Azure Load Balancer

→ What is Azure Load Balancer?

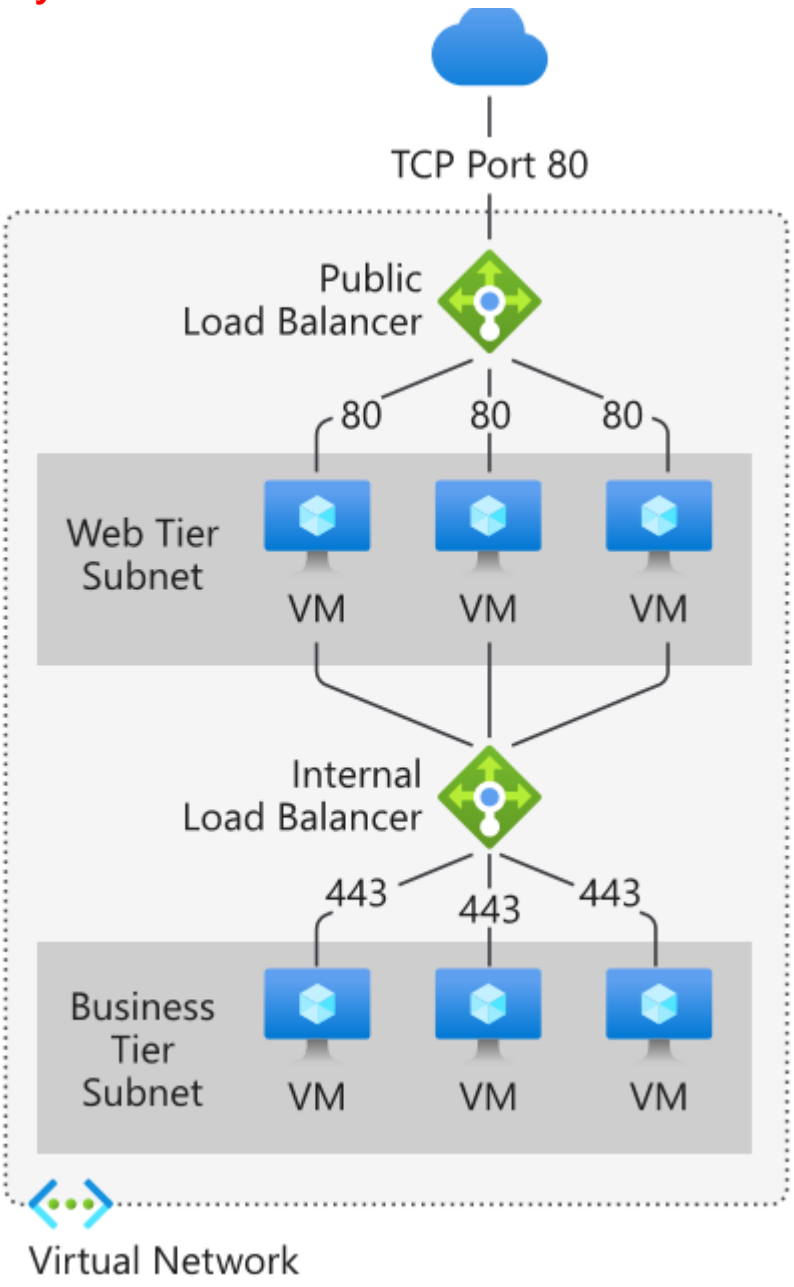
- ✓ *Load balancing* refers to evenly distributing load (incoming network traffic) across a group of backend resources or servers.

- ✓ Azure Load Balancer operates at layer four of the Open Systems Interconnection (OSI) model. It's the single point of contact for clients. Load Balancer distributes inbound flows that arrive at the load balancer's front end to backend pool instances. These flows are according to configured load balancing rules and health probes. The backend pool instances can be Azure Virtual Machines or instances in a virtual machine scale set.

- ✓ A **public load balancer** can provide outbound connections for virtual machines (VMs) inside your virtual network. These connections are accomplished by translating their private IP addresses to public IP addresses. Public Load Balancers are used to load balance internet traffic to your VMs.

- ✓ An **internal (or private) load balancer** is used where private IPs are needed at the frontend only. Internal load balancers are used to load balance traffic inside a virtual network. A load balancer frontend can be accessed from an on-premises network in a hybrid scenario.

→ **Why use Azure Load Balancer?**



→ Why use Azure Load Balancer?

- ✓ With Standard Load Balancer, you can scale your applications and create highly available services. Load balancer supports both inbound and outbound scenarios. Load balancer provides low latency and high throughput, and scales up to millions of flows for all TCP and UDP applications.

Key scenarios that you can accomplish using Standard Load Balancer include:

