Contents

[LOAD BALANCER 2](#_Toc130210427)

[LOAD BALANCER CONCEPT 2](#_Toc130210428)

[COMPONENTS OF LOAD BALANCER 2](#_Toc130210429)

[TYPES OF LOAD BALANCER 2](#_Toc130210430)

[LOAD BALANCER SKUs 3](#_Toc130210431)

[BASIC LOAD BALANCER 3](#_Toc130210432)

[STANDARD LOAD BALANCER 8](#_Toc130210433)

[AZURE TRAFFIC MANAGER 8](#_Toc130210434)

[GEOGRAPIC ROUTING METHOD 8](#_Toc130210435)

[AZURE TRAFFIC MANAGER SET UP 9](#_Toc130210436)

[AZURE TRAFFIC MANAGER ROUTING METHODS 9](#_Toc130210437)

[PRIORITY ROUTING METHOD 9](#_Toc130210438)

[PERFORMANCE ROUTING METHOD 12](#_Toc130210439)

[SUBNET ROUTING METHOD 12](#_Toc130210440)

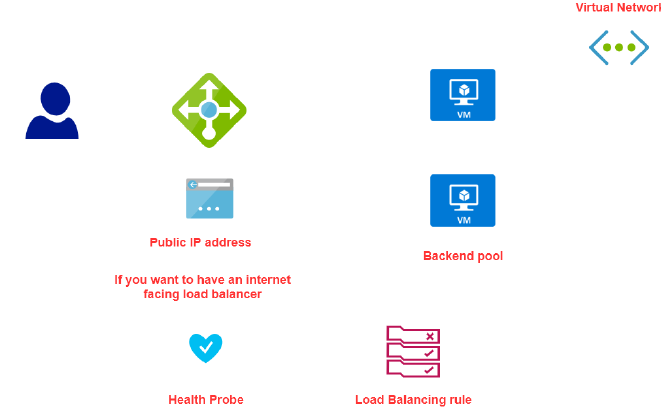
[MULTIVALUED ROUTING METHOD 13](#_Toc130210441)

[WEIGHTED ROUTING METHOD 13](#_Toc130210442)

## LOAD BALANCER

|  |  |
| --- | --- |
|  | * The Load Balancer Service, distribute the user requests across the pool of resources (e.g., Azure virtual machines) to maintain availability. * There are many distributions-based software available, the basic one is Azure Load Balancer. * *The azure load balancer works on Layer 4(Transport Layer) and Application Gateway (Application Layer) of the OSO Model* * The Azure load balancer comes in 2 pricing model (SKU)   + BASIC LOAD BALANCER   + STANDARD LOAD BALANCER |

### LOAD BALANCER CONCEPT



### COMPONENTS OF LOAD BALANCER

|  |  |
| --- | --- |
| BACKEND POOL | When we create a load balancer, we specific something called backend pool, which is a set of VMs to which the LB will route the traffic to. |
| FRONTEND IP | * The LB has a public IP address (called Frontend IP address). * The user can be able to access the VM via public IP of the load balancer * The redirect of request from LB to VM are done via private IP of VM |
| HEALTH PROBE | This helps the LB monitor the health of VMs in the backend pool |
| LOAD BALANCING RULES | These rules drive how request can be distributed across the VM (which are part of backend pool) |

### TYPES OF LOAD BALANCER

There are 2 types of load balancer

1. INTERNAL LOAD BALANCER
2. PUBLIC LOAD BALANCER

#### INTERNAL LOAD BALANCER

|  |  |
| --- | --- |
|  | * The internal load is used to direct traffic only between Azure’s internal resources i.e., the resources managed by Azure infrastructure or resources connected to Azure infrastructure using secure VPN * Internal load balancer can be used when want to divide the traffic to coming from other Azure Resources * It can also use internal load balancer or the traffic coming from on-premises network that is connected to an Azure resource via s secure VPN connection. |

