```
$ terraform init

Initializing the backend...

Initializing provider plugins...
- Finding hashicorp/azurerm versions matching "~> 2.0"...
- Finding hashicorp/random versions matching "~> 3.0"...
- Finding hashicorp/tls versions matching "~> 4.0"...
- Installing hashicorp/azurerm v2.99.0...
- Installed hashicorp/azurerm v2.99.0 (signed by HashiCorp)
- Installing hashicorp/random v3.4.3...
- Installed hashicorp/random v3.4.3 (signed by HashiCorp)
- Installing hashicorp/tls v4.0.4...
- Installed hashicorp/tls v4.0.4 (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.
```

```
$ terraform plan -out main.tfplan

Terraform used the selected providers to generate the following execution plan. Resource actions are
indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # azurerm_linux_virtual_machine.my_terraform_vm will be created
  + resource "azurerm_linux_virtual_machine" "my_terraform_vm" {
      + admin_username                  = "azureuser"
      + allow_extension_operations      = true
      + computer_name                   = "myvm"
      + disable_password_authentication = true
      + extensions_time_budget          = "PT1H30M"
      + id                              = (known after apply)
      + location                        = "eastus"
      + max_bid_price                   = -1
      + name                            = "myVM"
      + network_interface_ids           = (known after apply)
      + patch_mode                      = "ImageDefault"
      + platform_fault_domain           = -1
      + priority                        = "Regular"
      + private_ip_address              = (known after apply)
      + private_ip_addresses            = (known after apply)
      + provision_vm_agent              = true
      + public_ip_address               = (known after apply)
      + public_ip_addresses             = (known after apply)
      + resource_group_name             = (known after apply)
      + size                            = "Standard_DS1_v2"
      + virtual_machine_id              = (known after apply)
      + zone                            = (known after apply)

      + admin_ssh_key {
          + public_key = (known after apply)
          + username   = "azureuser"
        }

      + boot_diagnostics {
          + storage_account_uri = (known after apply)
        }

      + os_disk {
          + caching                   = "ReadWrite"
          + disk_size_gb              = (known after apply)
          + name                      = "myOsDisk"
          + storage_account_type      = "Premium_LRS"
          + write_accelerator_enabled = false
        }
```

```
    + source_image_reference {
        + offer     = "UbuntuServer"
        + publisher = "Canonical"
        + sku       = "18.04-LTS"
        + version   = "latest"
      }
  }

# azurerm_network_interface.my_terraform_nic will be created
+ resource "azurerm_network_interface" "my_terraform_nic" {
    + applied_dns_servers           = (known after apply)
    + dns_servers                   = (known after apply)
    + enable_accelerated_networking = false
    + enable_ip_forwarding          = false
    + id                            = (known after apply)
    + internal_dns_name_label       = (known after apply)
    + internal_domain_name_suffix   = (known after apply)
    + location                      = "eastus"
    + mac_address                   = (known after apply)
    + name                          = "myNIC"
    + private_ip_address            = (known after apply)
    + private_ip_addresses          = (known after apply)
    + resource_group_name           = (known after apply)
    + virtual_machine_id            = (known after apply)

    + ip_configuration {
        + gateway_load_balancer_frontend_ip_configuration_id = (known after apply)
        + name                                               = "my_nic_configuration"
        + primary                                            = (known after apply)
        + private_ip_address                                 = (known after apply)
        + private_ip_address_allocation                      = "Dynamic"
        + private_ip_address_version                         = "IPv4"
        + public_ip_address_id                               = (known after apply)
        + subnet_id                                          = (known after apply)
      }
  }

# azurerm_network_interface_security_group_association.example will be created
+ resource "azurerm_network_interface_security_group_association" "example" {
    + id                        = (known after apply)
    + network_interface_id      = (known after apply)
    + network_security_group_id = (known after apply)
  }
```

```
# azurerm_network_security_group.my_terraform_nsg will be created
+ resource "azurerm_network_security_group" "my_terraform_nsg" {
    + id                  = (known after apply)
    + location            = "eastus"
