraffic 8080 Tcp Any Any Allow
65000 AllowVnetInBound Any Any VirtualNetwork VirtualNetwork Allow
65001 AllowAzureLoadBalance... Any Any AzureLoadBalancer Any Allow
65500 DenyAllInBound Any Any Any Any Deny
Outbound Security Rules
65000 AllowVnetOutBound Any Any VirtualNetwork VirtualNetwork Allow
65001 AllowInternetOutBound Any Any Any Internet Allow
65500 DenyAllOutBound Any Any Any Any Deny


Home > Resource groups > cloud > app-nat-gateway
app-nat-gateway | Subnets
NAT gateway
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Settings
Outbound IP
Subnets
Configuration
Properties
Locks
Monitoring
Insights
Alerts
Metrics
Automation
Tasks (preview)
Export template
Save
Discard
Disassociate
Refresh
To use the NAT gateway, at least one subnet must be selected. You can add and remove subnets after creating the NAT gateway.
Virtual network
cloud-vnet
Create new
Subnets that have any of the following resources are not shown because they are not compatible:
A load balancer with a Basic SKU
A public IP address with a Basic SKU
An IPv6 address space
An existing NAT gateway
A virtual network gateway
Subnet name
Subnet address range
web-subnet
10.0.1.0/24
db-subnet
10.0.2.0/24
app-subnet
10.0.3.0/24
Manage subnets >
subnet associate to nat gateway

Que 4 →

- **Create Azure VM in the App Subnets & Validate your Connection using ssh.**
- **Also check if you are able to ping google.com from that VM.**
- **For this You need to create the Azure VM using Terraform.**
  - **Azure VM.**
  - **Enable Password Authentication.**
  - **Try to access a VM.**

Main.tf

```
terraform {
 required_providers {
   azurerm = {
     source  = "hashicorp/azurerm"
     version = "3.40.0"
   }
 }
}

provider "azurerm" {
 features {}
}

// RESOURCE GROUP
resource "azurerm_resource_group" "rg" {
 name     = var.rg_name
 location = var.rg_location
}
```

Network.tf

```
// VIRTUAL NETWORK
resource "azurerm_virtual_network" "vnet" {
 name                = var.vnet_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name
 address_space       = ["10.0.0.0/16"]
}


// SUBNET
```

```
resource "azurerm_subnet" "web_subnet" {
 name                 = var.web_subnet_name
 resource_group_name  = azurerm_resource_group.rg.name
 virtual_network_name = azurerm_virtual_network.vnet.name
 address_prefixes     = ["10.0.1.0/24"]
}

resource "azurerm_subnet" "db_subnet" {
 name                 = var.db_subnet_name
 resource_group_name  = azurerm_resource_group.rg.name
 virtual_network_name = azurerm_virtual_network.vnet.name
 address_prefixes     = ["10.0.2.0/24"]
}

resource "azurerm_subnet" "app_subnet" {
 name                 = var.app_subnet_name
 resource_group_name  = azurerm_resource_group.rg.name
 virtual_network_name = azurerm_virtual_network.vnet.name
 address_prefixes     = ["10.0.3.0/24"]
}


// SECURITY GROUP WEB
resource "azurerm_network_security_group" "web_sg" {
 name                = var.web_sg_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name

 security_rule {
   name                       = "AllowWebTraffic"
   priority                   = 101
   direction                  = "Inbound"
   access                     = "Allow"
   protocol                   = "Tcp"
   source_port_range          = "*"
   destination_port_range     = "80"
   source_address_prefix      = "*"
   destination_address_prefix = "*"
 }
}

resource "azurerm_subnet_network_security_group_association" "web_sg_ass" {
```

```
 subnet_id                 = azurerm_subnet.web_subnet.id
 network_security_group_id = azurerm_network_security_group.web_sg.id
}

// SECURITY GROUP APP
resource "azurerm_network_security_group" "app_sg" {
 name                = var.app_sg_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name

 security_rule {
   name                       = "AllowAppTraffic"
   priority                   = 201
   direction                  = "Inbound"
   access                     = "Allow"
   protocol                   = "Tcp"
   source_port_range          = "*"
   destination_port_range     = "8080"
   source_address_prefix      = "*"
   destination_address_prefix = "*"
 }
}

resource "azurerm_subnet_network_security_group_association" "app_sg_ass" {
 subnet_id                 = azurerm_subnet.app_subnet.id
 network_security_group_id = azurerm_network_security_group.app_sg.id
}

// SECURITY GROUP DB
resource "azurerm_network_security_group" "db_sg" {
 name                = var.db_sg_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name

 security_rule {
   name                       = "AllowDBTraffic"
   priority                   = 301
   direction                  = "Inbound"
   access                     = "Allow"
   protocol                   = "Tcp"
   source_port_range          = "*"
   destination_port_range     = "3306"
```

```
   source_address_prefix      = "*"
   destination_address_prefix = "*"
 }
}

resource "azurerm_subnet_network_security_group_association" "db_sg_ass" {
 subnet_id                 = azurerm_subnet.db_subnet.id
 network_security_group_id = azurerm_network_security_group.db_sg.id
}

//NAT GATEWAY
resource "azurerm_public_ip" "nat_public_ip" {
 name                = var.public_ip_name
 resource_group_name = azurerm_resource_group.rg.name
 location            = azurerm_resource_group.rg.location
 allocation_method   = "Static"
 sku                 = "Standard"
}

resource "azurerm_nat_gateway" "nat_gateway" {
 name                = var.nat_gateway_name
 resource_group_name = azurerm_resource_group.rg.name
 location            = azurerm_resource_group.rg.location
}

resource "azurerm_subnet_nat_gateway_association" "nat_gateway_subnet_ass" {
 subnet_id      = azurerm_subnet.app_subnet.id
 nat_gateway_id = azurerm_nat_gateway.nat_gateway.id
}
```