##### SETTING UP INTERNAL LOAD BALANCER

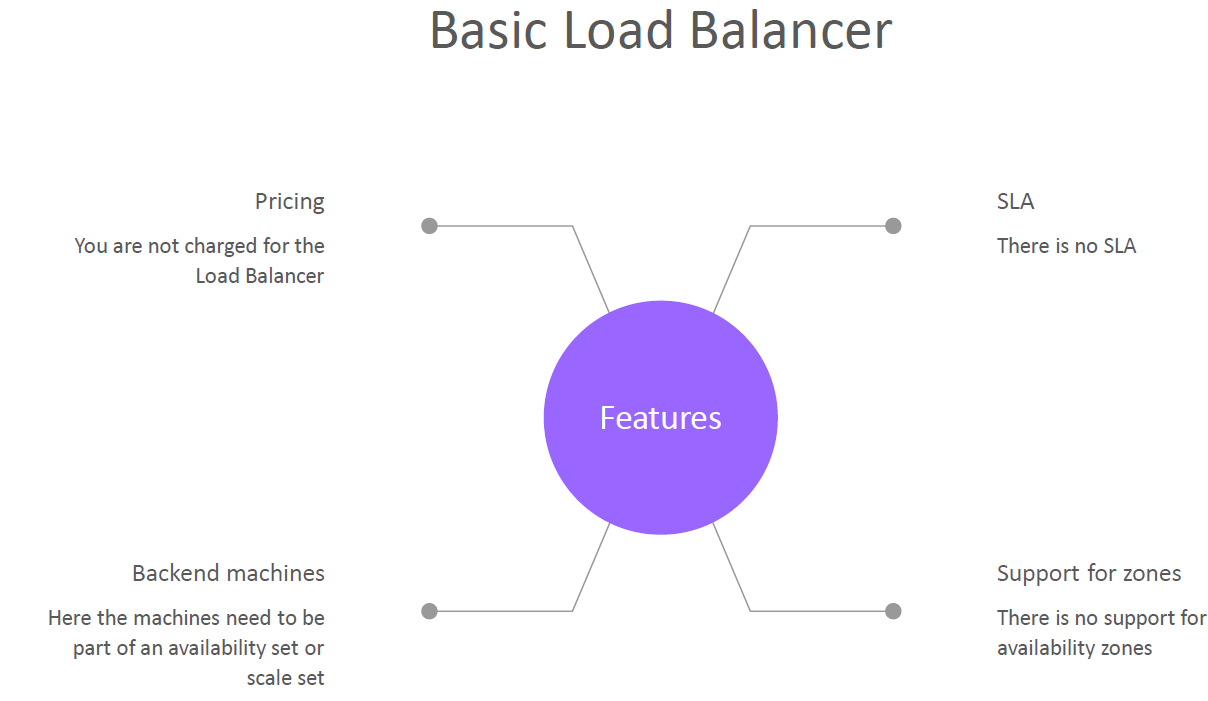
#### PUBLIC LOAD BALANCER

* Public load balancer is used to handle the traffic between public facing IP address of incoming traffic to private IP address of Azure resources.

### LOAD BALANCER SKUs

|  |  |
| --- | --- |
| **BASIC LOAD BALANCER** | **STANDARD LOAD BALANCER** |
| *Free* | *Charges on Per hour basis* |
| *The VM in the backend pool must be part of availability set or scale set* | *VM can be an independent machine that are part of a VNET* |
| *Health Probes support TCP and HTTP only* | *Health Probes support TCP and HTTP and HTTPS* |
| *No support for availability zone* | *Support for availability zone* |
| *NO SLA* | *SLA of 99.99%* |

### BASIC LOAD BALANCER



#### SETTING UP BASIC LOAD BALANCER (VM IN AVALIBILITY SET)

To illustrate the basic load balancer services set up. Let’s follow the following steps

