



Azure Basic Load Balancer

Azure Load Balancer is a networking service provided by Microsoft Azure that distributes incoming network traffic across multiple servers or virtual machines (VMs) to ensure the high availability and reliability of applications and services. It operates at the Transport layer (Layer 4) of the OSI model and can balance both TCP and UDP traffic.

Here are some key features and functionalities of Azure Load Balancer:

1. **Traffic Distribution:** Azure Load Balancer distributes incoming traffic evenly among healthy instances of VMs or services based on configured load balancing rules.
2. **High Availability:** It enhances the availability of applications by providing redundancy and failover capabilities. If one instance or VM becomes unavailable, traffic is automatically rerouted to healthy instances.
3. **Health Probes:** Azure Load Balancer continuously monitors the health of backend instances by sending health probes. If an instance fails health checks, it is automatically removed from the load-balanced pool.
4. **Backend Pool Configuration:** You can define backend pools, which are groups of VMs or services that will receive traffic from the load balancer.
5. **Session Persistence:** Azure Load Balancer supports session persistence, ensuring that all requests from the same client are sent to the same backend instance.
6. **Public and Private Load Balancers:** You can deploy Azure Load Balancers as either public or internal load balancers, depending on whether they are accessible from the internet or limited to internal network traffic.
7. **Integration with Azure Services:** Azure Load Balancer seamlessly integrates with other Azure services such as Virtual Machines, Virtual Machine Scale Sets, Azure Kubernetes Service (AKS), and Azure Virtual Network Gateways.
8. **Security:** It offers security features such as Network Security Groups (NSGs) to control traffic flow and access to backend instances.

Overall, Azure Load Balancer plays a crucial role in ensuring the scalability, reliability, and performance of applications and services hosted on the Azure cloud platform.



Use cases of Load Balancer:

Azure Load Balancer serves various use cases across different scenarios and industries. Here are some common ones:

1. **Web Applications:** Load balancing web traffic across multiple instances of web servers ensures high availability and scalability for web applications. Whether it's a simple website or a complex web application, Azure Load Balancer can distribute incoming traffic effectively.
2. **Microservices Architecture:** In a microservices architecture, different components of an application are deployed as separate services. Azure Load Balancer helps in distributing traffic across these microservices, ensuring each component is responsive and scalable.

3. **Highly Available Services:** Critical services such as databases, messaging queues, and APIs can be made highly available by distributing traffic across multiple instances or replicas. Azure Load Balancer ensures that these services remain accessible even if some instances fail.
4. **Hybrid Cloud Scenarios:** In hybrid cloud setups where applications span across on-premises data centers and Azure, Azure Load Balancer facilitates seamless traffic distribution between on-premises and cloud-based resources.
5. **Disaster Recovery:** Load balancing traffic between primary and secondary data centers or between regions helps in disaster recovery scenarios. In case of a failure in one location, Azure Load Balancer can automatically redirect traffic to the secondary location.
6. **Internet-Facing Applications:** Public-facing applications such as e-commerce websites, gaming platforms, and media streaming services benefit from Azure Load Balancer to handle incoming internet traffic efficiently and ensure a smooth user experience.
7. **Internal Applications:** Internal line-of-business applications, intranet portals, and backend services within an organization can leverage Azure Load Balancer as an internal load balancer to distribute traffic across backend servers or services.
8. **DevOps and Continuous Integration/Continuous Deployment (CI/CD):** Azure Load Balancer can be integrated into DevOps pipelines to automate the deployment and scaling of applications. It ensures that newly deployed instances are seamlessly added to the load-balanced pool.
9. **Testing and Staging Environments:** Load balancing traffic in testing and staging environments helps in simulating real-world conditions and ensures that applications perform optimally before being deployed to production.

In this setup, we're configuring Azure Load Balancer to evenly distribute incoming web traffic across two virtual machines running Windows Server 2022 with Internet Information Services (IIS) installed. The end goal is to ensure high availability and scalability for web applications by automatically routing traffic to healthy instances and providing redundancy in case of failures. This setup enhances application reliability and performance by distributing the workload across multiple servers.