compute.tf

```
locals {
 common_tags = {
   user = "devops"
 }
}

// NETWORK INTERFACE
```

```
resource "azurerm_network_interface" "nic" {
 name                = var.nic_name
 location            = azurerm_resource_group.rg.location
 resource_group_name = azurerm_resource_group.rg.name
 tags                = local.common_tags

 ip_configuration {
   name                          = "cloud-config-01"
   subnet_id                     = azurerm_subnet.app_subnet.id
   private_ip_address_allocation = "Dynamic"
   public_ip_address_id          = azurerm_public_ip.nat_public_ip.id
 }
}

// VIRTUAL MACHINE
resource "azurerm_virtual_machine" "vm" {
 name                             = var.vm_name
 location                         = azurerm_resource_group.rg.location
 resource_group_name              = azurerm_resource_group.rg.name
 network_interface_ids            = [azurerm_network_interface.nic.id]
 vm_size                          = var.vm_type
 delete_os_disk_on_termination    = true
 delete_data_disks_on_termination = true

 storage_image_reference {
   publisher = "Canonical"
   offer     = "UbuntuServer"
   sku       = "16.04-LTS"
   version   = "latest"
 }
 storage_os_disk {
   name              = "myosdisk1"
   caching           = "ReadWrite"
   create_option     = "FromImage"
   managed_disk_type = "Standard_LRS"
 }
 os_profile {
   computer_name  = var.vm_hostname
   admin_username = var.vm_username
   admin_password = var.vm_password
 }
 os_profile_linux_config {
```

```
   disable_password_authentication = false
 }
 tags = local.common_tags
}
```