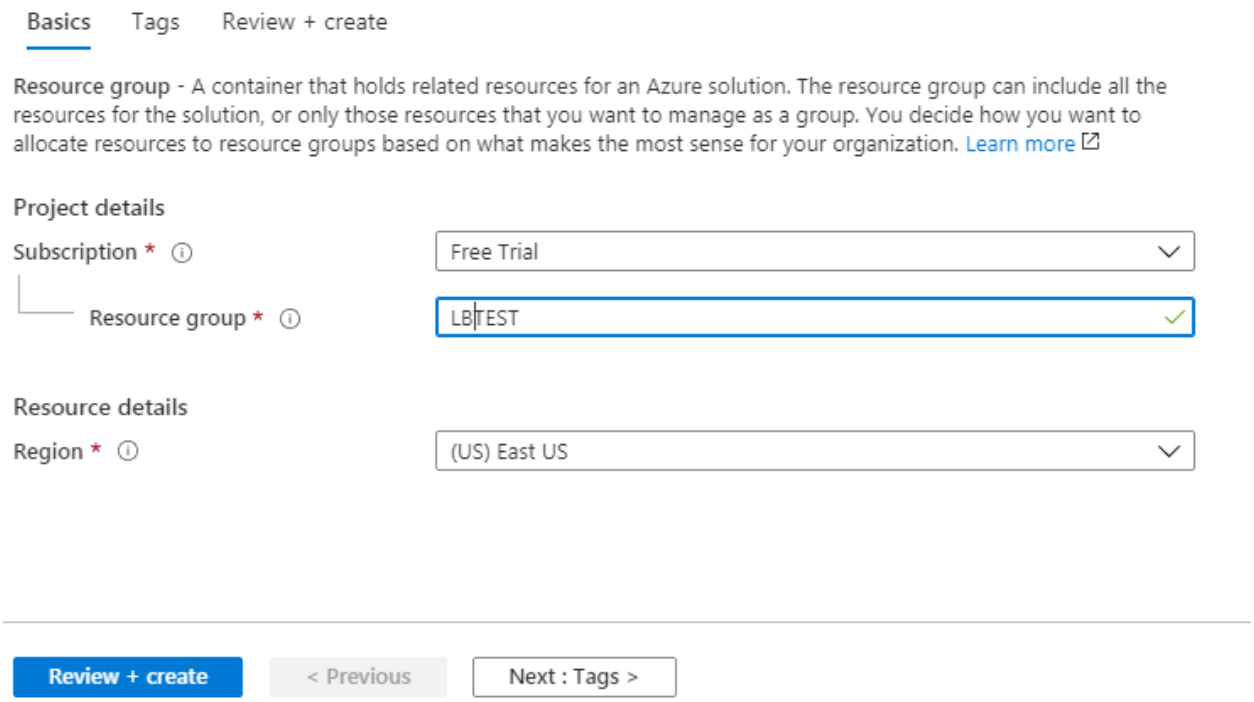
- ✓ Load balance **internal** and **external** traffic to Azure virtual machines.
- ✓ Increase availability by distributing resources **within** and **across** zones.
- ✓ Configure **outbound connectivity** for Azure virtual machines.
- ✓ Use **health probes** to monitor load-balanced resources.
- ✓ Employ **port forwarding** to access virtual machines in a virtual network by public IP address and port.
- ✓ Enable support for **load-balancing** of **IPv6**.
- ✓ Standard Load Balancer provides multi-dimensional metrics through Azure Monitor. These metrics can be filtered, grouped, and broken out for a given dimension. They provide current and historic insights into performance and health of your service. Resource Health is also supported. Review **Standard Load Balancer Diagnostics** for more details.
- ✓ Load balance services on **multiple ports, multiple IP addresses, or both**.
- ✓ Move **internal** and **external** load balancer resources across Azure regions.
- ✓ Load balance TCP and UDP flow on all ports simultaneously using **HA ports**.

Let's start work with Load Balancer

1) Login into Azure Portal by using your credentials.

2) Resource Group creation: -

A) Create a Resource Group with the name LBTEST and Location East US. As shown in the below picture.

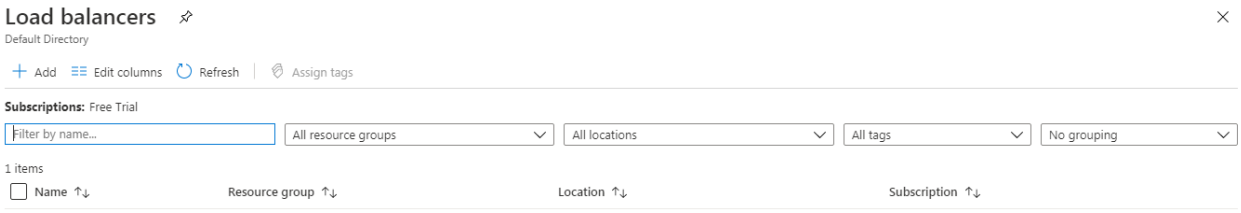


B) Complete the Resource Group creation by clicking on Review +create and the create buttons.

