



Introduction to Hashicorp Terraform for Azure – Part 1

Arnaud Lheureux

Cloud Chief Security Officer

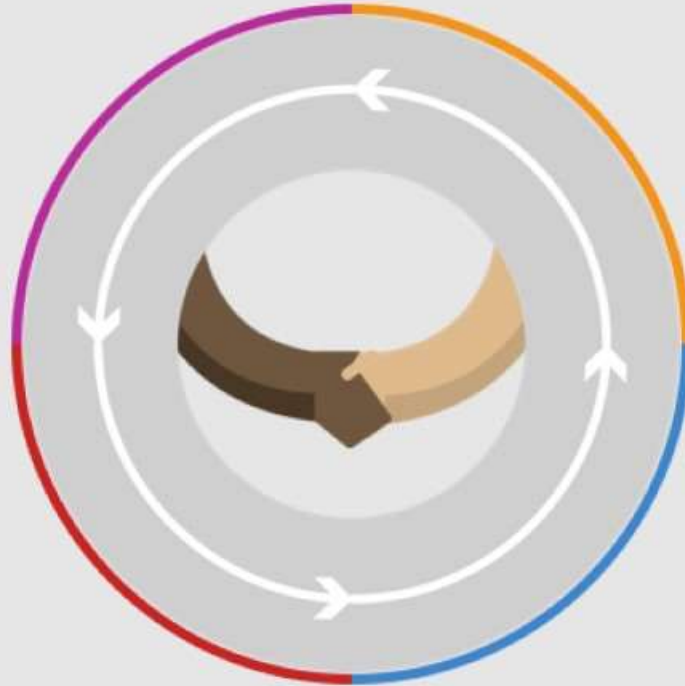
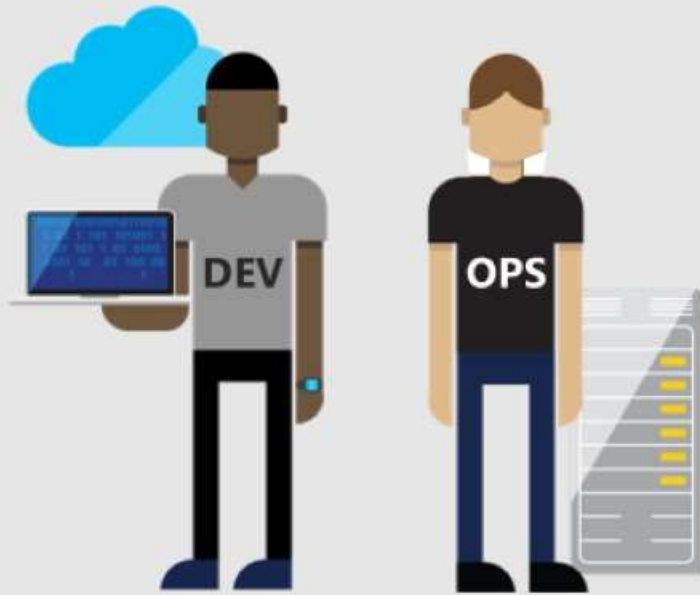
APAC Office of the CTO

One Commercial Partner

Twitter: @arnaudLheureux

DevOps: The Three Stage Conversation

DevOps = People + Process + Tools



1

People

2

Process

3

Tools

Infrastructure as Code with Terraform?

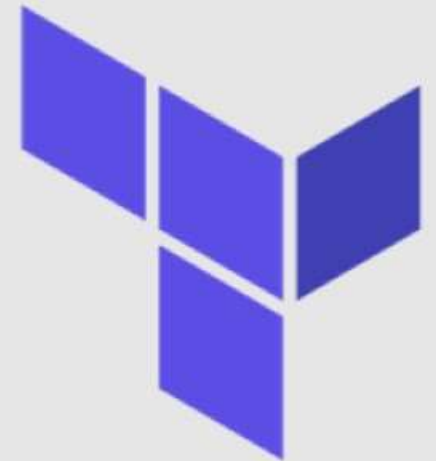
- ✓ Reproducible Environments
- ✓ Automation – CI/ CD
- ✓ Trackable – GitHub
- ✓ Language - HCL
- ✓ Workflow
- ✓ Providers

☐ Apply same config across clouds



Why people love Terraform?