    + name                = "myNetworkSecurityGroup"
    + resource_group_name = (known after apply)
    + security_rule       = [
        + {
            + access                                     = "Allow"
            + description                                = ""
            + destination_address_prefix                 = "*"
            + destination_address_prefixes               = []
            + destination_application_security_group_ids = []
            + destination_port_range                     = "22"
            + destination_port_ranges                    = []
            + direction                                  = "Inbound"
            + name                                       = "SSH"
            + priority                                   = 1001
            + protocol                                   = "Tcp"
            + source_address_prefix                      = "*"
            + source_address_prefixes                    = []
            + source_application_security_group_ids      = []
            + source_port_range                          = "*"
            + source_port_ranges                         = []
          },
      ]
  }

# azurerm_public_ip.my_terraform_public_ip will be created
+ resource "azurerm_public_ip" "my_terraform_public_ip" {
    + allocation_method       = "Dynamic"
    + availability_zone       = (known after apply)
    + fqdn                    = (known after apply)
    + id                      = (known after apply)
    + idle_timeout_in_minutes = 4
    + ip_address              = (known after apply)
    + ip_version              = "IPv4"
    + location                = "eastus"
    + name                    = "myPublicIP"
    + resource_group_name     = (known after apply)
    + sku                     = "Basic"
    + sku_tier                = "Regional"
    + zones                   = (known after apply)
  }

# azurerm_resource_group.rg will be created
+ resource "azurerm_resource_group" "rg" {
    + id       = (known after apply)
    + location = "eastus"
    + name     = (known after apply)
  }
```

```
# azurerm_storage_account.my_storage_account will be created
+ resource "azurerm_storage_account" "my_storage_account" {
    + access_tier                      = (known after apply)
    + account_kind                     = "StorageV2"
    + account_replication_type         = "LRS"
    + account_tier                     = "Standard"
    + allow_blob_public_access         = false
    + enable_https_traffic_only        = true
    + id                               = (known after apply)
    + infrastructure_encryption_enabled = false
    + is_hns_enabled                   = false
    + large_file_share_enabled         = (known after apply)
    + location                         = "eastus"
    + min_tls_version                  = "TLS1_0"
    + name                             = (known after apply)
    + nfsv3_enabled                    = false
    + primary_access_key               = (sensitive value)
    + primary_blob_connection_string   = (sensitive value)
    + primary_blob_endpoint            = (known after apply)
    + primary_blob_host                = (known after apply)
    + primary_connection_string        = (sensitive value)
    + primary_dfs_endpoint             = (known after apply)
    + primary_dfs_host                 = (known after apply)
    + primary_file_endpoint            = (known after apply)
    + primary_file_host                = (known after apply)
    + primary_location                 = (known after apply)
    + primary_queue_endpoint           = (known after apply)
    + primary_queue_host               = (known after apply)
    + primary_table_endpoint           = (known after apply)
    + primary_table_host               = (known after apply)
    + primary_web_endpoint             = (known after apply)
    + primary_web_host                 = (known after apply)
    + queue_encryption_key_type        = "Service"
    + resource_group_name              = (known after apply)
    + secondary_access_key             = (sensitive value)
    + secondary_blob_connection_string = (sensitive value)
    + secondary_blob_endpoint          = (known after apply)
    + secondary_blob_host              = (known after apply)
    + secondary_connection_string      = (sensitive value)
    + secondary_dfs_endpoint           = (known after apply)
    + secondary_dfs_host               = (known after apply)
    + secondary_file_endpoint          = (known after apply)
    + secondary_file_host              = (known after apply)
    + secondary_location               = (known after apply)
    + secondary_queue_endpoint         = (known after apply)
    + secondary_queue_host             = (known after apply)
    + secondary_table_endpoint         = (known after apply)
    + secondary_table_host             = (known after apply)
    + secondary_web_endpoint           = (known after apply)