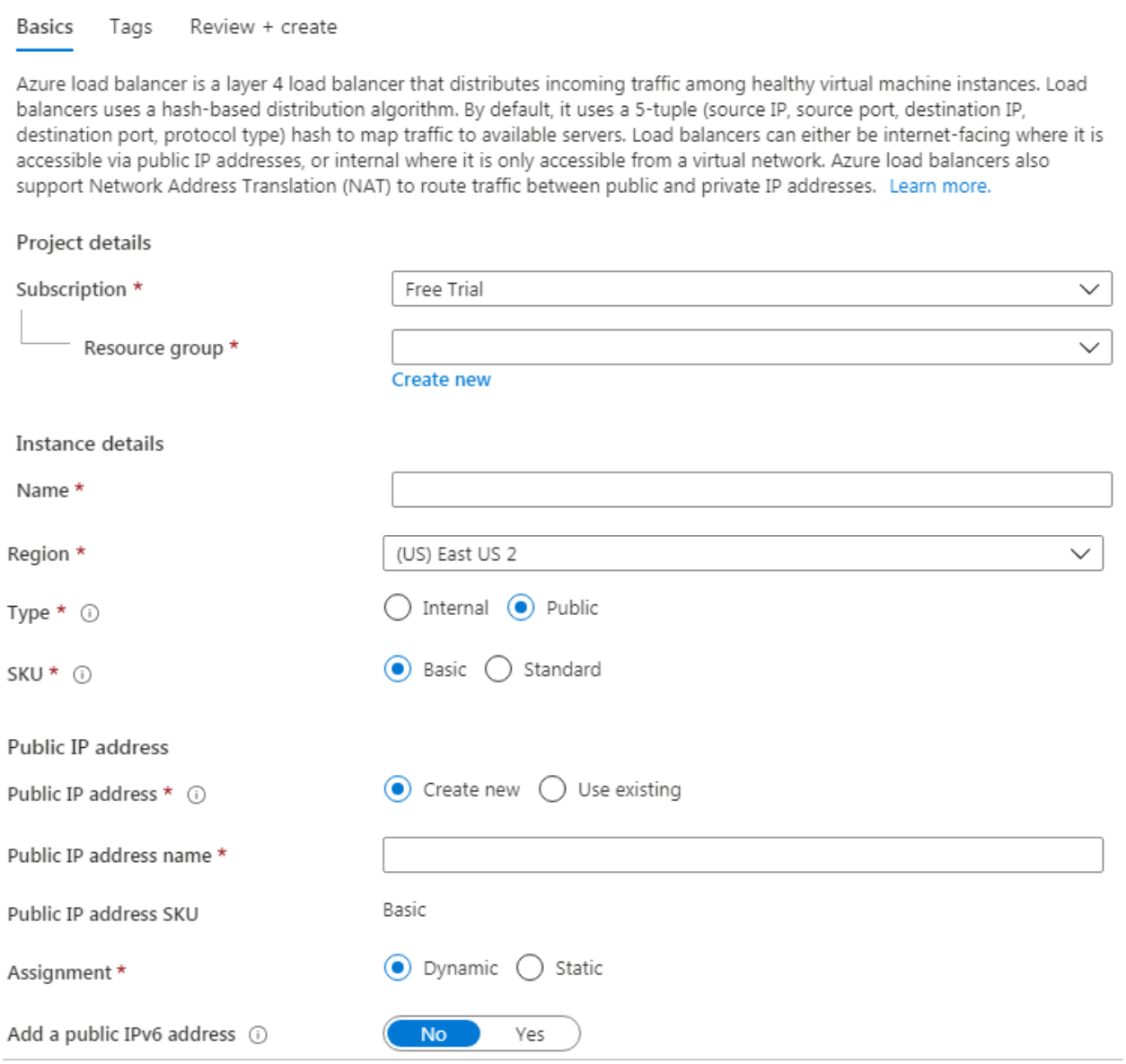
Now we are going to create the Load Balancer

3) Load Balancer: -

A) In the Dashboard search box, search with the Load Balancers.



B) Now Click on the Add Button it will navigate to Load balance creation page.



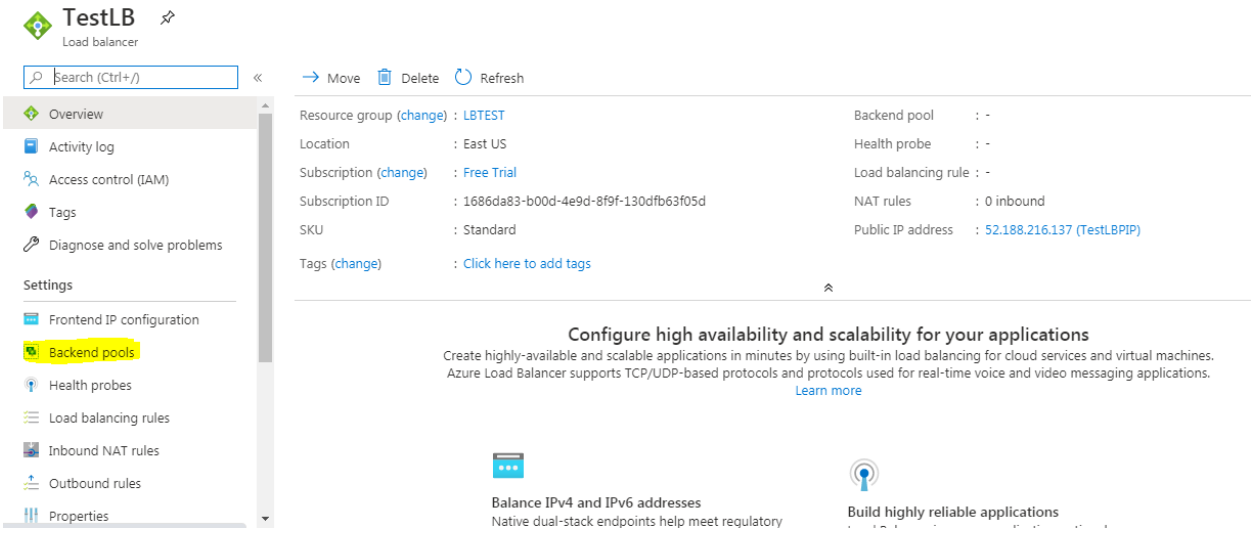
C) Fill the following details in the above fields.

