# **Azure Automation using Terraform**

# Terraform: Hashicorp

* Open-source Infrastructure as code software tool.
* High-level configuration language known as Hashicorp configuration Language (HCL) 0r optionally JSON.
* Supports Multi cloud option. (AWS, GCP, Azure).

# Azure Virtual WAN:

https://docs.microsoft.com/en-us/azure/virtual-wan/virtual-wan-site-to-site-portal

* Azure Virtual WAN is a new network service that allows you to optimize and automate the branch-to-branch connectivity through Azure. Azure regions serve as hubs that you can choose to connect your branches to.
* It enables global transit network architecture based on a classic hub-and-spoke connectivity model where the cloud hosted network 'hub' enables transitive connectivity between endpoints that may be distributed across different types of 'spokes'.

# Before you begin:

* Create a VNet to be connected.
* Check if any VPN Gateways or express route is connected to VNet. If so remove all the connections.
* Obtain an IP address range for your hub region. Hub cannot overlap with any of your existing virtual networks.

# Deployment:

* Create a virtual WAN -- Using Terraform
* Create a hub -- Using Terraform
* Create a site -- Using Terraform
* Connect a site to a hub --Using Terraform
* Create a VPN site and connect to a hub -- Using Powershell
* Connect a VNet to a hub -- Using Powershell in Automation Accounts

# SPN Creation:

az ad sp create-for-rbac --role="Contributor" --scopes="/subscriptions/SUBSCRIPTION\_ID"

"appId": "b59396eb-1925-4328-b719-640c420d739d",

"displayName": "azure-cli-2020-01-24-04-42-30",

"name": "http://azure-cli-2020-01-24-04-42-30",

"password": "f9dd9fb0-1f8a-4a54-998c-4f8c14a56c72",

"tenant": "7571a489-bd29-4f38-b9a6-7c880f8cddf0"

subscription\_id = "b75f9849-8afd-4e03-95ce-aebeb22e23e6"

client\_id = "b59396eb-1925-4328-b719-640c420d739d"

client\_secret = "f9dd9fb0-1f8a-4a54-998c-4f8c14a56c72"

tenant\_id = "7571a489-bd29-4f38-b9a6-7c880f8cddf0"

Subscription ID = subscription\_id

App id = Client\_id

Password = client\_secret

Tenant ID = tenant\_id

# Terraform commands:

terraform init

terraform validate

terraform plan

terraform apply

terraform plan -var-file="C:\Karthik\Infosys\Terraform\WAN\authentication.tfvars" -var-file="C:\Karthik\Infosys\Terraform\WAN\env.tfvars"

terraform apply -var-file="C:\Karthik\Infosys\Terraform\WAN\authentication.tfvars" -var-file="C:\Karthik\Infosys\Terraform\WAN\env.tfvars"

terraform apply -var-file=" C:\Users\karthikeya.tandra\Desktop\Infosys\PPCservices\Servicebus\auth.tfvars" -var-file=" C:\Users\karthikeya.tandra\Desktop\Infosys\PPCservices\Servicebus \env.tfvars"