

Cloud Computing

H3 Hitema 2023-2025

Realiser par :

Ouafi Omar

EREKYSY Anass

Dans le cadre du projet du module de cloud Computing on est censé de de déployer avec des ressources cloud une application web simple comme un blog, un petit site e-commerce. Le site devra avoir à minima un backend connecté à une base de données. Les éventuels fichiers uploadés par les utilisateurs ne devront pas être stockés sur la vm. L'application devra être opérationnelle et accessible à tous nos collègues.

Pour faire ça on a utilisé Terraform oqui va nous faciliter le travail et qui va automatisé la création des ressources voulues, tout ça se fait dans un fichier main.tf, et dans ce document on va expliquer les différents blocks de ce fichier.

Je donne l'exemple du construction d'une maison pour faire simple :

1. Configuration de Terraform :

```
terraform {  
  required_providers {  
    azurerm = {  
      source = "hashicorp/azurerm"  
      version = "3.87.0"  
    }  
  }  
  required_version = ">= 0.14.9"  
}
```

Cette section indique à Terraform quelle version et quel plugin d'Azure on veut utiliser

2. Configuration du Provider d'Azure :

```
provider "azurerm" {  
  features {  
    resource_group {  
      prevent_deletion_if_contains_resources = false  
    }  
  }  
}
```

```
}
```

Cette section configure le AzureRM afin de dire qu'il ne faut pas empêcher la suppression des ressources même s'elles contiennent des ressources.

3. Création d'un endroit pour tout mettre :

```
resource "azurerm_resource_group" "rg" {  
  
  name    = "omar-anass-terraform-rg"  
  
  location = "westeurope"  
  
}
```

Là on crée un endroit nommé omar-anass-terraform-rg dans la région westeurope c'est considéré comme le terrain de la maison.

4. Création d'une adresse IP :

```
resource "azurerm_public_ip" "omar-terraform-public-ip" {  
  name          = "omar-anass-terraform-public-ip"  
  location      = azurerm_resource_group.rg.location  
  resource_group_name = azurerm_resource_group.rg.name  
  allocation_method = "Static"  
  sku           = "Standard"  
  
}
```

On crée une adresse ip public pour le groupe de ressource créé, comme si on donne une adresse nommée omar-anass-terraform-public-ip à notre terrain créé ci-dessus.

5. Création d'un groupe de sécurité réseau :

```
resource "azurerm_network_security_group" "nsg" {  
  name          = "omar-anass-terraform-nsg"  
  location      = azurerm_resource_group.rg.location  
  resource_group_name = azurerm_resource_group.rg.name  
  
}
```

Cela crée un groupe de sécurité réseaux dans le groupe de ressource créé

Comme si on crée une porte spéciale pour la maison afin de s'assurer que seulement les bonnes personnes peuvent venir

6. Définition d'une règle de sécurité réseaux (SSH) :

```
resource "azurerm_network_security_rule" "ssh_rule" {  
  name          = "AllowSSH"  
  priority      = 1001  
  direction     = "Inbound"  
  access        = "Allow"  
  protocol      = "Tcp"  
  source_port_range = "*"   
  destination_port_range = "22"  
  source_address_prefix = "*"   
  destination_address_prefix = "*"   
  resource_group_name   = azurerm_resource_group.rg.name  
  network_security_group_name = azurerm_network_security_group.nsg.name  
}
```

Cela autorise le Trafic SSH entrant, c'est comme on dit à Terraform d'autoriser au gens de venir à notre maison mais seulement s'ils utilisent la connexion SSH

7. Création d'un réseaux virtuel :

```
resource "azurerm_virtual_network" "vnet" {  
  name          = "omar-anass-terraform-vnet"  
  resource_group_name = azurerm_resource_group.rg.name  
  address_space   = ["10.0.0.0/16"]  
  location       = azurerm_resource_group.rg.location  
}
```

Cette section crée un réseau virtuel avec l'adresse ip 10.0.0.0/16 dans le groupe de ressource créé.