- Name: TestLB
- Region: East Us
- Type: Public
- SKU: Standard
- Public IP Address: Create New
- Public Ip address name: TestLBPIP
- Availability zone: Zone-redundant
- Add a public IPv6 address: Leave as it is

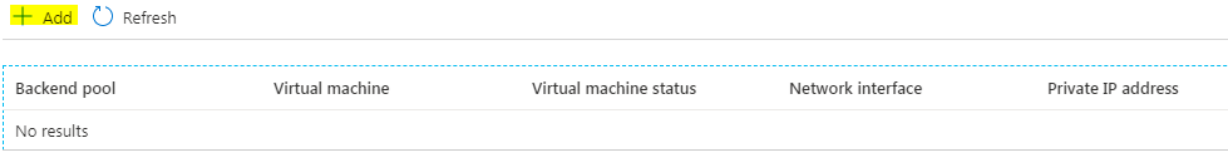
D)Then click on Review + create and then create.

E) Your Load Balancer is created.

F) After creating the Load Balancer go to the Overview of TestLB Load balancer.

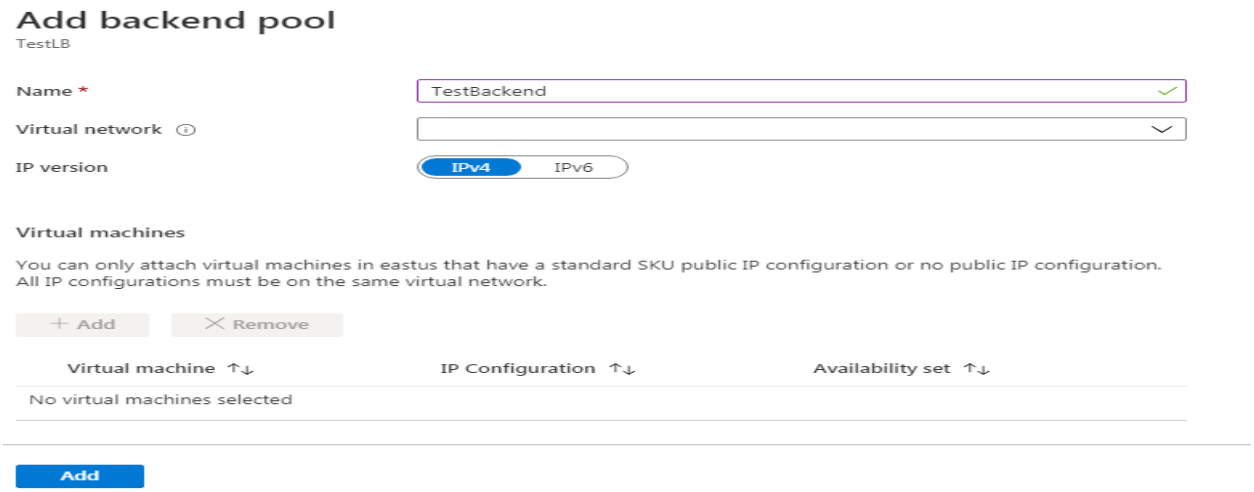


G) Now click on the Backend pools in the Settings. It will navigate to another page.



H) In this page Click on Add button to add the backend pool (VMS).

Note: Still we are not created any VMS.

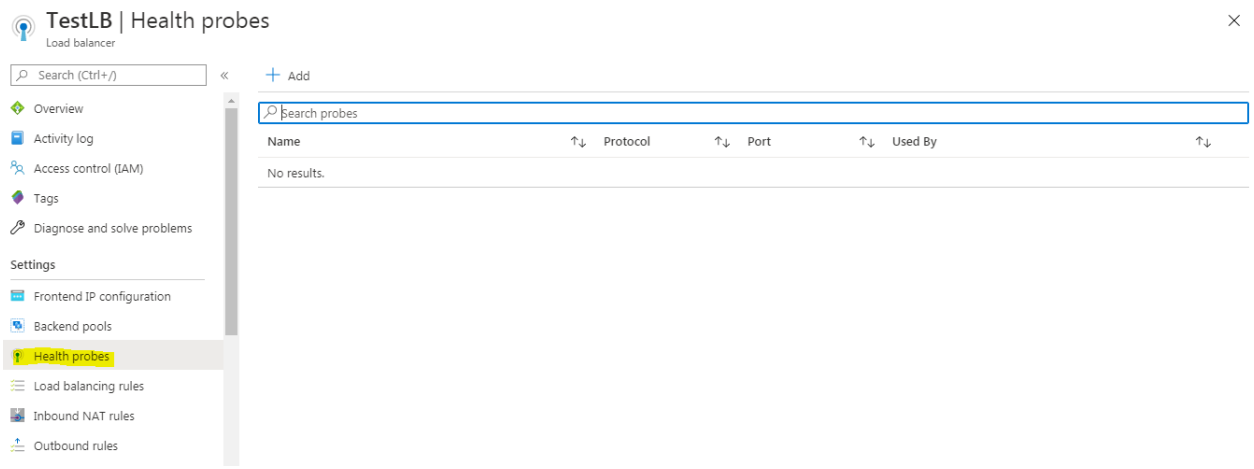


I)Now Add the following details in the Add Backend pool page

- Name : TestBackend
- Virtual Network: Leave it empty
- IPVersion: Leave it as it is

J) Now click on Add button

K) Now go to the overview of TestLB and click on Healthprob under the settings.