To begin with the Lab:

1. Now we're going to move on to the implementation of the load balancer. Before that, we are going to have a simple setup in place.
2. We are first going to look at the load balancer when it comes to the basic SKU.
3. For this, we are going to create two virtual machines based on Windows Server 2022. We are going to make both machines part of an availability set, that'll be part of a virtual network.
4. We'll install internet information services on both of the machines. We'll have sort of a default HTML page in place.
5. Now in your Azure Portal navigate to Virtual Machine and create a new virtual machine. Here you need to choose a resource group and then give your VM a name

and then choose your region. After that in the availability options, you have to choose Availability Set.

Basics Disks Networking Management Monitoring Advanced Tags Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ ▼

Resource group * ⓘ ▼
[Create new](#)

Instance details

Virtual machine name * ⓘ ✓

Region * ⓘ ▼

Availability options ⓘ ▼

6. Then you have to create your availability set keep things to default and you have to give it name then just create it.

Create availability set ×

Group two or more VMs in an availability set to ensure that at least one is available during planned or unplanned maintenance events. [Learn more](#)

Name *

 ✓

Fault domains ⓘ



2

Update domains ⓘ



5

Use managed disks ⓘ

No (Classic) Yes (Aligned)

7. Then you have to choose your Image as Windows server 2022 datacenter.

Availability set * ⓘ

(new) loadset ▼

Create new

Security type ⓘ

Standard ▼

Image * ⓘ

Windows Server 2022 Datacenter - x64 Gen2 ▼

[See all images](#) | [Configure VM generation](#)

This image is compatible with additional security features. [Click here to swap to the Trusted launch security type.](#)

VM architecture ⓘ

Arm64
 x64

Arm64 is not supported with the selected image.

- Now give a username and password then in the inbound rules you have to choose HTTP as your inbound rule.

Administrator account

Username * ⓘ

loaduser ✓

Password *

..... ✓

Confirm password *

..... ✓

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports * ⓘ

None
 Allow selected ports

Select inbound ports *

HTTP (80), RDP (3389) ▼

All traffic from the internet will be blocked by default. You will be able to change inbound port rules in the VM > Networking page.

- Now in the Networking tab you have to create a new Virtual Network, for that click on new and just change the name of your VN.

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution.

[Learn more](#)

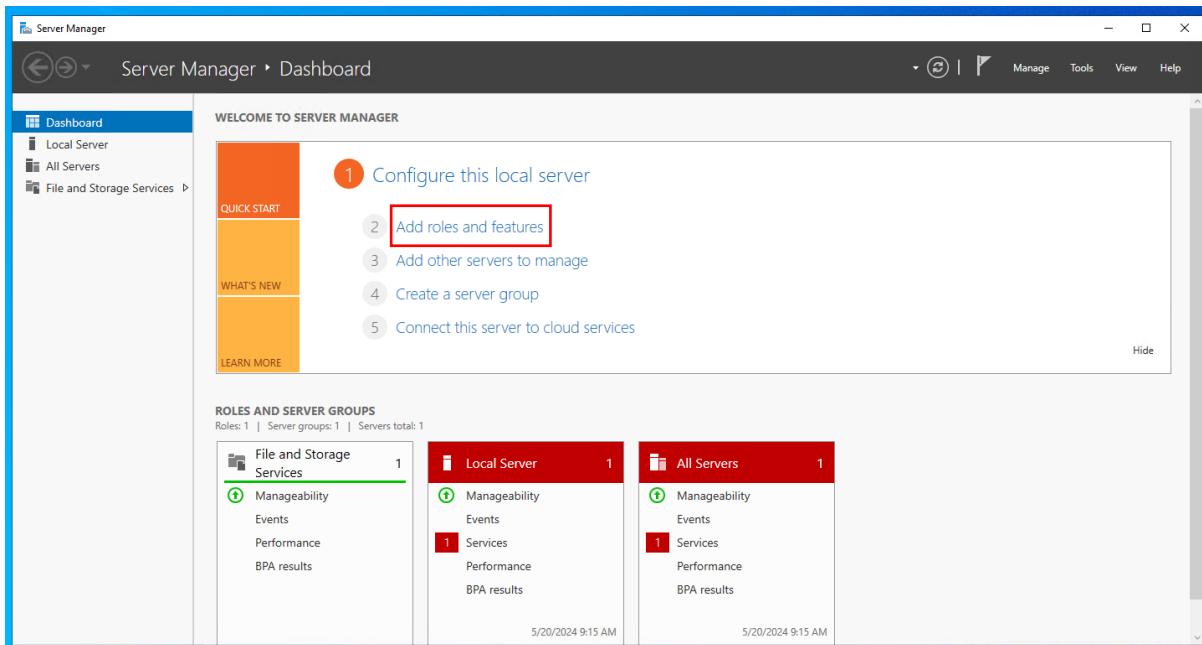
Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network *	<input type="text" value="(new) load-VN"/> ▼ Create new
Subnet *	<input type="text" value="(new) default (10.0.0.0/24)"/> ▼
Public IP	<input type="text" value="(new) loadVM1-ip"/> ▼ Create new