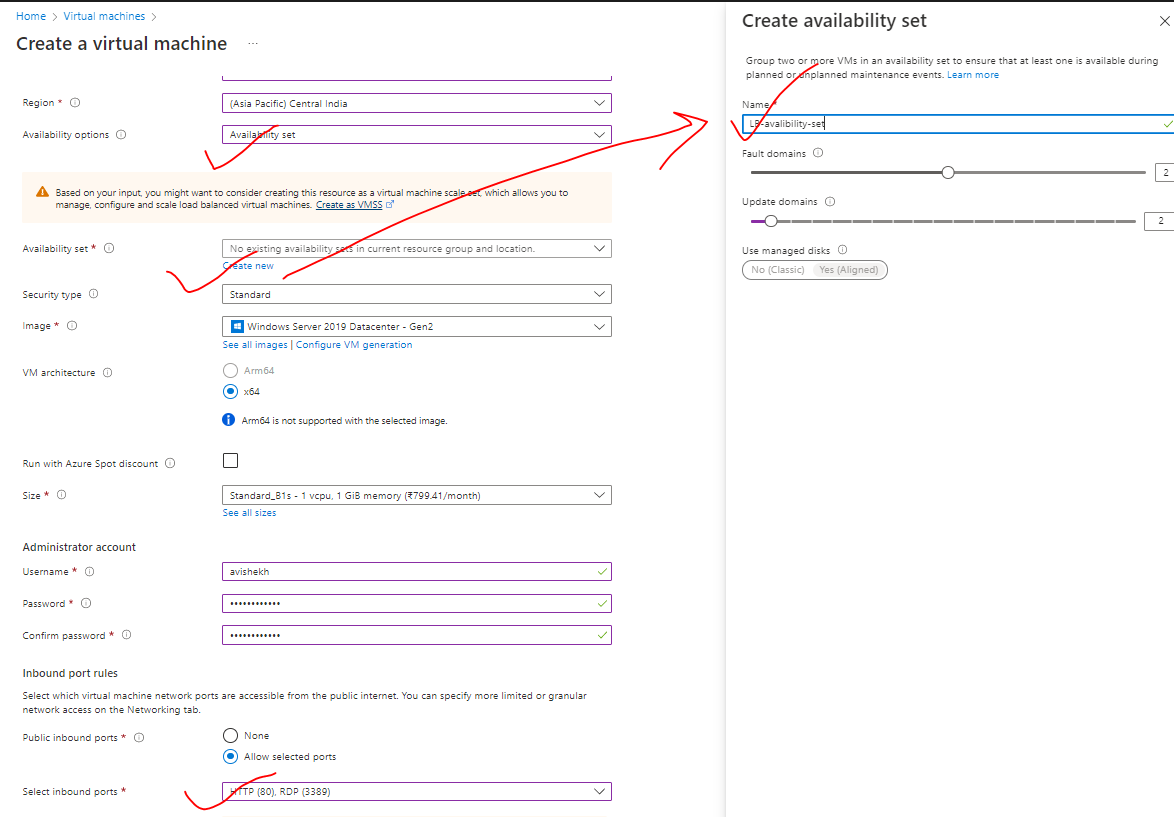
* ***STEP 1*:** *We will create VMs, which will be a part of an* ***Availability Set (Make sure the VMs are part of same VNET)***
* ***STEP 2:*** *Install IIS with a Default.html page*
* ***STEP 3:*** *Create a Public IP address (Note public IP address is a resource in Azure). The Public IP will be assigned to the Load Balancer. Also known as Front End public IP address*
* ***STEP 4:*** *Create and configure a Load Balancer resource. As part of configuration -we need to set up a Backend pool of the VMs created in Step 1*
* ***STEP 5:*** *Configure the health probe to check the health of VM in the backend pool*
* ***STEP 6:***  *Create Load balancing Rules – which will give the routing of request to the VM in backend pool*
* ***STEP7:*** *Now we can be able send the request to load balancer via its public IP address, which will then redirect the traffic to VM in the backend pool.*

#### SETTING UP BASIC LOAD BALANCER (VM IN SCALE SET)

STEP 1: *We will create VMs, which will be a part of an* ***Availability Set***

**In this step**

1. **Port 80 has been enabled**
2. **ISS has been installed on the VM**
3. **After validation of IIS installation – Disassociate the Public IP address of VM**