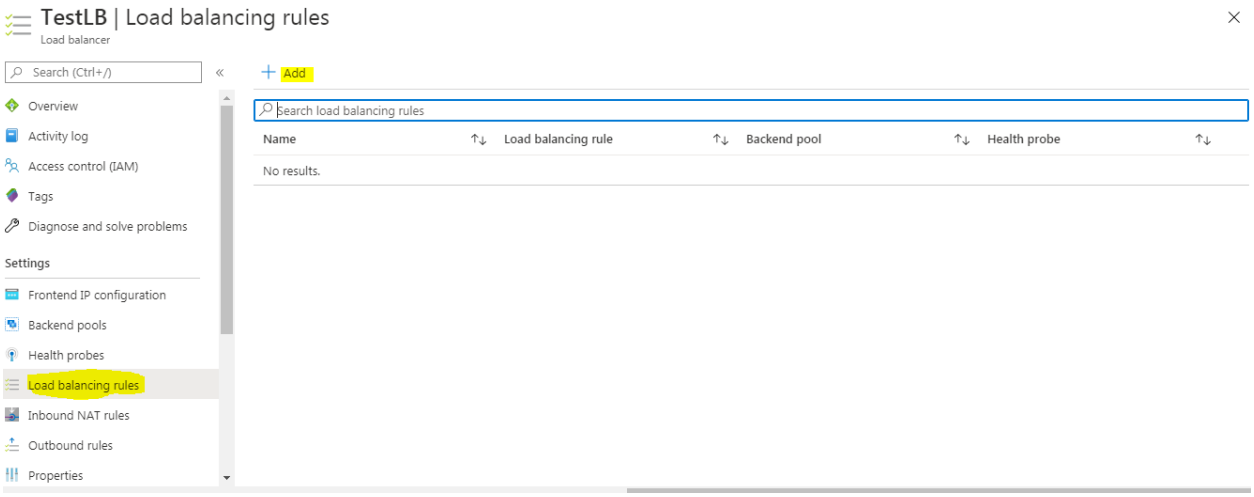


J) Click on Add button to add **Health probes**.

K) Enter the following details in Add **Health probe page**

- Name: TestHealthprobe
- Protocol: Select HTTP
- Interval: Set as 15 secs
- Leave remaining fields as it is
- Click on **OK** Button

L) Now go to the TestLB overview and click on the Load balancing rules under the settings and then click on the Add button to add the Load balancing rules.



M) In the Load balancing rule page just add the name as **TestLBRule** and **leave the remaining fields as it is** and click on **OK** Button.

4) Create the Virtual Network

Name: LBVN

IPAddress: 10.5.0.0/16

Subnet:10.5.1.0/24

5) Virtual Machines

A) Now we need to create the two VMS. For this search the Virtual Machines in the search box in the dashboard and click on Add button to create the virtual machines.

Virtual Machine 1:

1. Basic:

   Resource Group: TestLB1

   Location: East US

   Virtual Machine Name: TestVM1

   Availability options: Availability zone

   Availability zone: 1

   Image: Windows server 2019 Datacenter

   Size: B2ms

   User Name: Azureuser

   Password: Azureuser@123

Public inbound rules: Allow selected ports

Select inbounds port: RDP

2.Disks:

No changes

3.Network Interface

Virtual Network: Don’t change any thing

Subnet: Don’t change any thing

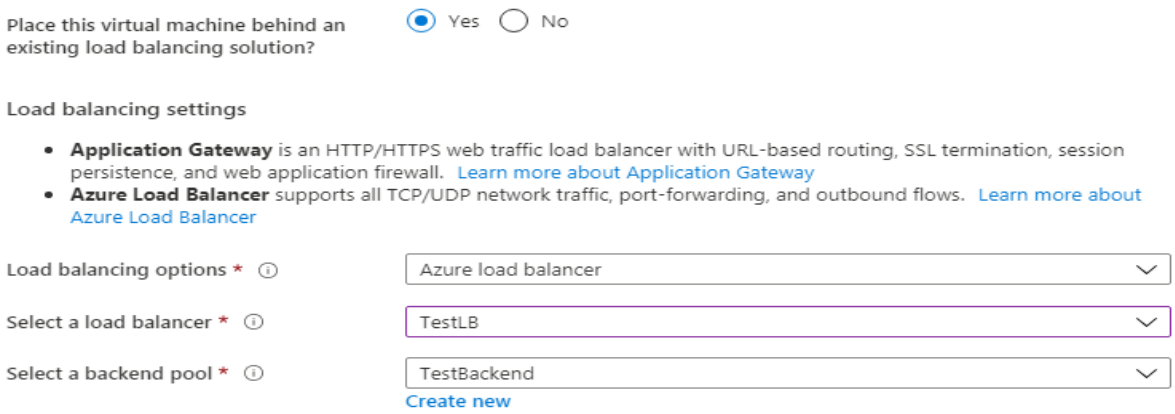
Public Ip: Click **Create New** and set as availability zone as **Zone-redundant** and click on **OK**.

NIC network security group: Advanced

Select Inbound ports: RDP

Accelerate Network: Off

Load Balancing: Yes



Load Balancing options: Azure load balancer

Select a load balancer: TestLB

Select a backend pool: TestBackend

4)Management

Turn off boot diagnostics and keep remaining fields as it is.

5) Advanced

Click on **Review + Create** and then click **Create**.