Variable.tf

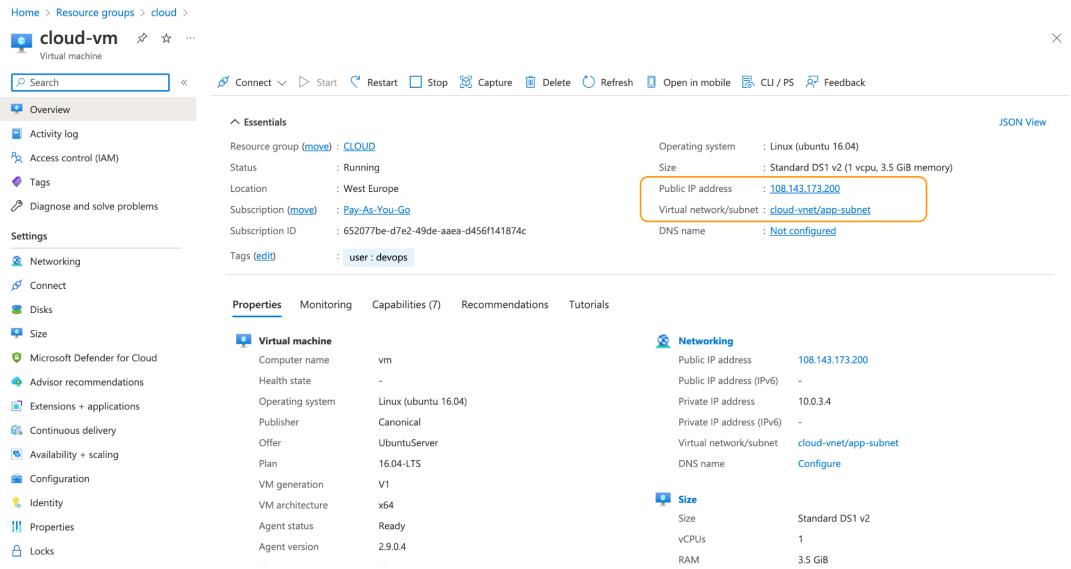
```
// RESOURCE GROUP
variable "rg_name" {
 type = string
}

variable "rg_location" {
 type = string
}

// VIRTUAL NET
variable "vnet_name" {
 type = string
}

// SUBNET
variable "web_subnet_name" {
 type = string
}
variable "db_subnet_name" {
 type = string
}
variable "app_subnet_name" {
 type = string
}

// SECURITY GROUP
variable "web_sg_name" {
 type = string
}
variable "db_sg_name" {
 type = string
}
variable "app_sg_name" {
```

```
 type = string
}

// PUBLIC IP
variable "public_ip_name" {
 type = string
}
variable "nat_gateway_name" {
 type = string
}

// NETWORK INTERFACE
variable "nic_name" {
 type = string
}

// VM
variable "vm_name" {
 type = string
}

variable "vm_type" {
 type = string
}

variable "vm_hostname" {
 type = string
}

variable "vm_username" {
 type = string
}

variable "vm_password" {
 type = string
}
```

Terraform.tfvars

```
// RESOURCE GROUP
rg_location = "West Europe"
rg_name     = "cloud"
```

```
// NETWORK
vnet_name        = "cloud-vnet"
web_subnet_name  = "web-subnet"
db_subnet_name   = "db-subnet"
app_subnet_name  = "app-subnet"

// SECURITY GROUP
web_sg_name  = "web-sg"
app_sg_name  = "app-sg"
db_sg_name   = "db-sg"

//NAT
public_ip_name    = "nat-public-ip"
nat_gateway_name  = "app-nat-gateway"

// NIC
nic_name     = "cloud-nic"

// VM
vm_name      = "cloud-vm"
vm_type      = "Standard_DS1_v2"
vm_hostname  = "vm.cloud.local"
vm_username  = "adminvm"
vm_password  = "admin@123"
```



```
Haneefs-MacBook-Pro-M2:02_Assignment haneefshaikh$ ssh adminvm@108.143.173.200
ssh: connect to host 108.143.173.200 port 22: Operation timed out
Haneefs-MacBook-Pro-M2:02_Assignment haneefshaikh$ ssh adminvm@108.143.173.200
The authenticity of host '108.143.173.200 (108.143.173.200)' can't be established.
ED25519 key fingerprint is SHA256:Ool3k8nLqwh9YuZxtipy5w4dCusIpW1It2BitsVsCl4.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '108.143.173.200' (ED25519) to the list of known hosts.
adminvm@108.143.173.200's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.15.0-1113-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

UA Infra: Extended Security Maintenance (ESM) is not enabled.

0 updates can be applied immediately.

52 additional security updates can be applied with UA Infra: ESM
Learn more about enabling UA Infra: ESM service for Ubuntu 16.04 at
https://ubuntu.com/16-04

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

_____________________________________________________________________
WARNING! Your environment specifies an invalid locale.
 The unknown environment variables are:
   LC_CTYPE=UTF-8 LC_ALL=
 This can affect your user experience significantly, including the
 ability to manage packages. You may install the locales by running:

   sudo apt-get install language-pack-UTF-8
     or
   sudo locale-gen UTF-8

To see all available language packs, run:
   apt-cache search "^language-pack-[a-z][a-z]$"
To disable this message for all users, run:
   sudo touch /var/lib/cloud/instance/locale-check.skip
_____________________________________________________________________

adminvm@vm:~$ hostname
vm
adminvm@vm:~$ hostname -i
hostname: Name or service not known
adminvm@vm:~$ ip r l
default via 10.0.3.1 dev eth0
10.0.3.0/24 dev eth0  proto kernel  scope link  src 10.0.3.4
168.63.129.16 via 10.0.3.1 dev eth0
169.254.169.254 via 10.0.3.1 dev eth0
adminvm@vm:~$
```

GIT REPO -> https://github.com/haneefshaikh/Terraform_Azure_Assignment