- Clean and easy code to write and maintain
- Fully declarative configuration
- Version control on infra
- Implicit dependencies management – explicit can be forced
- Ecosystem of providers and skilled personnel



Azure Provider

Authentication

Azure CLI

Service Principal

Azure Managed Identities

Arguments

```
provider "azurerm" {  
  subscription_id = "{My Subscription ID}"  
  client_id       = "{My Service Principle ID}"  
  client_secret   = "{My Service Principle Password}"  
  tenant_id      = "{My Tenant ID}"  
}
```

Environment Variables

Resources & Data Sources



Azure Resources & Datasources

```
# Configure the Azure Provider
provider "azurerm" { }

# Create a resource group
resource "azurerm_resource_group" "network" {
  name     = "production"
  location = "West US"
}

# Create a virtual network within the resource group
resource "azurerm_virtual_network" "network" {
  name            = "production-network"
  address_space   = ["10.0.0.0/16"]
  location        = "${azurerm_resource_group.network.location}"
  resource_group_name = "${azurerm_resource_group.network.name}"

  subnet {
    name           = "subnet1"
    address_prefix = "10.0.1.0/24"
  }

  subnet {
    name           = "subnet2"
    address_prefix = "10.0.2.0/24"
  }

  subnet {
    name           = "subnet3"
    address_prefix = "10.0.3.0/24"
  }
}
```

Provisioning for Azure IaaS

Compute (VMSS, Disk, Image, Snapshot, ...)

Networking (Vnet, LB, DNS, ...)

Azure Active Directory

Database (MySQL, PostgreSQL, SQL)

Monitoring

Storage (Storage Account, Blob, Share, ...)

...

Provisioning for Azure PaaS

Containers (AKS, ACI)

Web Apps

CosmosDB

Data Lake

Logic Apps

KeyVault

...

