**END TO END WEBSITE HOSTING**

**Project Overview**

This project focuses on architecting and implementing a simple web site from end to end through Azure which enables users to access the site from all over the world and also includes geography specific contents.

**Architecture diagram**

DNS zone

**DNS-label(domain name)**

Traffic manager

**TM-websrvr-01**

Application gateway

**Lb-websrvr-IN**

Application gateway

**Lb-websrvr-US**

Virtual Machine

**Vm-websrvr-US-01**

Virtual Machine

**Vm-websrvr-US-02**

Virtual Machine

**Vm-websrvr-IN-01**

Virtual Machine

**Vm-websrvr-IN-02**

**Objective**

* Task1: Creating Resource Groups
* Task2: Creating 4 Linux Virtual Machines
* Task3: Install Web servers into virtual machines and test
* Task4: Implement application gateway in front of VM’s and test if working
* Task5: Implement Traffic manager to distribute traffic to load balancers based on Geography
* Task6: Create DNS zone and the domain name .Also connect the traffic manager endpoint to complete the final step

**Resources/Topics covered as part of this Project**

* Resource Group
* Virtual Machine
* Security groups
* Apache Web server
* Power shell
* Virtual network
* subnets
* OS disks
* Network interface
* Ports:http,ssh
* Application gateway(load balancer)
* Public ip addresses
* Traffic manager
* DSN zone
* Name servers

**Task 1: Creating Resource Groups**

1. Sign in to the Azure portal.
2. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
3. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.
4. From the Cloud Shell pane, run the following to create the first resource group that will be hosting a pair of virtual machines (replace the [Azure\_region] placeholder with the name of an Azure region where you intend to deploy Azure virtual machines)(you can use the "(Get-AzLocation).Location" cmdlet to get the region list):

$location = 'EastUS2'

$rgName = 'rg-websrvr-01'

New-AzResourceGroup -Name $rgName -Location $location

1. From the Cloud Shell pane, run the following to create the second resource group that will be hosting the second pair of virtual machines.

$location = 'CentralIndia'

$rgName = ' rg-websrvr-02’

New-AzResourceGroup -Name $rgName -Location $location

**Task2: Creating 4 Linux Virtual Machines**

In this task we will be creating 1st and 2nd Linux virtual machines in EastUS2 region (rg-websrvr-01)

1. In the portal search and select **Virtual Machines**
2. Click **add** on the Virtual machines blade and **Virtual machine**
3. Fill the Basics tab as below and leave others as default

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Subscription | Your subscription |
| Resource Group | rg-websrvr-01 |
| Virtual machine name | Vm-websrvr-US-01 |
| Region | East US2 |
| Availability options | No infrastructure redundancy required |
| Image | Ubuntu Server 16.04 LTS –Gen1 |
| Size | Standard\_B1s- 1vcpu, 1Gib memory |
| Authentication type | Password |
| UserNAme | Student |
| Password | PassWord.1234 |

1. Click **Next:Disks.** Select **OS disk type : Standard SSD**
2. Click **Next: Networking** and create a new virtual network**.** Click **OK** once the details are entered and leave other fields as default. Now go to **Management** tab

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Virtual Network | Click “create new” |

In the Create Virtual Network Window ,provide the below values

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Address range | 10.1.0.0/16 |
| Subnet Name | Default |
| Address Range | 10.1.1.0/24 |
| Subnet Name (add 2nd subnet) | Gtw-subnet-US |
| Address Range | 10.1.2.0/24 |

1. In **Management tab, Disable** Boot diagnostics and go to **review + create** tab
2. Once validation is passed, click **Create** to spin up the VM.

**Create the second VM**

1. Repeat steps 1 and 2
2. Fill the Basics tab as below and leave others as default

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Subscription | Your subscription |
| Resource Group | rg-websrvr-01 |
| Virtual machine name | Vm-websrvr-US-02 |
| Region | East US2 |
| Availability options | No infrastructure redundancy required |
| Image | Ubuntu Server 16.04 LTS –Gen1 |
| Size | Standard\_B1s- 1vcpu, 1Gib memory |
| Authentication type | Password |
| UserNAme | Student |
| Password | PassWord.1234 |

1. Repeat step 4.
2. In Step 5 no need to create a new vnet as it would automatically take the existing one. Just review the details and continue steps 6 and 7 to spin up 2nd VM

Creating the 3rd and 4th VM

In this task we will be 3rd and 4th vm in Central India Region (rg-websrvr-01)

1. In the portal search and select **Virtual Machines**
2. Click **add** on the Virtual machines blade and **Virtual machine**
3. Fill the Basics tab as below and leave others as default

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Subscription | Your subscription |
| Resource Group | rg-websrvr-02 |
| Virtual machine name | Vm-websrvr-IN-01 |
| Region | Central India |
| Availability options | No infrastructure redundancy required |
| Image | Ubuntu Server 16.04 LTS –Gen1 |
| Size | Standard\_B1s- 1vcpu, 1Gib memory |
| Authentication type | Password |
| UserNAme | Student |
| Password | PassWord.1234 |