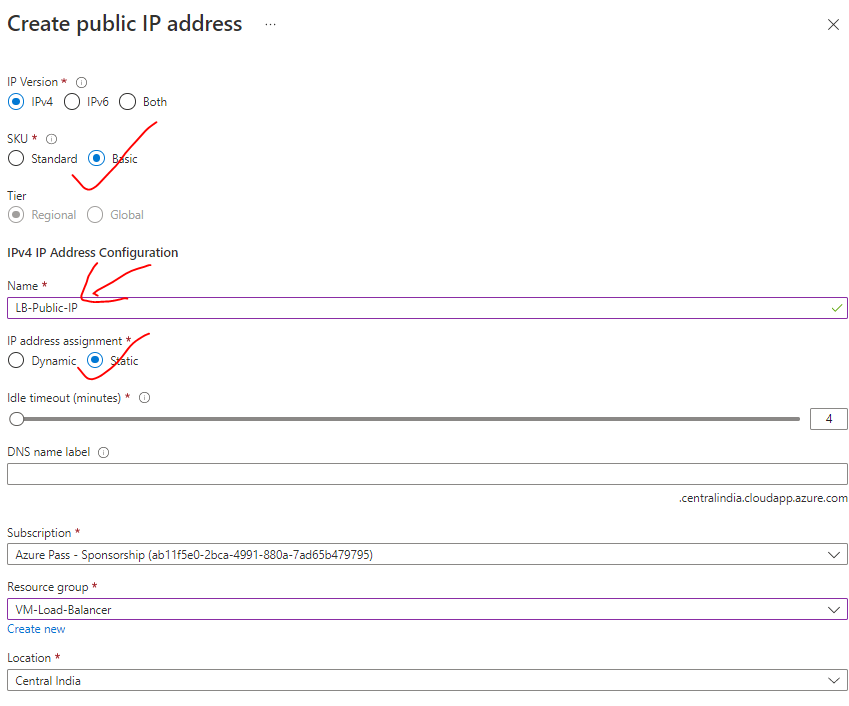


|  |  |
| --- | --- |
|  | * The communication between the Load balancer and the VMs in the backend pool – happens using private IP address * For now – let’s keep the public IP address so that we can deploy IIS and validate it * After that we can disassociate the public IP address from the VM as the communication between LB and VM are through private IPs |

**DISASSOCIATE PUBLIC IP FROM VM**

|  |  |
| --- | --- |
|  | 1. Go to VM 🡪 Networking 🡪Network Interface 🡪 IP Configuration 2. Disassociate the Public IP and make the provide IP as a static Ip 🡪 Save |

***STEP 3:*** *Create a Public IP address (Note public IP address is a separate resource in Azure). The Public IP will be assigned to the Load Balancer. Also known as Front End public IP address*



***STEP 4:*** *Create and configure a Basic Load Balancer resource. As part of configuration -we need to set up a Backend pool of the VMs created in Step 1*

|  |  |
| --- | --- |
| BASIC | FRONT END CONFIGURATION |
|  | * Create the Load balancer * Note – Backend pool configuration can be done while configuring the LB   **TIER CONFIGURATION**   * REGIONAL – VMs is a region will be load balanced * **GLOBAL –** VMs across multiple regions can be load balanced. |

##### LOAD BALANCER CONFIGURATIONS

CONFIGURING BACKEND POOL

|  |  |
| --- | --- |
|  | Add the VMs in the backend pool |

CONFIGURING HEALTH PROBE

|  |  |
| --- | --- |
|  | * After the interval of every 5 second – Load balancer will initiate a TCP handshake. if the VM acknowledge the TCP request – then the LB will know that VM is healthy |

