Exercise 4.06 - Typical infrastructure

The objective of this exercise is to help you assessing your capability to build a typical infrastructure using Terraform.

NOTE: We recommend to start from scratch. Again, the objective is for you to check you get the logic right. Nevertheless, most of the files in the root can be copied from previous exercise (EX 4.05)

Step 1 - Create the structure

- Create the "usual" (suspects) files: main.tf, outputs.tf, providers.tf, terraform.tfvars, variables.tf and versions.tf
- Create a folder structure to support the 2 modules
- Create in each subfolders the following files: main.tf, outputs.tf and variables.tf

Step 2 - ROOT structure

- The main.tf file should contain:

  - A declaration of a resource group
  - A declaration of the "networking" module
  - A declaration of the "application" module
  - The attributes will be set whilst you are populating the modules

- The outputs.tf file should contain:

  - An output value representing the public IP of the server that will be accessed through SSH.

- 3 variables are required for this exercice: the cost center, the location and the suffix.

Step 3 - Networking module

- 6 resources are required to elaborate a "decent" network infrastructure. The tables below provide some details about each of them:

- Network Interface (azurerm_network_interface)

Attribute	Value	Comment
location	?	from root's resource group
name	"nic-"+suffix	from root module
resource group name	?	from root's resource group

  - The network interface will require an IP configuration. The most important attributes are:

Attribute	Value	Comment

Attribute	Value	Comment
subnet id	?	from subnet
private ip address allocation	"Dynamic"	-
public ip address id	?	from public ip

- Network Security Group (azurerm_network_security_group)

Attribute	Value	Comment
location	?	from root's resource group
name	"scg-"+suffix	from root module
resource group name	?	from root's resource group

  - The network security group will require a security rule. The most important attributes are:

Attribute	Value	Comment
priority	1001	-
direction	"Inbound"	-
access	"Allow"	-
protocol	"Tcp"	-

- Network Interface/Security Group (azurerm_network_interface_security_group_association)

Attribute	Value	Comment
network_interface_id	?	from network interface
network_security_group_id	?	from network security group

- Public IP (azurerm_public_ip)

Attribute	Value	Comment
allocation_method	"Dynamic"	-
location	?	from root's resource group
name	"pip-"+suffix	from root module
resource group name	?	from root's resource group

- Subnet (azurerm_subnet)

Attribute	Value	Comment
address_prefixes	["10.0.2.0/24"]	-

Attribute	Value	Comment
name	"vnt-"+suffix	from root module
resource group name	?	from root's resource group
virtual network name	?	virtual network name

- Virtual Network (azurerm_virtual_network).

Attribute	Value	Comment
address_space	["10.0.0.0/16"]	-
location	?	from root's resource group
name	"vnt-"+suffix	from root module
resource group name	?	from root's resource group

- Make sure you export the network interface identifier as an output value

Step 4 - Application module

- Only one resource is required to cover the application layer. The tables below provide some additional details:

- Virtual Machine (azurerm_linux_virtual_machine)

Attribute	Value	Comment
admin password	?	whatever you like
admin username	?	whatever you like
disable_password_authentication	?	from root's resource group
location	?	from root's resource group
name	"lvm-"+suffix	from root module
resource group name	?	from root's resource group
size	"Standard_B1s"	-

  ○ Add the network interface identifier (from the "networking" module) to the "network interface ids" collection

  ○ The configuration for "os_disk" is as follow:

Attribute	Value	Comment
caching	"ReadWrite"	-
storage_account_type	"Standard_LRS"	-

- The configuration for image (source_image_reference) is as follow:

Attribute	Value	Comment
offer	"Canonical"	-
publisher	"UbuntuServer"	-
sku	"18.04-LTS"	-
version	"latest"	-

- Make sure you export the public IP address (public_ip_address) from the VM as an output value

Step 5 - Back to ROOT module

- Adapt accordingly (if not done yet) the module declarations based on the arguments/variables requirements.

Step 6 - Proceed to deployment

- Proceed to deployment
- Evaluate ssh access using ssh «admin_username»@public_ip