D'une façon plus simple on crée un réseau pour notre maison pour qu'elle puisse parler aux autres maisons via Internet

8. Création d'un sous réseau :

```
resource "azurerm_subnet" "subnet" {  
  name = "omar-anass-terraform-subnet"  
  resource_group_name = azurerm_resource_group.rg.name  
  virtual_network_name = azurerm_virtual_network.vnet.name  
  address_prefixes = ["10.0.0.0/24"]  
}
```

On crée un sous réseau avec une adresse ip 10.0.0.0/24 dans le réseau virtuel créé précédemment.

Comme si on construit une chambre spéciale dans notre maison.

9. Création d'une interface réseau :

```
resource "azurerm_network_interface" "nic" {  
  name = "omar-anass-terraform-nic"  
  location = azurerm_resource_group.rg.location  
  resource_group_name = azurerm_resource_group.rg.name  
  ip_configuration {  
    name = "omar-anass-terraform-nic-ipconfig"  
    subnet_id = azurerm_subnet.subnet.id  
    private_ip_address_allocation = "Dynamic"  
    public_ip_address_id = azurerm_public_ip.omar-terraform-public-ip.id  
  }  
}
```

Cela crée une interface réseau liée à l'adresse ip public et au groupe de ressource créé. Comme si on ajoute un téléphone spécial pour que notre maison parle aux autres maisons.

10. Ajout de la sécurité au niveau de l'interface réseau :

```
resource "azurerm_network_interface_security_group_association"
  "nsg_association" {
    network_interface_id = azurerm_network_interface.nic.id
    network_security_group_id = azurerm_network_security_group.nsg.id
  }
```

Ici on s'assure que le téléphone est protégée pour éviter que les mauvaises choses arrivent.

11. Création d'une machine virtuelle Linux :

```
resource "azurerm_linux_virtual_machine" "vm" {
  name = "reservation-vm-omar-anass"
  resource_group_name = azurerm_resource_group.rg.name
  location = azurerm_resource_group.rg.location
  size = "Standard_F2"
  admin_username = var.vm_admin_username
  admin_password = var.vm_admin_password
  disable_password_authentication = false
  network_interface_ids = [
    azurerm_network_interface.nic.id,
  ]
  os_disk {
    caching = "ReadWrite"
    storage_account_type = "Standard_LRS"
  }
  source_image_reference {
    publisher = "Canonical"
    offer = "0001-com-ubuntu-server-jammy"
```

```
sku    = "22_04-lts"
```

```
version = "latest"
```

```
}
```

```
}
```

On demande à Terraform de mettre en place un ordinateur dans notre maison qui va faire et servir pour des choses magiques.

12. Création d'un serveur MySQL

```
resource "azurerm_mysql_server" "omar-anass-terraform-mysql" {
```

```
  name          = "omar-anass-terraform-mysql"
```

```
  location      = azurerm_resource_group.rg.location
```

```
  resource_group_name = azurerm_resource_group.rg.name
```

```
  administrator_login      = var.db_admin_username
```

```
  administrator_login_password = var.db_admin_password
```

```
  sku_name = "B_Gen5_2"
```

```
  storage_mb = 5120
```

```
  version = "5.7"
```

```
  auto_grow_enabled      = true
```

```
  backup_retention_days  = 7
```

```
  geo_redundant_backup_enabled = false
```

```
  infrastructure_encryption_enabled = false
```

```
  public_network_access_enabled = true
```

```
  ssl_enforcement_enabled = true
```

```
  ssl_minimal_tls_version_enforced = "TLS1_2"
```

```
}
```

on construit notre garage pour stocker les information importantes.

13. Création d'un compte de stockage :

```
resource "azurerm_storage_account" "omar-anass-terraform-storage" {  
  name                = "omaranassstorage"  
  resource_group_name = azurerm_resource_group.rg.name  
  location            = azurerm_resource_group.rg.location  
  account_tier        = "Standard"  
  account_replication_type = "LRS"  
}
```

Cela crée un compte de stockage comme si on crée un dépôt ou une cave dans notre maison ou on stocke les choses importantes.

14. Création d'un conteneur de stockage :

```
resource "azurerm_storage_container" "omaranasscontainer" {  
  name                = "omaranasscontainer"  
  storage_account_name = azurerm_storage_account.omar-anass-terraform-  
storage.name  
  container_access_type = "private"  
}
```

Là on dit à Terraform de créer une boîte dans notre compte de stockage pour stocker nos jouets.