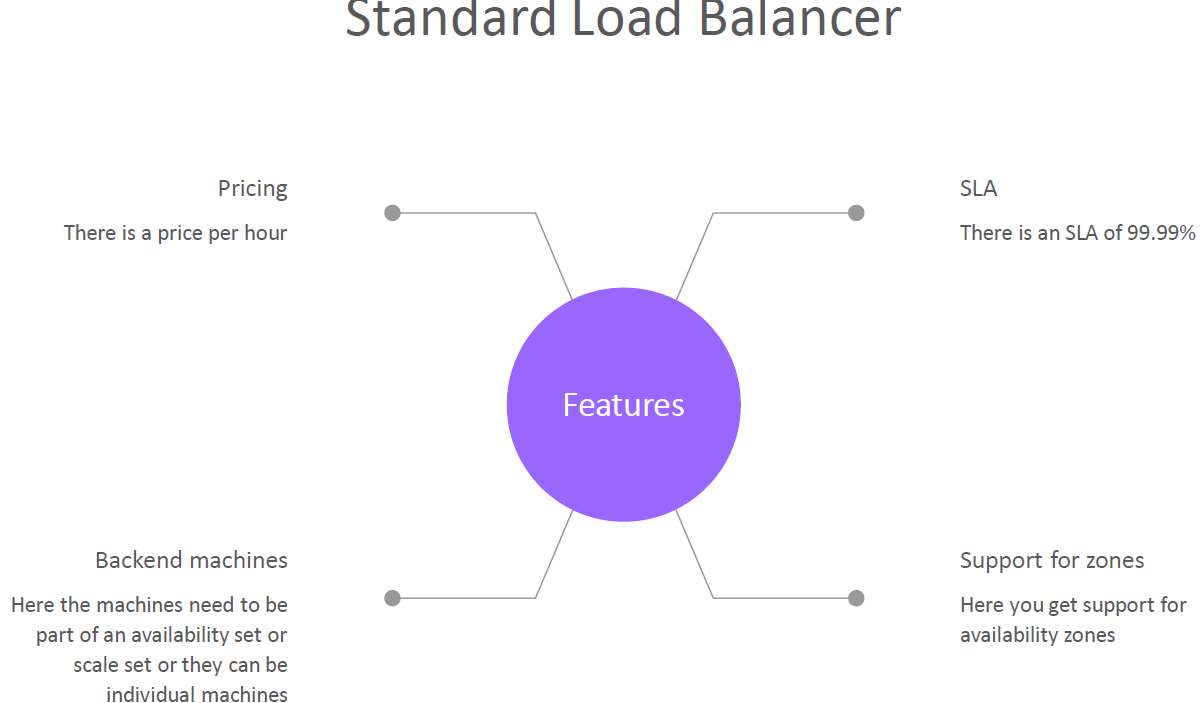
CONFIGURING LOAD BALACING RULES

|  |  |
| --- | --- |
|  | 1. PORT- 80 2. BACKEND PORT – It’s the Port in the VM which is the PORT of the IIS  * This means – If the request comes to port number 80 of the load balancer (normal HTTP web request default port is 80) – then it will be distributing the traffic to backend pool port. |

#### LOAD BALANCER NAT RULES

* For the load balancer, we have our backend virtual machines, which are part availability set.
* Since, these VMs don’t have public IP addresses, we can’t be able to connect these VM to internet or RDP onto these machines from our workstations
* For such use cases, Load balancer provide option to connect using its own public IP address - onto a port or use a particular service on the virtual machine- this is done with the help of ***NAT (Network Address Translation)***
* Using the virtual machine scale self-service - we were able to spin up multiple machines based on certain conditions. For example - let’s say the CPU percentage of the machines and the skill set was going beyond a particular threshold, then we would increase the number of machines that were part of the skill set.
* Scale Set give a capability on the platform to scale
* Azure Load Balancer service ensures that all the user requests are evenly distributed across these machines so that the load is uniformly dispersed across the machines that are part of your scale set.
* This service is used to distribute the incoming network traffic across a group of backend resources of servers
* You can define two types of load balancers –***Public or Private Load Balancers***
* You have 2 SKUs for the Load Balancer –**Standard and Basic Load Balancer**