1. Click **Next:Disks.** Select **OS disk type : Standard SSD**
2. Click **Next: Networking** and create a new virtual network**.** Click **OK** once the details are entered and leave other fields as default. Now go to **Management** tab

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Virtual Network | Click “create new” |

In the Create Virtual Network Window ,provide the below values

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Address range | 10.3.0.0/16 |
| Subnet Name | Default |
| Address Range | 10.3.1.0/24 |
| Subnet Name (add 2nd subnet) | Gtw-subnet-IN |
| Address Range | 10.3.2.0/24 |

1. In **Management tab, Disable** Boot diagnostics and go to **review + create** tab
2. Once validation is passed, click **Create** to spin up the VM.

**Create the 4th VM**

1. Repeat steps 1 and 2
2. Fill the Basics tab as below and leave others as default

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Subscription | Your subscription |
| Resource Group | rg-websrvr-02 |
| Virtual machine name | Vm-websrvr-IN-02 |
| Region | Central India |
| Availability options | No infrastructure redundancy required |
| Image | Ubuntu Server 16.04 LTS –Gen1 |
| Size | Standard\_B1s- 1vcpu, 1Gib memory |
| Authentication type | Password |
| UserNAme | Student |
| Password | PassWord.1234 |

1. Repeat step 4.
2. In Step 5 no need to create a new vnet as it would automatically take the existing one. Just review the details and continue steps 6 and 7 to spin up 4th VM

**Task3: Install Web servers into virtual machines and test**

In this task we would be installing Apache Webserver in the 4 VM’s created in previous task and test them if they are working as expected.

1. Copy the public ip of the VM which will be used to login to the machine
2. On your computer , download putty from web and Login to the Linux VM we created using putty and ssh through port 22 by pasting the public ip of the vm and port as 22
3. Provide username and password to login to the machine
4. Update the ubunto up to date using command “sudo apt-get update”
5. Install the Apache2 webserver using command “sudo apt-get install apache2”
6. In the Azure portal go to rg-websrvr-01 and select Vm-websrvr-US-01 from the list
7. In the VM blade, select networking.
8. On the inbound port rules tab, click add inbound port rule.
9. In the Add inbound security rule window, enter the below values and leave others fields default

|  |  |
| --- | --- |
| Setting | Value |
| Basic | Any |
| Source Port Ranges | \* |
| Destination | Any |
| Destination port ranges | 80 |
| Protocol | Any |
| Priority | 310 |
| Name | Port\_8080 |

This is to allow internet to speak to the VM

1. Copy the public IP address of the VM and paste it in browser. This will fetch the index page of the apache webserver (index.html)which confirms the VM and the webserver installation is successful
2. Now we need to replace the index.html with the contents to distinguish each VM.
3. In the Linux server, locate the index.html file using command “/var/www/html”
4. Remove the index.html using “sudo rm index.html”
5. Create a index.html file using cmd “sudo nano index.html” this will open a editor where we can type our code. Type the code as below and save

*<html>*

*<body>*

*<h1> This is ecommerce website (country) server (number)</h1>*

*</body>*

*</html>*

1. *In the place of country, replace US for VM1 and VM2 and India for VM3 and 4 and numbers as well*
2. Repeat steps 1 to 12 for vm2,3and4 to allow port 80 , install apache2 and replace index.html

**Task4: Implement application gateway in front of VM’s and test if working**

In this task we would be creating 2 application gate ways which is a load balancer and behind it would be the virtual machines. The application gateway splits the traffic to each VM’s so that they are not overloaded.

1. In the Azure portal search and select **application gateways**
2. Click **add** on the application gateways blade to create a new gateway
3. Fill out the basics tab with the values given

|  |  |
| --- | --- |
| Setting | Value |
| Subscription | Provide your subscription |
| Resource Group | rg-websrvr-01 |
| Application gateway name | Lb-websrvr-US-01 |
| Region | East US 2 |
| Tier | Standard V2 |
| Enable Auto scaling | No |
| Virtual Network | Rg-websrvr-01-vnet |
| Subnet | Gtw-subnet-US |

1. Click Frontends tab and fill with the values given

|  |  |
| --- | --- |
| Setting | Value |
| Public ip address | Pip-gtw-us |

1. Click Backends and click add a backend pool
2. In the add a backend pool window enter the below values

|  |  |
| --- | --- |
| Setting | Value |
| Name | Gtw-websrvr-US-be |
| Backend targets | Add the 2 virtual machines |

1. Click Configuration and click add a routing rule and fill in the values in Listener tab

|  |  |
| --- | --- |
| Setting | Value |
| Rule Name | Routingrule-gtw-US |
| Listener Name | Listener-http-US |
| Frontend IP | Public |
| Protocol | http |
| Port | 80 |

1. Click the backend targets and fill the values below

|  |  |
| --- | --- |
| Setting | Values |
| Target type | Backend pool |
| Backend Target | Gtw-websrvr-US-be |
| Http settings | Click add new |