Virtual Machine 2:

2. Basic:

Resource Group: TestLB

Location: East US

Virtual Machine Name: TestVM2

Availability options: Availability zone

Availability zone: 2

Image: Windows server 2019 Datacenter

Size: B2ms

User Name: Azureuser

Password: Azureuser@123

Public inbound rules: Allow selected ports

Select inbounds port: RDP

2.Disks:

No changes

3.Network Interface

Virtual Network: Don’t change any thing

Subnet: Don’t change any thing

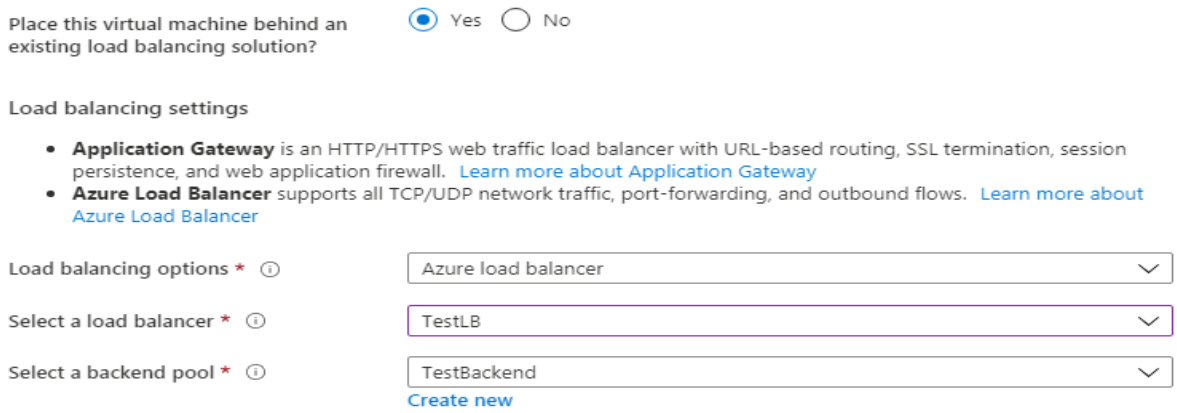
Public Ip: TestVM2-ip

NIC network security group: Advanced

Select Inbound ports: RDP

Accelerate Network: Off

Load Balancing: Yes



Load Balancing options: Azure load balancer

Select a load balancer: TestLB

Select a backend pool: TestBackend

4)Management

Turn off boot diagnostics and keep remaining fields as it is.

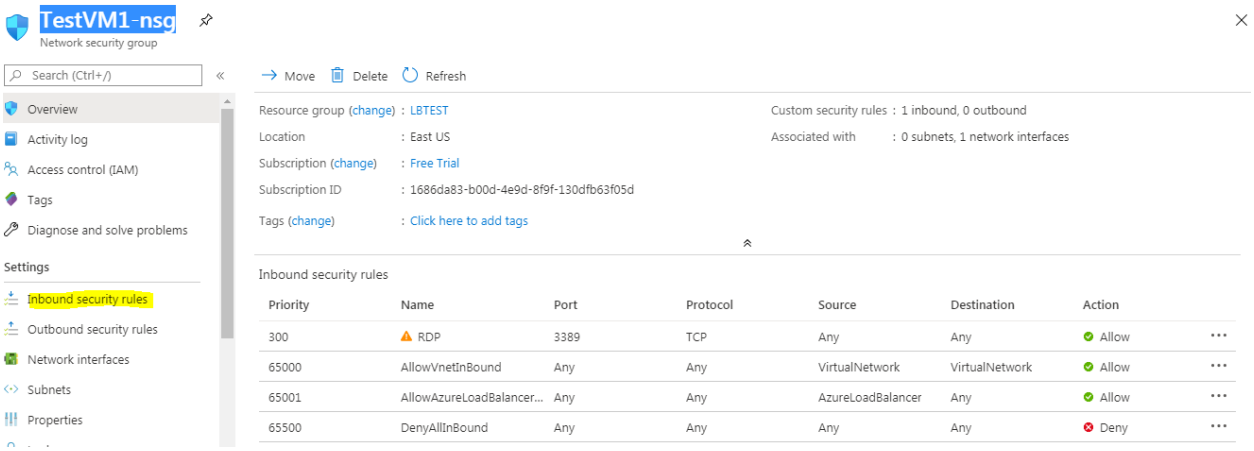
5) Advanced

Click on **Review + Create** and then click **Create**.

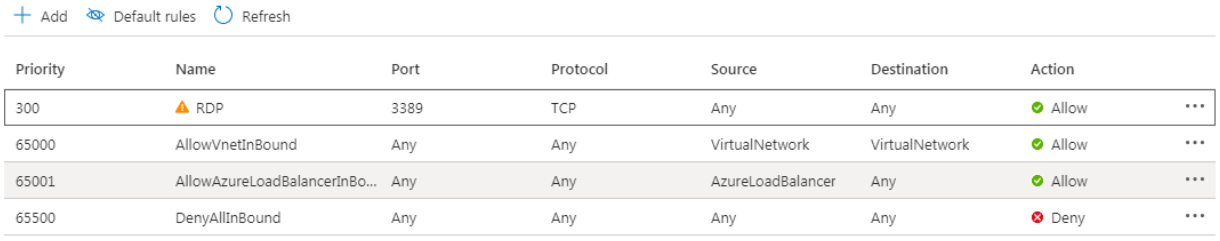
6) Network Security Groups:

1) search the Network Security Groups.