Catch-All

ARM Template

Assembly

Cloud Shell

Terraform integration

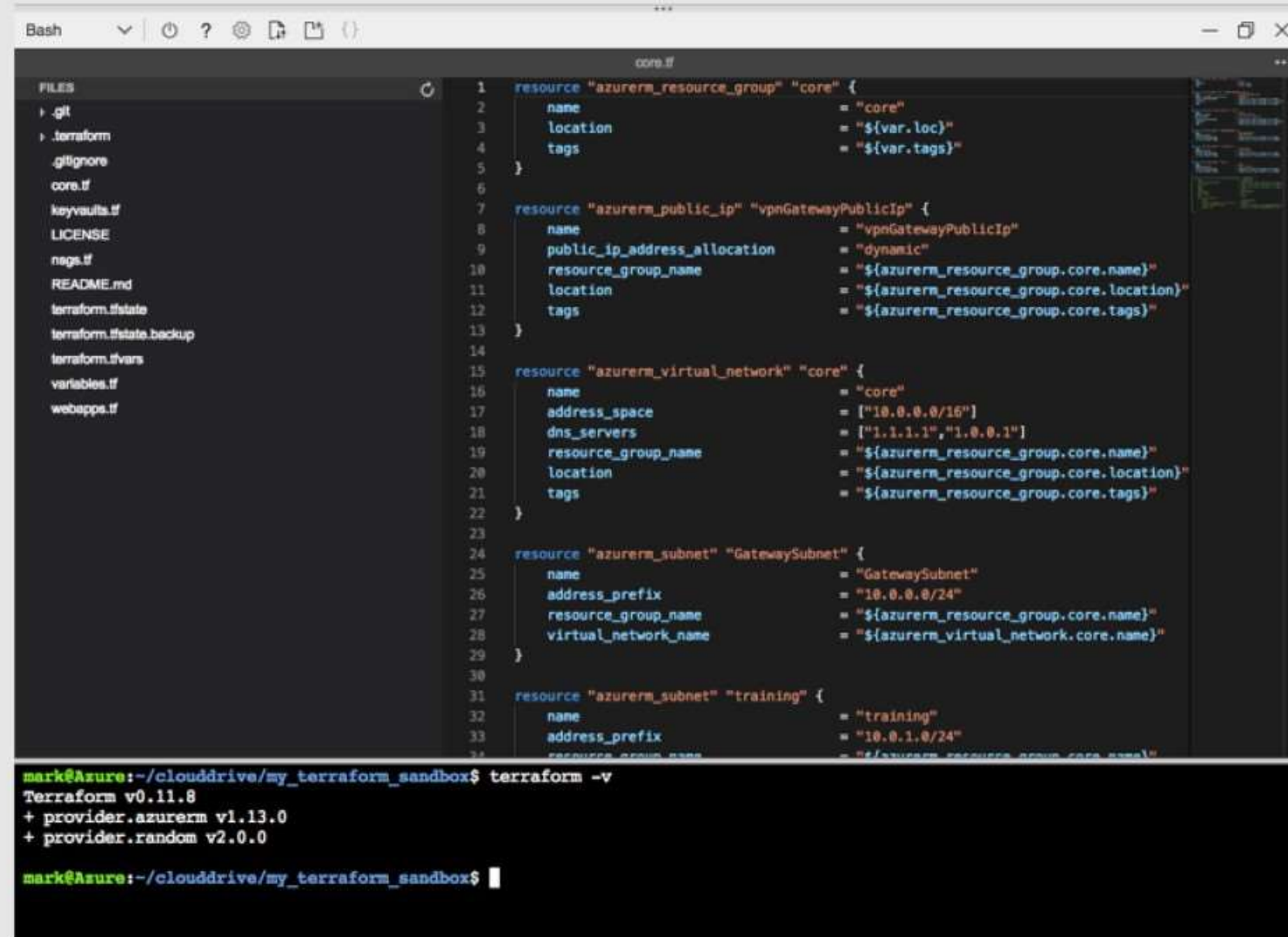
Editor with Terraform Syntax

Highlighting

Visual Studio Code

Terraform extension

Azure Terraform extension



The screenshot displays the Visual Studio Code interface. On the left, a file explorer shows a project structure with files like .git, .terraform, .gitignore, core.tf, keyvaults.tf, LICENSE, nags.tf, README.md, terraform.tfstate, terraform.tfstate.backup, terraform.tfvars, variables.tf, and webapps.tf. The main editor area shows the content of core.tf, which defines several Azure resources in HCL syntax:

```
1 resource "azurerm_resource_group" "core" {
2   name      = "core"
3   location  = "${var.loc}"
4   tags      = "${var.tags}"
5 }
6
7 resource "azurerm_public_ip" "vpnGatewayPublicIp" {
8   name                        = "vpnGatewayPublicIp"
9   public_ip_address_allocation = "dynamic"
10  resource_group_name         = "${azurerm_resource_group.core.name}"
11  location                    = "${azurerm_resource_group.core.location}"
12  tags                        = "${azurerm_resource_group.core.tags}"
13 }
14
15 resource "azurerm_virtual_network" "core" {
16   name                = "core"
17   address_space       = ["10.0.0.0/16"]
18   dns_servers         = ["1.1.1.1", "1.0.0.1"]
19   resource_group_name = "${azurerm_resource_group.core.name}"
20   location             = "${azurerm_resource_group.core.location}"
21   tags                = "${azurerm_resource_group.core.tags}"
22 }
23
24 resource "azurerm_subnet" "GatewaySubnet" {
25   name            = "GatewaySubnet"
26   address_prefix  = "10.0.0.0/24"
27   resource_group_name = "${azurerm_resource_group.core.name}"
28   virtual_network_name = "${azurerm_virtual_network.core.name}"
29 }
30
31 resource "azurerm_subnet" "training" {
32   name            = "training"
33   address_prefix  = "10.0.1.0/24"
34   resource_group_name = "${azurerm_resource_group.core.name}"
35 }
```