### STANDARD LOAD BALANCER



## AZURE TRAFFIC MANAGER

* **Traffic Manager a DNS based traffic load balancer.**
* The Azure load balancer and Application gateways are regional (i.e.it can manage the resources in a given region) , but a traffic manager is a global service. Hence when we deploy a traffic manager profile- we don’t to specify a region for the deployment of the resource.
* The traffic manager allows us to direct the client requests to an appropriate service endpoint (end point can be VMs , Web App or application hosted on-premise datascenter) based on routing method. The endpoint needs to be public endpoint that can be hosted inside or outside Azure
* The Traffic manager monitors the health monitoring of the endpoints.

### GEOGRAPIC ROUTING METHOD

1. ***In Geographic routing method the end point depending upon the location the DNS query originates from (it depends upon user’s location).***
2. This is different from the performance routing method. In the performance routing method, the traffic manager looks for a resource which takes less time in terms of latency. That is the endpoint that will be returned to the user.

#### SETTING GEOGRAPIC ROUTING METHOD

Step 1: Create 2 VMs in two different Region

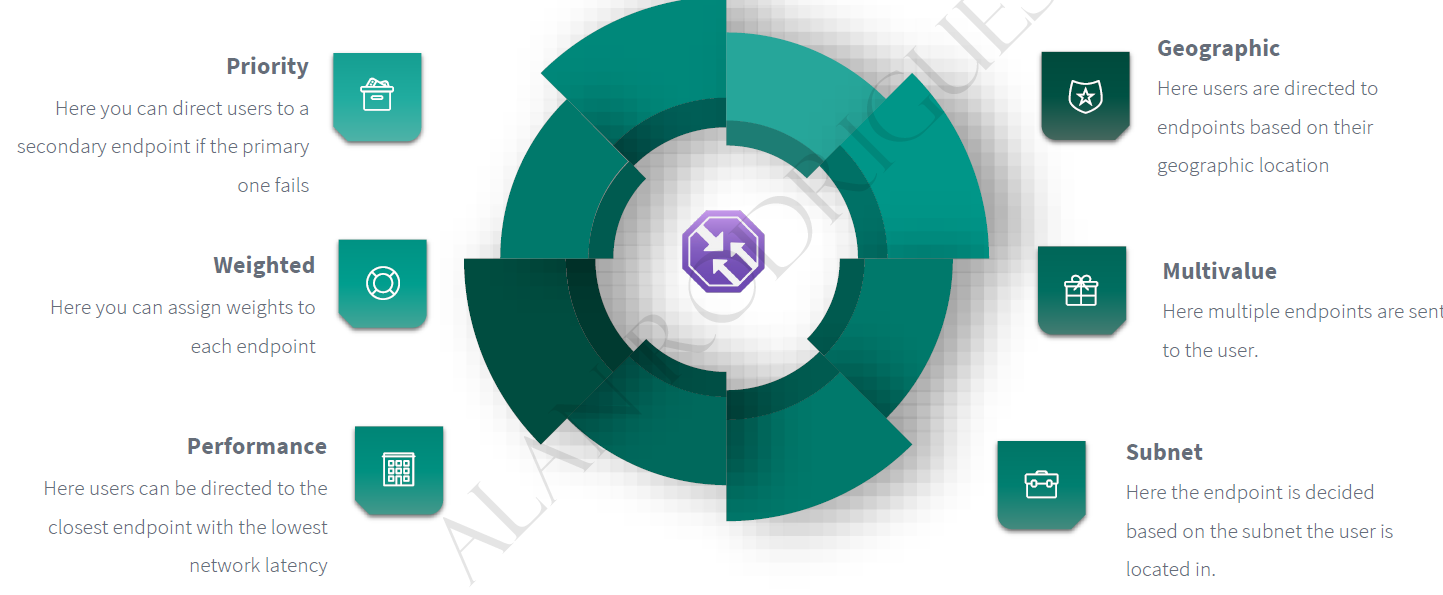
Step 2: Configure the traffic manager endpoint 🡪 Select the region which will serve the request which is coming from the same region