    + secondary_web_host               = (known after apply)
    + shared_access_key_enabled        = true
    + table_encryption_key_type        = "Service"
```

```
    + blob_properties {
        + change_feed_enabled      = (known after apply)
        + default_service_version  = (known after apply)
        + last_access_time_enabled = (known after apply)
        + versioning_enabled       = (known after apply)

        + container_delete_retention_policy {
            + days = (known after apply)
          }

        + cors_rule {
            + allowed_headers    = (known after apply)
            + allowed_methods    = (known after apply)
            + allowed_origins    = (known after apply)
            + exposed_headers    = (known after apply)
            + max_age_in_seconds = (known after apply)
          }

        + delete_retention_policy {
            + days = (known after apply)
          }
      }

    + customer_managed_key {
        + key_vault_key_id          = (known after apply)
        + user_assigned_identity_id = (known after apply)
      }

    + network_rules {
        + bypass                     = (known after apply)
        + default_action             = (known after apply)
        + ip_rules                   = (known after apply)
        + virtual_network_subnet_ids = (known after apply)

        + private_link_access {
            + endpoint_resource_id = (known after apply)
            + endpoint_tenant_id   = (known after apply)
          }
      }

    + queue_properties {
        + cors_rule {
            + allowed_headers    = (known after apply)
            + allowed_methods    = (known after apply)
            + allowed_origins    = (known after apply)
            + exposed_headers    = (known after apply)
            + max_age_in_seconds = (known after apply)
          }
```

```
      + hour_metrics {
          + enabled              = (known after apply)
          + include_apis         = (known after apply)
          + retention_policy_days = (known after apply)
          + version              = (known after apply)
        }

      + logging {
          + delete               = (known after apply)
          + read                 = (known after apply)
          + retention_policy_days = (known after apply)
          + version              = (known after apply)
          + write                = (known after apply)
        }

      + minute_metrics {
          + enabled              = (known after apply)
          + include_apis         = (known after apply)
          + retention_policy_days = (known after apply)
          + version              = (known after apply)
        }
    }

    + routing {
        + choice                      = (known after apply)
        + publish_internet_endpoints  = (known after apply)
        + publish_microsoft_endpoints = (known after apply)
      }

    + share_properties {
        + cors_rule {
            + allowed_headers    = (known after apply)
            + allowed_methods    = (known after apply)
            + allowed_origins    = (known after apply)
            + exposed_headers    = (known after apply)
            + max_age_in_seconds = (known after apply)
          }

        + retention_policy {
            + days = (known after apply)
          }

        + smb {
            + authentication_types            = (known after apply)
            + channel_encryption_type         = (known after apply)
            + kerberos_ticket_encryption_type = (known after apply)
            + versions                        = (known after apply)
          }
      }
  }
```

```
# azurerm_subnet.my_terraform_subnet will be created
+ resource "azurerm_subnet" "my_terraform_subnet" {
    + address_prefix                                 = (known after apply)
    + address_prefixes                               = [
        + "10.0.1.0/24",
      ]
    + enforce_private_link_endpoint_network_policies = false
    + enforce_private_link_service_network_policies  = false
    + id                                             = (known after apply)
    + name                                           = "mySubnet"
    + resource_group_name                            = (known after apply)
    + virtual_network_name                           = "myVnet"
  }

# azurerm_virtual_network.my_terraform_network will be created
+ resource "azurerm_virtual_network" "my_terraform_network" {
    + address_space         = [
        + "10.0.0.0/16",
      ]
    + dns_servers           = (known after apply)
    + guid                  = (known after apply)
    + id                    = (known after apply)
    + location              = "eastus"
    + name                  = "myVnet"
    + resource_group_name   = (known after apply)
    + subnet                = (known after apply)
    + vm_protection_enabled = false
  }

# random_id.random_id will be created
+ resource "random_id" "random_id" {