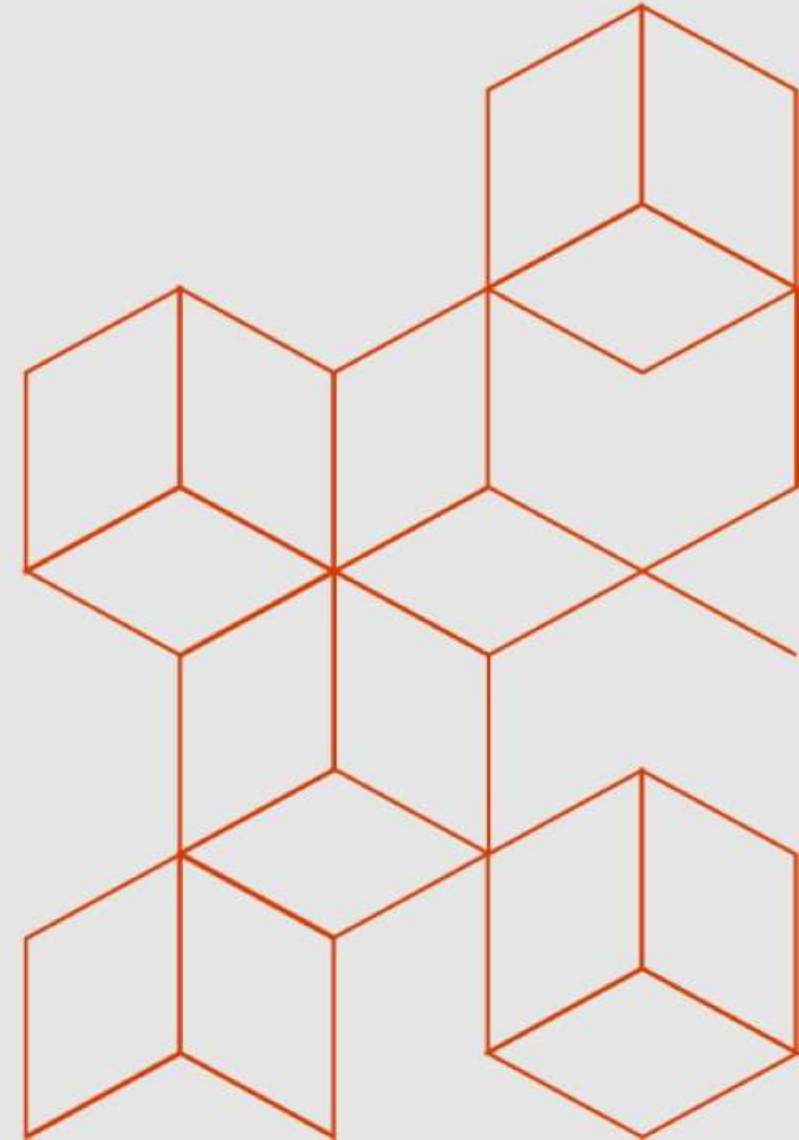
At the bottom, a terminal window shows the output of the command `terraform -v`:

```
mark@Azure:~/clouddrive/my_terraform_sandbox$ terraform -v
Terraform v0.11.8
+ provider.azurearm v1.13.0
+ provider.random v2.0.0

mark@Azure:~/clouddrive/my_terraform_sandbox$
```


Demo 1

Authoring



Language fundamentals

- Variables

- Values can be supplied as a .tfvars file containing simple key/value pairs, env variables, or command parameters.

- Functions

- String and math (all the usual)
- Count – simple method for deploying multiple resources
- Conditional "\${var.env == "production" ? var.prod_subnet : var.dev_subnet}"
- CIDR