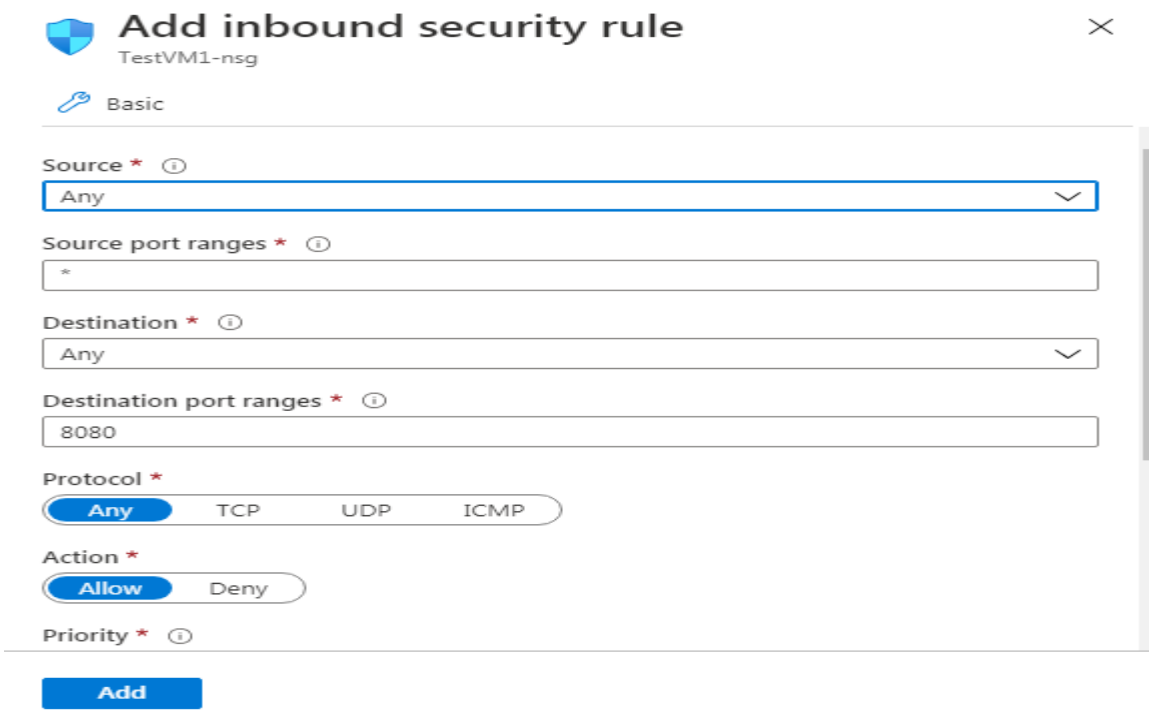
2) Now select the **TestVM1-nsg**



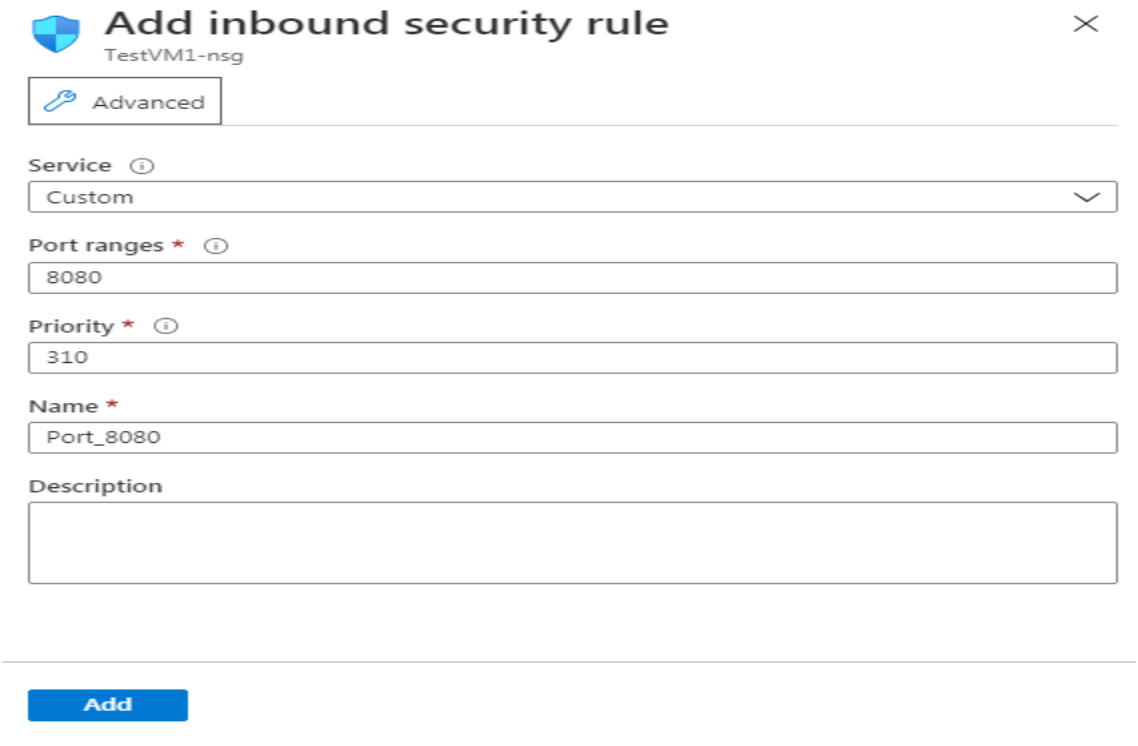
3) Select the Inbound security rules under Settings.



4) click on the Add button to add the security rules.



5) Click on the **Basic** tab.



6) select the **service** as **HTTP** and Name as **HTTP_8080** and click on Add button.