|  |  |
| --- | --- |
|  |  |

### AZURE TRAFFIC MANAGER SET UP

### AZURE TRAFFIC MANAGER ROUTING METHODS

Azure Traffic Manager supports **six traffic-routing methods to determine how to route network traffic to the various service endpoints.** For any profile, Traffic Manager applies the traffic-routing method associated to it to each DNS query it receives. The traffic-routing method determines which endpoint is returned in the DNS response.



### PRIORITY ROUTING METHOD

|  |  |
| --- | --- |
|  | * This routing method help in direct traffic to secondary endpoint of primary endpoint is not available. |

#### SETTING UP PRIORITY ROUTING METHOD

STEP 1*: SET UP VMs IN DIFFRENT REGION (it can be in same region too. Since TM is a global service, it can manage resources in any region)*

STEP 2:

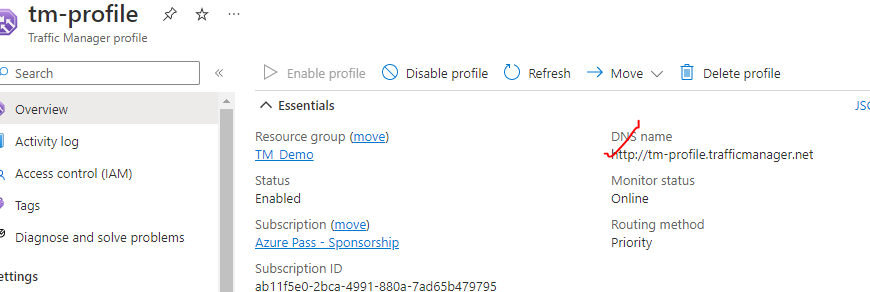
* + INSTALL IIS on both the VM – These are the VM where the traffic will be directed
  + Add a DNS names to the VMs – as traffic manager does DNS based routing.

|  |  |
| --- | --- |
|  | STEP 3: Search for **Traffic Manager Profile 🡪 Create**  *As it is a global service, hence we don’t need to mention the region of the Traffic Manager Profile (unlike Load Balancer and App Gateway).* |
|  | * Add a DNS names to the VMs * Go to VM Overview 🡪 Click on Public IP address🡪 DNS name label |

STEP 4: Add the VMs to the traffic Manager endpoint

|  |  |  |
| --- | --- | --- |
|  | **The priority will define – which public IP(VM public IP) will be a Primary End point** | |
| * When the priority is higher means when users access the traffic manager end point, all of the users will be directed to this VM. Only when the primary VM goes down that user will be directed the secondary VM the IBM or the endpoint, this is like a fallback mechanism. * After adding the endpoint – make sure Monitor Status is “Online” | |

STEP 4: Now the VMs can be accessed via Traffic Manager DNS name.



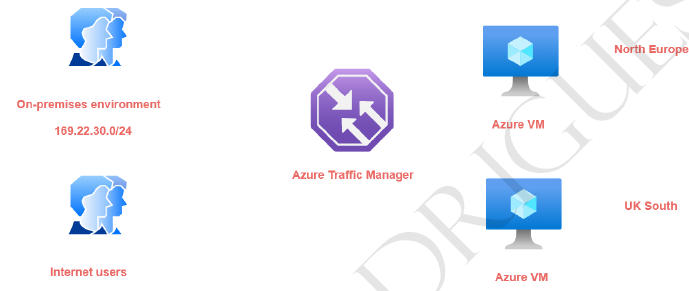
|  |  |
| --- | --- |
|  | * Note – In the above example – if we stop the primary VM – the request will be then route to secondary endpoint * PROBING INTERVAL (In sec.) – The time interval in which probing will happen * TOLERATED NUMBER OF FAILURES – After this TM will consider it as unhealthy resource * PROBE TIMEOUT - |
|  | |

|  |  |
| --- | --- |
|  | **TO UPDATE THE EXISTING ROUTING METHOD**   1. Delete the existing endpoints 2. Go to configuration of TM 🡪 Routing Method 🡪 Save 3. Add the endpoints again |