- Provisioners

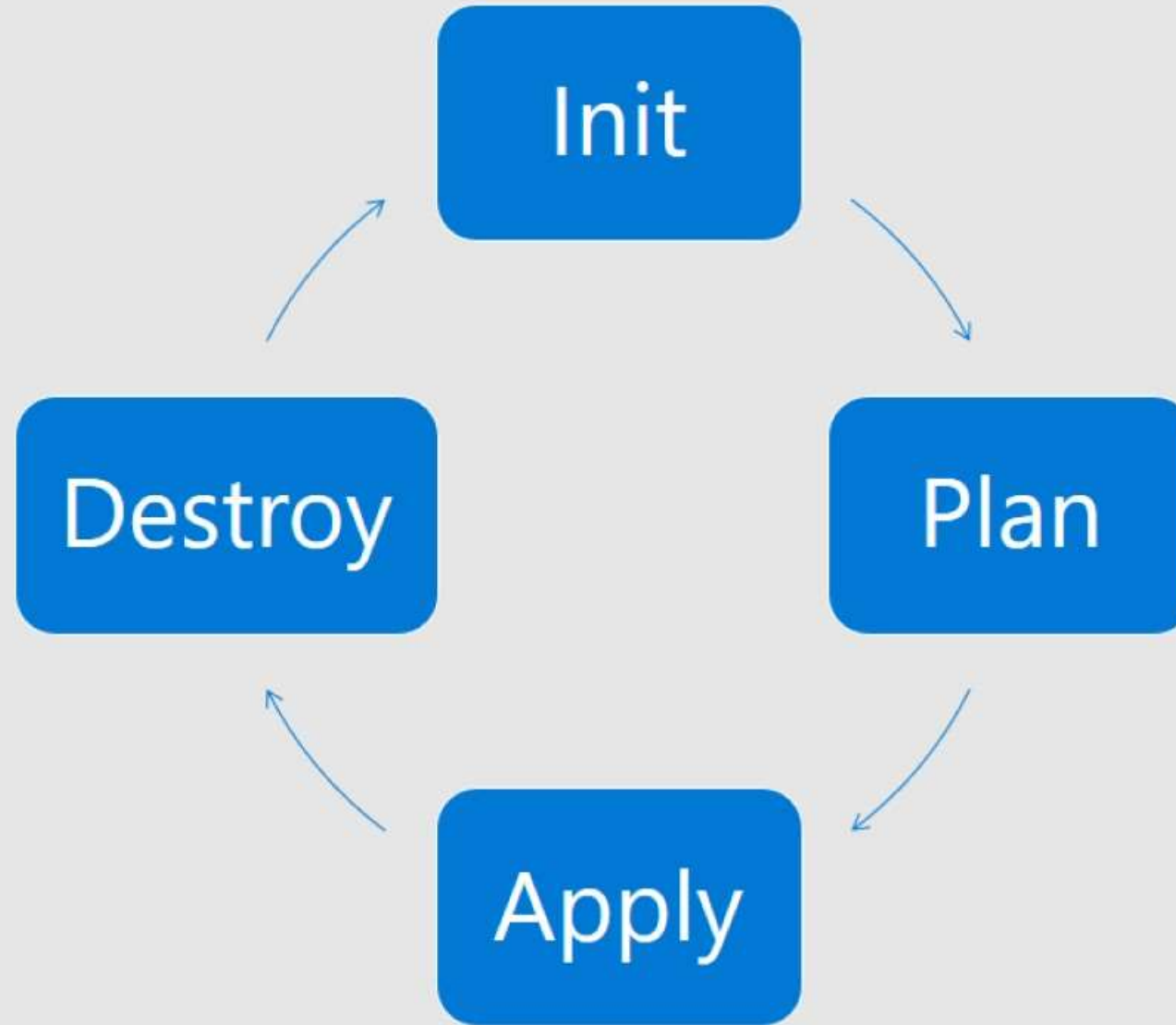
- local-exec, remote-exec, file

```
module "vaultstorage" {
  source = "../modules/storage/account"

  name          = "${var.vault_storage_account_name}"
  resource_group_name = "${azurerm_resource_group.storage_rg.name}"
  location      = "${azurerm_resource_group.storage_rg.location}"

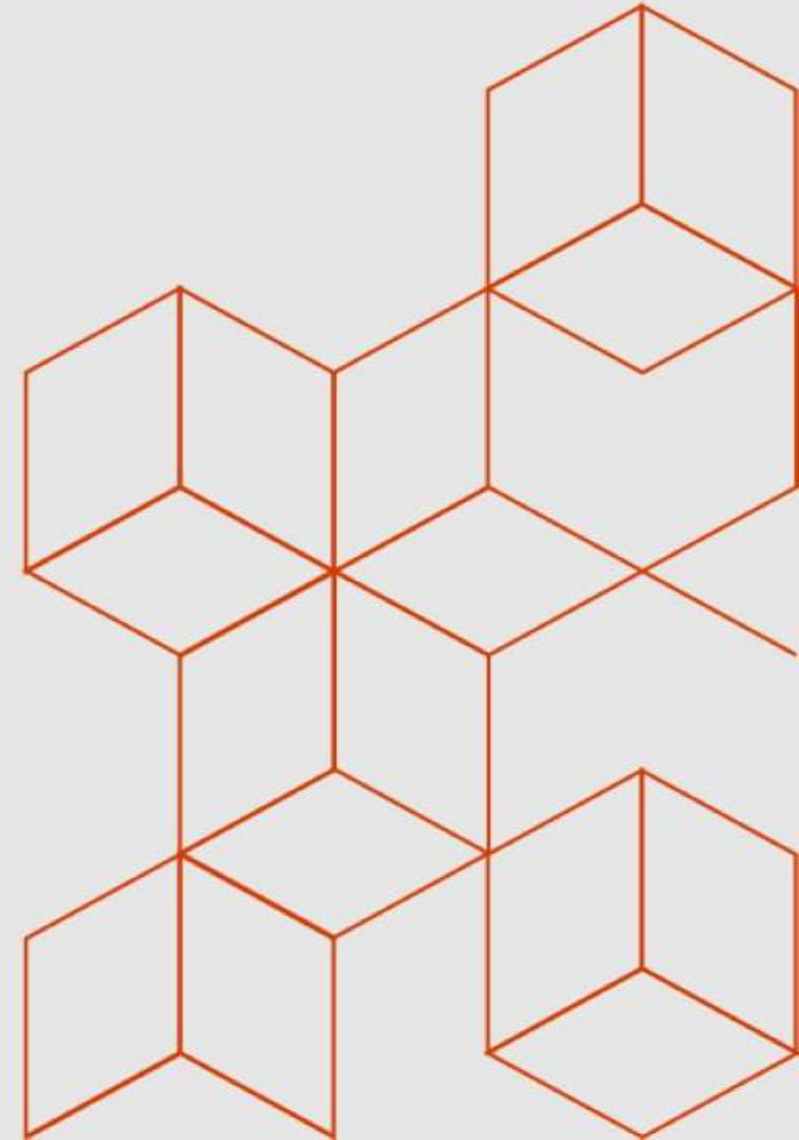
  tags {
    Application = "Vault"
    Environment = "SS-Prod"
    Category    = "Storage Account"
  }
}
```