    + b64_std     = (known after apply)
    + b64_url     = (known after apply)
    + byte_length = 8
    + dec         = (known after apply)
    + hex         = (known after apply)
    + id          = (known after apply)
    + keepers     = {
        + "resource_group" = (known after apply)
      }
  }

# random_pet.rg_name will be created
+ resource "random_pet" "rg_name" {
    + id        = (known after apply)
    + length    = 2
    + prefix    = "rg"
    + separator = "-"
  }
```

```
  # tls_private_key.example_ssh will be created
  + resource "tls_private_key" "example_ssh" {
      + algorithm                   = "RSA"
      + ecdsa_curve                 = "P224"
      + id                          = (known after apply)
      + private_key_openssh         = (sensitive value)
      + private_key_pem             = (sensitive value)
      + private_key_pem_pkcs8       = (sensitive value)
      + public_key_fingerprint_md5  = (known after apply)
      + public_key_fingerprint_sha256 = (known after apply)
      + public_key_openssh          = (known after apply)
      + public_key_pem              = (known after apply)
      + rsa_bits                    = 4096
    }

Plan: 12 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + public_ip_address   = (known after apply)
  + resource_group_name = (known after apply)
  + tls_private_key     = (sensitive value)

---

Saved the plan to: main.tfplan

To perform exactly these actions, run the following command to apply:
    terraform apply "main.tfplan"
```

```
$ terraform apply main.tfplan
random_pet.rg_name: Creating...
random_pet.rg_name: Creation complete after 0s [id=rg-distinct-octopus]
tls_private_key.example_ssh: Creating...
tls_private_key.example_ssh: Creation complete after 2s [id=ae9a74ebf56dd7f499472c4761247d9df9c3d8aa]
azurerm_resource_group.rg: Creating...
azurerm_resource_group.rg: Creation complete after 2s [id=/subscriptions/ac1156f0-0914-4629-90c5-323d8c283fc7/resourceGroups/rg-distinct-octopus]
azurerm_public_ip.my_terraform_public_ip: Creating...
azurerm_network_security_group.my_terraform_nsg: Creating...
azurerm_virtual_network.my_terraform_network: Creating...
random_id.random_id: Creating...
random_id.random_id: Creation complete after 0s [id=ukLcQ6cxNZI]
azurerm_storage_account.my_storage_account: Creating...
azurerm_public_ip.my_terraform_public_ip: Creation complete after 2s [id=/subscriptions/ac1156f0-0914-4629-90c5-323d8c283fc7/resourceGroups/rg-distinct-octopus/providers/Microsoft.Network/publicIPAddresses/myPublicIP]
azurerm_network_security_group.my_terraform_nsg: Creation complete after 5s [id=/subscriptions/ac1156f0-0914-4629-90c5-323d8c283fc7/resourceGroups/rg-distinct-octopus/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup]
azurerm_virtual_network.my_terraform_network: Creation complete after 5s [id=/subscriptions/ac1156f0-0914-4629-90c5-323d8c283fc7/resourceGroups/rg-distinct-octopus/providers/Microsoft.Network/virtualNetworks/myVnet]
azurerm_subnet.my_terraform_subnet: Creating...
azurerm_subnet.my_terraform_subnet: Creation complete after 4s [id=/subscriptions/ac1156f0-0914-4629-90c5-323d8c283fc7/resourceGroups/rg-distinct-octopus/providers/Microsoft.Network/virtualNetworks/myVnet/subnets/mySubnet]
azurerm_network_interface.my_terraform_nic: Creating...
azurerm_storage_account.my_storage_account: Still creating... [10s elapsed]
azurerm_network_interface.my_terraform_nic: Creation complete after 3s [id=/subscriptions/ac1156f0-0914-4629-90c5-323d8c283fc7/resourceGroups/rg-distinct-octopus/providers/Microsoft.Network/networkInterfaces/myNIC]
azurerm_network_interface_security_group_association.example: Creating...
azurerm_network_interface_security_group_association.example: Creation complete after 1s [id=/subscriptions/ac1156f0-0914-4629-90c5-323d8c283fc7/resourceGroups/rg-distinct-octopus/providers/Microsoft.Network/networkInterfaces/myNIC|/subscriptions/ac1156f0-0914-4629-90c5-323d8c283fc7/resourceGroups/rg-distinct-octopus/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup]
azurerm_storage_account.my_storage_account: Still creating... [20s elapsed]
azurerm_storage_account.my_storage_account: Creation complete after 21s [id=/subscriptions/ac1156f0-0914-4629-90c5-323d8c283fc7/resourceGroups/rg-distinct-octopus/providers/Microsoft.Storage/storageAccounts/diagba42dc43a7313592]
azurerm_linux_virtual_machine.my_terraform_vm: Creating...
azurerm_linux_virtual_machine.my_terraform_vm: Still creating... [10s elapsed]
azurerm_linux_virtual_machine.my_terraform_vm: Creation complete after 18s [id=/subscriptions/ac1156f0-0914-4629-90c5-323d8c283fc7/resourceGroups/rg-distinct-octopus/providers/Microsoft.Compute/virtualMachines/myVM]

Apply complete! Resources: 12 added, 0 changed, 0 destroyed.

Outputs:

public_ip_address = "20.168.216.218"
resource_group_name = "rg-distinct-octopus"
tls_private_key = <sensitive>
```