1. In add a HTTP setting fill in the values below and leave other fields as defaults and click save. Once done click Add. Click Review + create

|  |  |
| --- | --- |
| Setting | Value |
| HTTP settings name | HTTP-setting-US |

Creating 2nd application gateway for Indian servers

1. In the Azure portal search and select **application gateways**
2. Click **add** on the application gateways blade to create a new gateway
3. Fill out the basics tab with the values given

|  |  |
| --- | --- |
| Setting | Value |
| Subscription | Provide your subscription |
| Resource Group | rg-websrvr-02 |
| Application gateway name | Lb-websrvr-IN-01 |
| Region | Central India |
| Tier | Standard V2 |
| Enable Auto scaling | No |
| Virtual Network | Rg-websrvr-02-vnet |
| Subnet | Gtw-subnet-IN |

1. Click Frontends tab and fill with the values given

|  |  |
| --- | --- |
| Setting | Value |
| Public ip address | Pip-gtw-in |

1. Click Backends and click add a backend pool
2. In the add a backend pool window enter the below values

|  |  |
| --- | --- |
| Setting | Value |
| Name | Gtw-websrvr-IN-be |
| Backend targets | Add the 2 virtual machines |

1. Click Configuration and click add a routing rule and fill in the values in Listener tab

|  |  |
| --- | --- |
| Setting | Value |
| Rule Name | Routingrule-gtw-IN |
| Listener Name | Listener-http-IN |
| Frontend IP | Public |
| Protocol | http |
| Port | 80 |

1. Click the backend targets and fill the values below

|  |  |
| --- | --- |
| Setting | Values |
| Target type | Backend pool |
| Backend Target | Gtw-websrvr-IN-be |
| Http settings | Click add new |

1. In add a HTTP setting fill in the values below and leave other fields as defaults and click save. Once done click Add. Click Review + create

|  |  |
| --- | --- |
| Setting | Value |
| HTTP settings name | HTTP-setting-IN |

Adding DNS name to the Public IP created for application gateways

1. In the Azure portal search and select Public IP addresses
2. In the list of ip address select **pip-gtw-in**
3. In the pip-gtw-in blade, click configuration section and fill in DNS label name as **pip-gtw-in-dns**
4. Click save
5. Repeat the steps 1 to 4 for **pip-gtw-us** and give DNS label name as **pip-gtw-us-dns**

**Task5: Implement Traffic manager to distribute traffic to load balancers based on Geography**

In this task we would be deploying traffic manager to distribute traffic to the application gateways.In addition this follows geography based routing where Indian users will mostly accessing Indian VM’s and US users and users from other parts of the world will be accessing US virtual machines.

1. In the Azure portal search and select **Traffic Manager profiles**
2. Click **add** and put in the values below and click Create

|  |  |
| --- | --- |
| **Setting** | **Value** |
| Name | TM-websrvr-01 |
| Routing method | Geographic |
| Subscription | Your own subscription |
| Resource Group | Rg-websrvr-01 |

1. In the Traffic manager profiles, click TM-websrvr-01
2. In TM-websrvr-01 blade, click endpoints and click add
3. In the add endpoint window, fill the below values and click add

|  |  |
| --- | --- |
| Setting | Value |
| Type | Azure endpoint |
| Name | endpnt-tm-websrvr-in |
| Target resource type | Public ip address |
| Public ip address | Pip-gtw-in |
| Regional grouping | Asia |
| Country/region | India |

1. In the add endpoint window, fill the below values and click add

|  |  |
| --- | --- |
| Setting | Value |
| Type | Azure endpoint |
| Name | endpnt-tm-websrvr-us |
| Target resource type | Public ip address |
| Public ip address | Pip-gtw-us |
| Regional grouping | All world |

Click overview and copy the DNS name (<http://tm-websrvr-01.trafficmanager.net/>) and try it in a browser and you can see the US/Indian web page is getting displayed

**Task6: Create DNS zone and the domain name .Also connect the traffic manager endpoint to complete the final step**

We need to have a domain name to link our azure services. If you do not have a domain name , one can be created using **www.freenom.com**

1. In the Azure portal search and select **DNS zones**
2. Click add and provide the values as below

|  |  |
| --- | --- |
| Setting | value |
| Subscription | Your sub |
| Resource group | Rg-websrvr-01 |
| Name | myfrstazureprjt.tk (domain name from freenom) |

1. Go to myfrstazureprjt.tk resource and click the overview blade and click record set
2. In the Add record set window fill the values as below and click save

|  |  |
| --- | --- |
| Setting | Value |
| Name | www |
| Type | CNAME |
| Alias record set | Yes |
| Alias Type | Azure Resource |

1. In the DNS overview, Get the DNS name servers and replace them in your domain present in freenom website.
2. Wait for few mins (it might take even 24 hrs to sync the name servers)and type the domain name in the browser to access our website we created.