Workflow



Demo 2

Workflow



State

Compare configuration to current

Responsible for mapping "azure_virtual_machine" "vm" to "/subscriptions/dcf628c7-fc9d-4e40-af2c-5c963345a237/resourceGroups/BDIterraformdemo/providers/Microsoft.Compute/virtualMachines/BDlvm-vm"

Tracks dependencies between resources

Knows that if the VM is deleted, to also delete the Disk(s)

Provides the ability to pass in previous deployments as parameters

Develop plan

Local - default

terraform.tfstate

Show

Import

Collaboration

Workspaces

Multiple deployments with common backend and separate state

Backends

Partial Configurations

- Interactive

- File (.tfvars)

- Command-line

Standard backends

- AzureRM (Blob storage), Artifactory, Consul

Enhanced backend

- Terraform Enterprise

Azurerm backend

Standard Backend with state locking & consistency checking

Azure Storage (Blob)

remote-state.tf

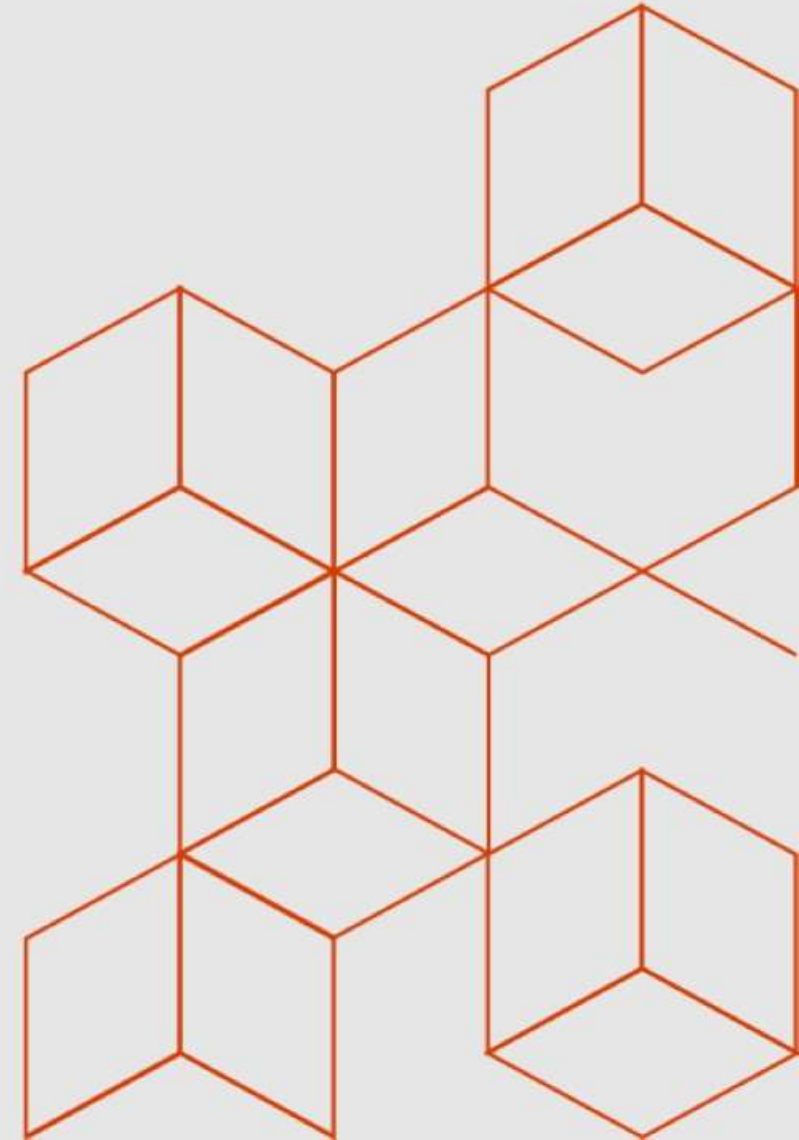
```
terraform {  
  backend "azurerm" {}  
}
```

```
.backend.tfvars
```

```
storage_account_name = "tfbackend4mcg"
container_name = "tfstate"
key = "sandbox.terraform.tfstate"
access_key = ""
```