Resource groups

Centennial College (CentennialCollegeEDU.onmicrosoft.com)

+ Create | Manage view | Refresh | Export to CSV | Open query | Assign tags

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

0 Unsecure resources | 0 Recommendations

No grouping | List vi

Name	Subscription	Location
cloud-shell-storage-eastus	Azure for Students	East US
NetworkWatcherRG	Azure for Students	East US
rg-distinct-octopus	Azure for Students	East US

Home >

All resources

Centennial College (CentennialCollegeEDU.onmicrosoft.com)

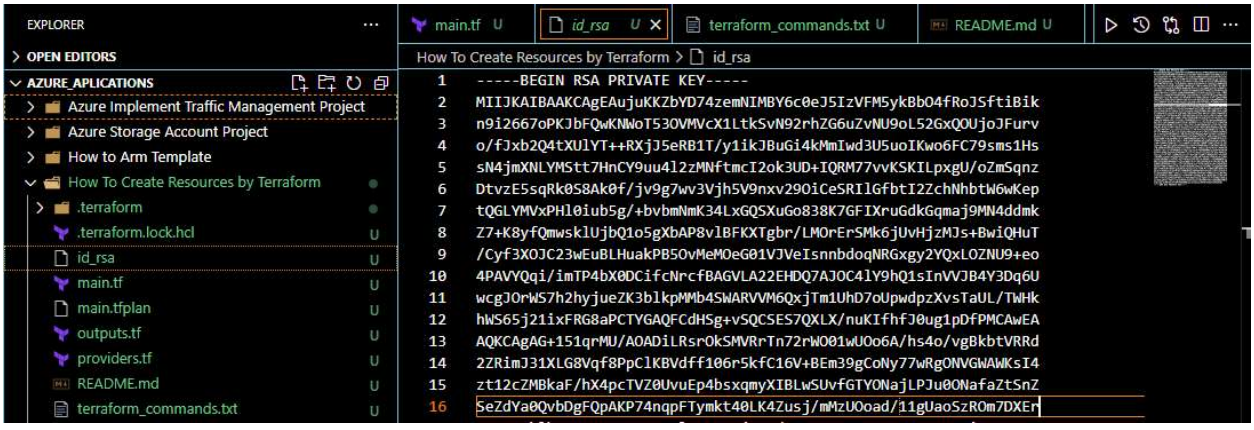
Create | Manage view | Refresh | Export to CSV | Open query | Assign tags

Filter for any field... | Subscription equals **all** | Resource group equals **all** | Add filt

0 Unsecure resources | 0 Recommendations

No grouping

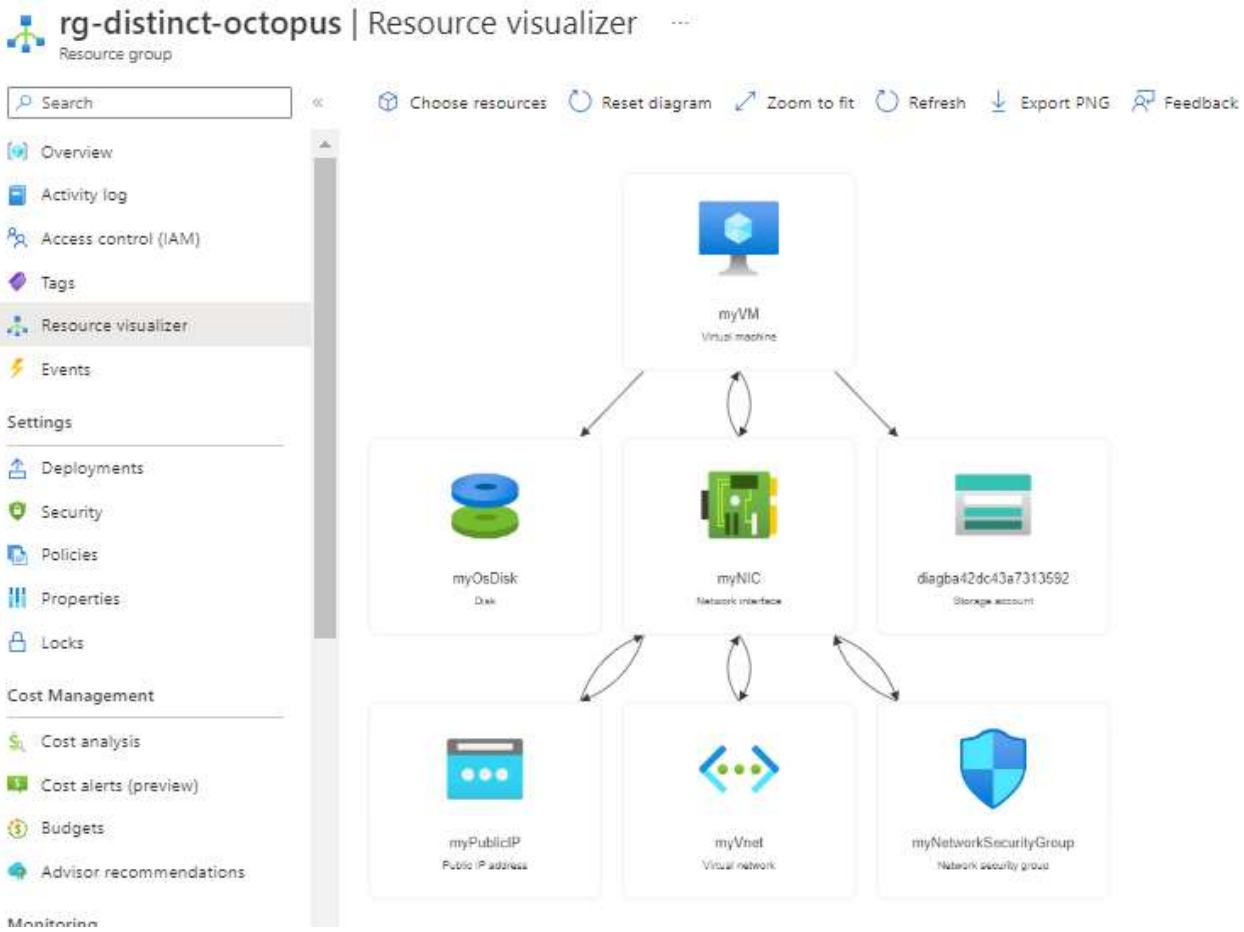
Name	Type	Resource group	Location
cs21003200046bf8711	Storage account	cloud-shell-storage-e...	East US
diagba42dc43a7313592	Storage account	rg-distinct-octopus	East US
myNetworkSecurityGroup	Network security group	rg-distinct-octopus	East US
myNIC	Network Interface	rg-distinct-octopus	East US
myOsDisk	Disk	RG-DISTINCT-OCTOP...	East US
myPublicIP	Public IP address	rg-distinct-octopus	East US
myVM	Virtual machine	rg-distinct-octopus	East US
myVnet	Virtual network	rg-distinct-octopus	East US
NetworkWatcher_eastus	Network Watcher	NetworkWatcherRG	East US