Note: Repeat the same steps for TestVM2-nsg

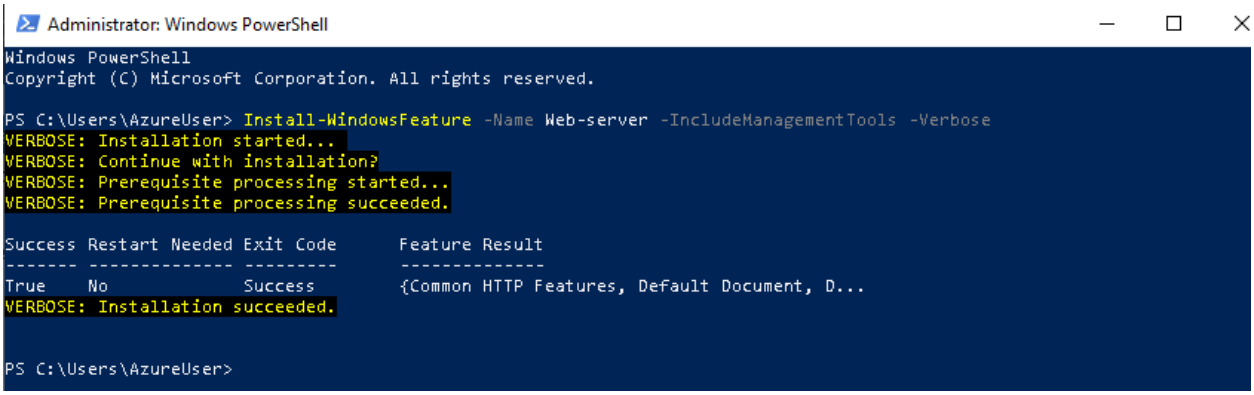
7) Now go to the **TestVM1** virtual Machine and select the **Networking** tab under the settings. And copy the **NIC Public IP.**

8) Using this IP address open the Remote Virtual Machine and similarly copy the **NIC Public IP** of second VM and open the remote virtual machine.

9) Now we need to create the IIS in the two VMS. For this open the Powershell as administrator in both the VMS and run the following command.

Install-WindowsFeature -Name Web-server -IncludeManagementTools –Verbose

10) After successful installation of IIS in the VMS the following message will be display in powershell command prompt.

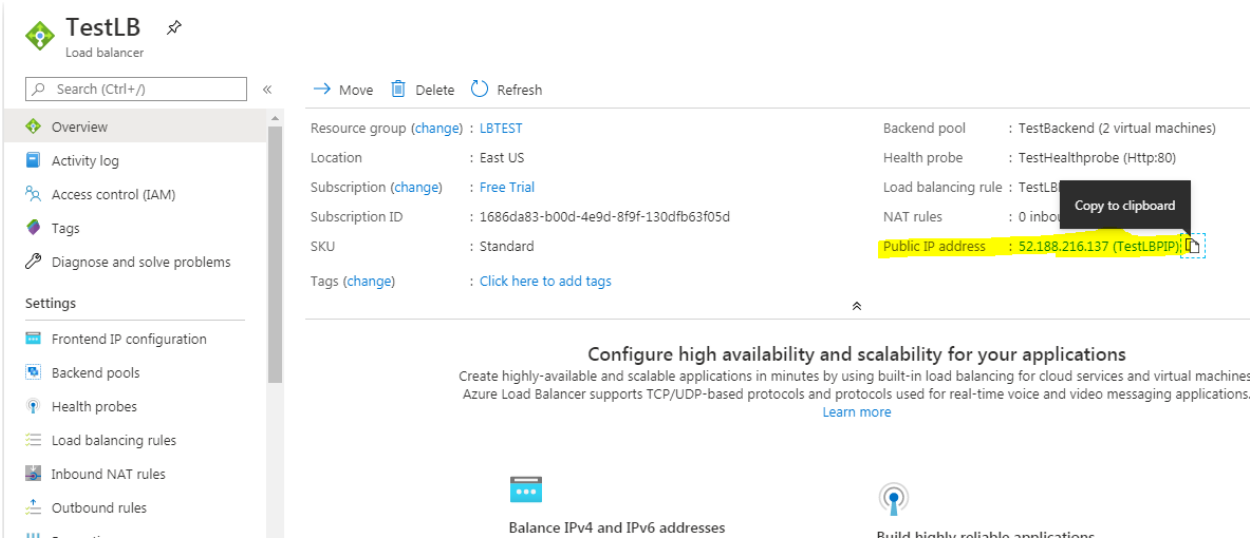


11) Run the Following two commands in powershell to remove the iisstart page and add the content in iisstart page.

Remove-Item C:\inetpub\wwwroot\iisstart.htm -Force -Recurse -Verbose

Add-Content -Path "C:\inetpub\wwwroot\iisstart.htm" -Value $("Hello LoadBalancer Node " + $env:COMPUTERNAME)

12) Now to the Test LB Load balancer and copy the public IP and paste it in browser.


TestLB
Load balancer
Search (Ctrl+/)
Move
Delete
Refresh
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Settings
Frontend IP configuration
Backend pools
Health probes
Load balancing rules
Inbound NAT rules
Outbound rules
Resource group (change) : LBTEST
Location : East US
Subscription (change) : Free Trial
Subscription ID : 1686da83-b00d-4e9d-8f9f-130dfb63f05d
SKU : Standard
Tags (change) : Click here to add tags
Backend pool : TestBackend (2 virtual machines)
Health probe : TestHealthprobe (Http:80)
Load balancing rule : TestLB
NAT rules : 0 inbo
Public IP address : 52.188.216.137 (TestLBPIP)
Copy to clipboard
Configure high availability and scalability for your applications
Create highly-available and scalable applications in minutes by using built-in load balancing for cloud services and virtual machines
Azure Load Balancer supports TCP/UDP-based protocols and protocols used for real-time voice and video messaging applications.
Learn more
Balance IPv4 and IPv6 addresses