Demo 3

State



Abstraction

Modules



Modules

Reusable template code

- Container of resources that are used together

```
module "MyApp-Network-AWS" {  
  source      = "terraform-aws-modules/vpc/aws"  
  
  name        = "network1"  
  
  tags = {  
    environment = "dev"  
    costcenter  = "it"  
  }  
}
```

```
resource "module" "MyApp-Network-Azure" {  
  name      = "MyApp-Network-Azure"  
  source    = "Azure/network/azurerm"  
  
  vnet_name = "network1"  
  location  = "westus"  
  
  tags = {  
    environment = "dev"  
    costcenter  = "it"  
  }  
}
```


FILTER BY

azurerm



Verified


loadbalancer

azurerm



Terraform Azure RM Module for Load Balancer

Version 1.2.1 - By Azure


network

azurerm



Terraform Azure RM Module for Network

Version 2.0.0 - By Azure


computegroup

azurerm



Terraform Azure RM Compute Group Module

Version 2.1.0 - By Azure


compute

azurerm



Terraform Azure RM Compute Module

Version 1.2.1 - By Azure


database

azurerm



Terraform Azure RM Module for Database

Version 1.1.0 - By Azure


consul

azurerm



A Terraform Module for how to run Consul on AzureRM using Terraform...

Version 0.0.5 - By hashicorp


vault

azurerm



A Terraform Module for how to run Vault on AzureRM using Terraform...

Version 0.0.2 - By hashicorp


kubernetes

azurerm



Install a Kubernetes cluster the CoreOS Tectonic Way: HA, self-hosted, RBAC,...

Version 1.8.9-tectonic.1 - By coreos


nomad

azurerm

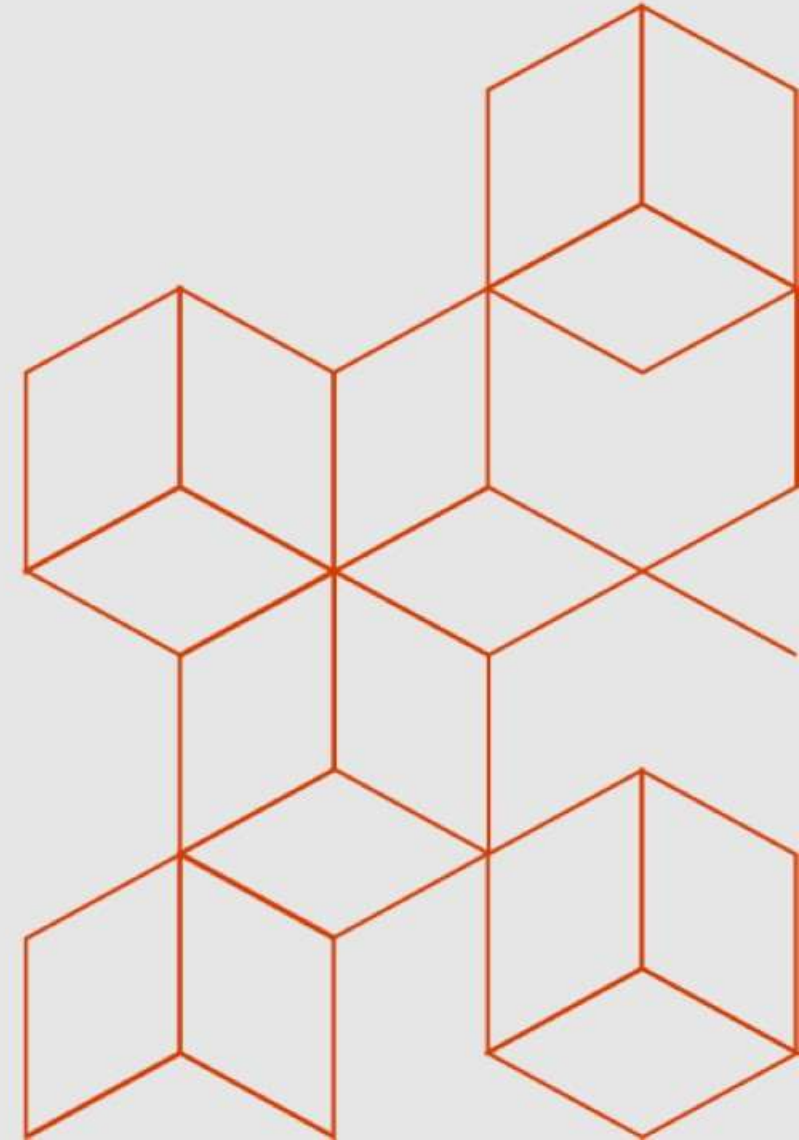


A Terraform Module for how to run Nomad on AzureRM using Terraform...