### PERFORMANCE ROUTING METHOD

1. In performance routing method – it routes the traffic to the location which is closest to use location

### SUBNET ROUTING METHOD



|  |  |
| --- | --- |
|  | 1. In this routing method – we map a specific set of end-user IP address range to specific endpoint. For example, if we want to direct the traffic from on premises environment- having a particular IP address range to an endpoint and all other internet users onto another endpoint. 2. The subnet routing methods - looks at the IP address range of the clients that are connecting to the traffic manager and then they be directed onto the end point. 3. If we want to create a default endpoint that is created- so that if the other end points don't   match the subnet range then the default end point will serve the request. **For default endpoint the subnet routing settings is kept empty.**   1. Hence – let’s say if the user hit the TM DNS and if the user’s IP does not fall in the range of Subnet routing setting – it will fallback to the default route |

### MULTIVALUED ROUTING METHOD

### WEIGHTED ROUTING METHOD

|  |  |
| --- | --- |
|  | * In weighted routing method we weight each endpoint in the traffic manager profile * Lets say if we 2 VM (endpoints)- which are weighted **50.**  The traffic to TM will be distibuted across endoint independently evenly. |

#### USE CASE OF WEIGHTED ROUTING METHOD

* So in our case, we have two AC vents in place
* Now we can assign leads on to the end points that are mapped on to our traffic margin profile.
* So, for example, I can't assign a bit of go on to the end point for the as the IBM in the North,
* their application.
* And then I can assign a weight of 50 here by the end point that is marked onto the UK South location.
* So this is like actually load balancing now request across these virtual machines.
* So when half of the users are connecting onto the air traffic manager profile, half of them will be
* directed onto the beam in the Noctua application and the other half will be directed onto the VM in
* the UK South location.
* So here the requests are actually distribute based on the bits.
* Now, one of the most useful scenarios of using the reedit routing method is when it comes onto something
* known as blue green deployments.
* So normally what happens is that initially all your users.
* So you could be having an endpoint with a bit of full 100 percent when an after users are being directed
* onto one end point.
* Now let's say you are deploying a newer version of the application on new end point, so it could be
* an A-Z IBM.
* It could be a ACR bebop.
* It could be another endpoint.
* So now you want us to form some of the users to be detected on to the newer version of your application.
* So you might maybe change the reteach from 80 percent here and 20 here.
* So 20 percent of the users are now directed onto the newer version of the application.
* Once you know you know that there are no issues with the newer version of your application, then you
* can slowly switch up.
* You know, you could make this 80 percent and you can make this 20 percent.
* And in the end, you could do a full cut over that, and this could be 100 percent and you can just
* go and disable this endpoint.
* So this is good for actually distributing the traffic across multiple endpoints.
* Again, this is very easy to achieve.
* Let's see how we can implement this.
* So again, first of all, I'll go onto my traffic manager profile.
* I'll go on to my end points and let's delete and points.
* And I'll go back to my profile.
* Let's change the configuration.
* So from the multi value onto the Beated.
* Let me click on Save.
* Go on to the profile.
* Let me go on to and points out and then point.
* You can put an end point one there, I can choose the cloud service, the public IP address, you can
* put traffic via one.
* And here in the weeds, you can put 50 and you can go out and click on Add.
* Take a look at the other end point.
* Give it a name.
* Yeah, public IP address to the public IP does a stop at the M2 and put the webpage has again 15.
* And click on Add.
* You have no unequal distribution when it comes onto the beach on two end points.
* And if I go ahead and just take the the innocent point.
* In a new type, if I go on to be followed, Estima, so I'm being directed onto the Not Europe location.
* So with the help of the state, you can actually go in and disrupt traffic across the end points when
* it comes as air traffic manager.