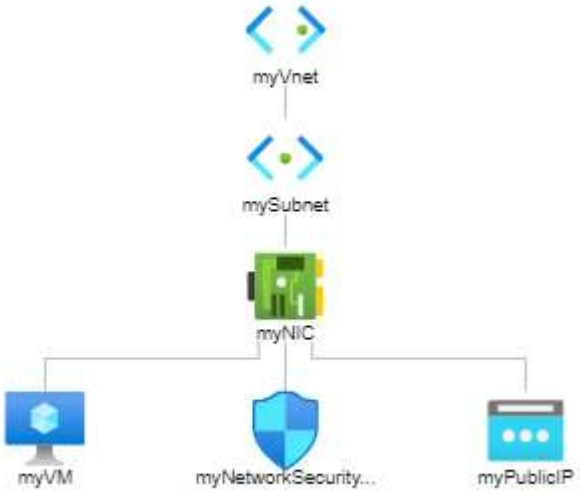



```
$ terraform output public_ip_address
"20.168.216.218"
```

```
$ ssh -i id_rsa azureuser@20.168.216.218
The authenticity of host '20.168.216.218 (20.168.216.218)' can't be established.
ED25519 key fingerprint is SHA256:aIzETMKruhIp9ZoZyNc+p1GSxKbaYRZt3kZPgYX4a/g.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '20.168.216.218' (ED25519) to the list of known hosts.
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1094-azure x86_64)

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

  System information as of Sat Dec 10 05:55:26 UTC 2022

  System load:  0.0               Processes:           108
  Usage of /:   4.5% of 28.89GB   Users logged in:     0
  Memory usage: 5%                IP address for eth0: 10.0.1.4
  Swap usage:   0%

0 updates can be applied immediately.



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.

azureuser@myvm:~$
```

```
azureuser@myvm:~$ exit
logout
Connection to 20.168.216.218 closed.
```


rg-distinct-octopus | Resource visualizer
Resource group
Search
Choose resources
Reset diagram
Zoom to fit
Refresh
Export PNG
Feedback
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
myVM
Virtual machine
myOsDisk
Disk
myNIC
Network interface
diagba42dc43a7313592
Storage account
myPublicIP
Public IP address
myVnet
Virtual network
myNetworkSecurityGroup
Network security group


myVnet
mySubnet
myNIC
myVM
myNetworkSecurity...
myPublicIP

```
Plan: 0 to add, 0 to change, 12 to destroy.

Changes to Outputs:
  - public_ip_address   = "20.168.216.218" -> null
  - resource_group_name = "rg-distinct-octopus" -> null
  - tls_private_key     = (sensitive value)
```