Version 0.0.1 - By hashicorp

Demo 4

Modules

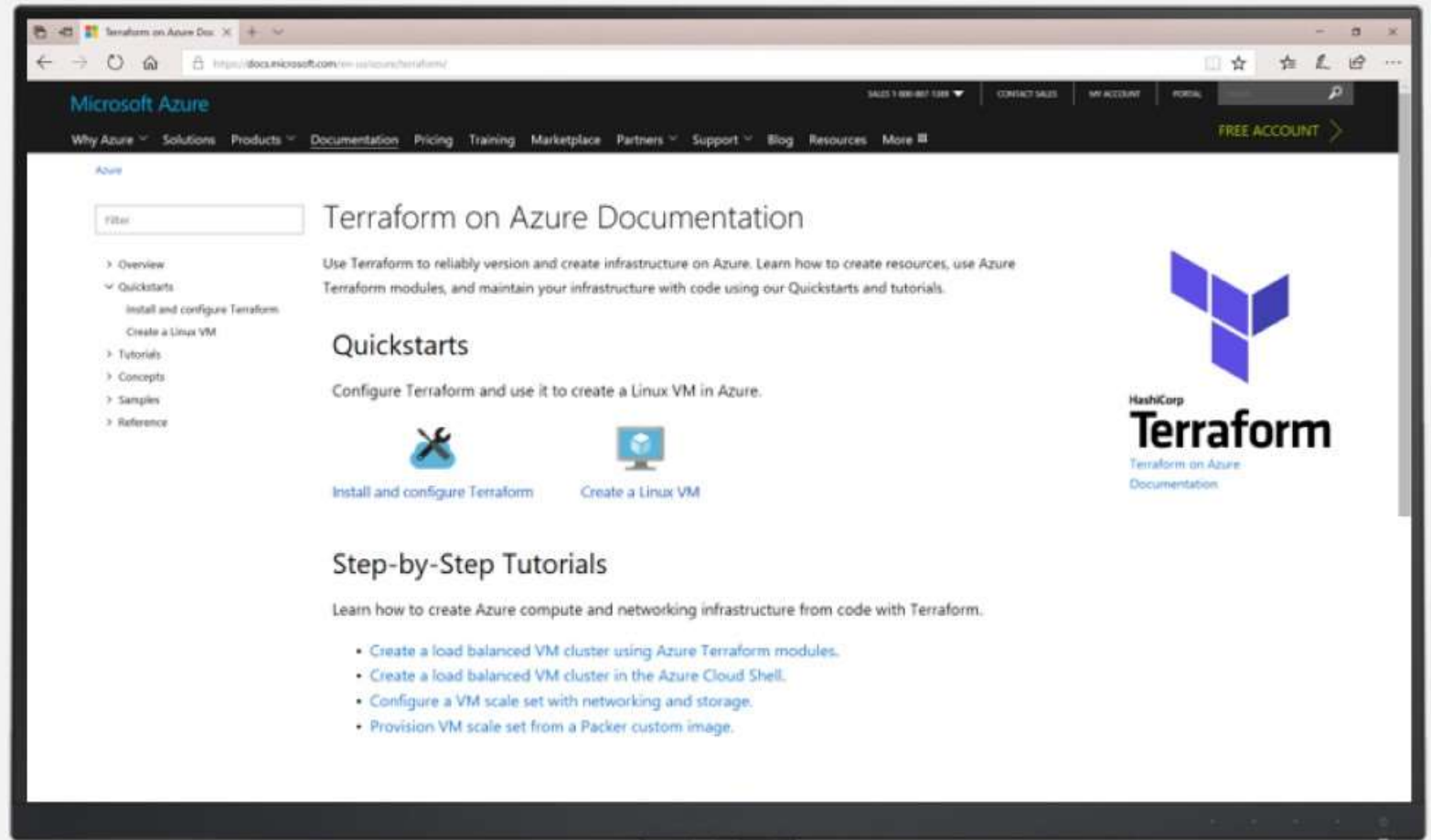


Developer Hub for Terraform

<http://aka.ms/tfhub>

→ <https://docs.microsoft.com/azure/Terraform/>

→ The best place to find technical guidance for Terraform on Azure



Up next:

Azure Landing Zones Using Terraform



Thanks for your attention!



Arnaud Lheureux, CISSP

Cloud Security Officer for APAC Partners

<https://aka.ms/arnaud>

Twitter : @arnaudLheureux