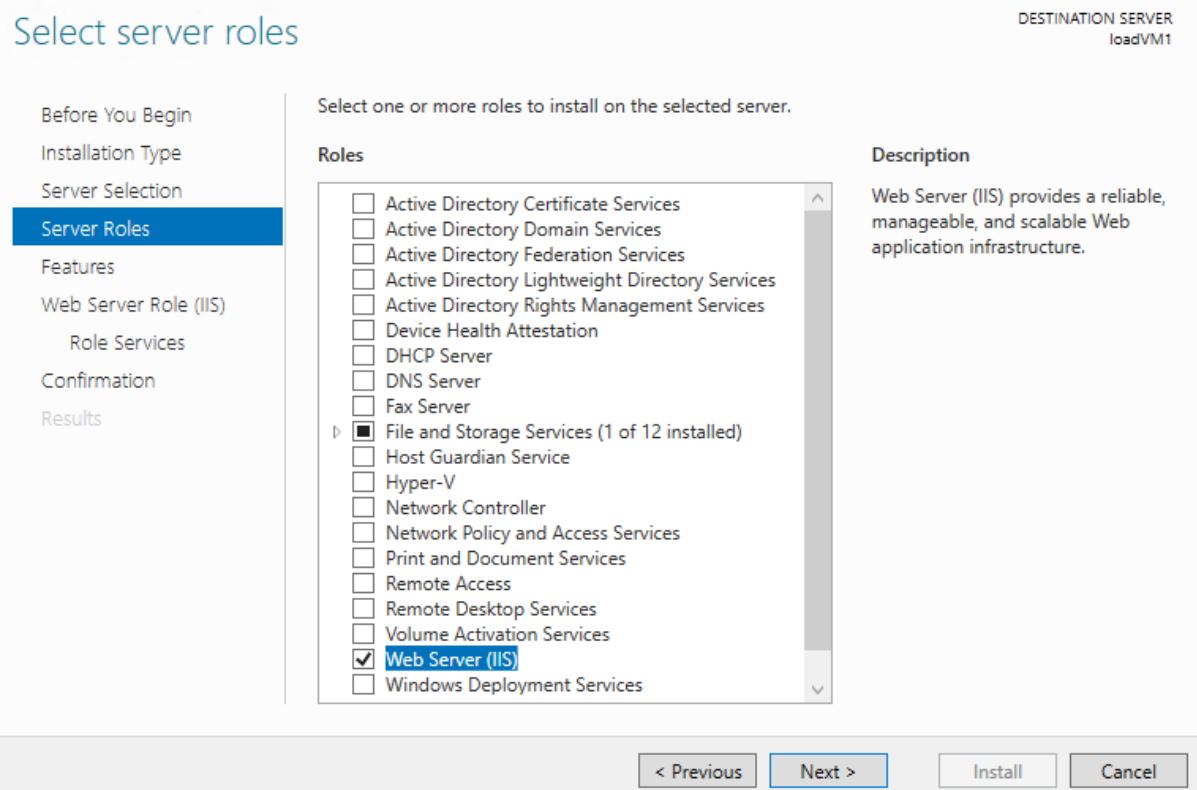
10. After that go ahead and create your Virtual Machine. Once your resources are deployed then you need to RDP into your VM.

11. Now in your VM you have to install IIS. For that you need to click on Add roles and features in your Server Manager.



12. Then you have to click on next until you reach to Server Roles and from here you have to add Web Server (IIS) as shown in the snapshot.

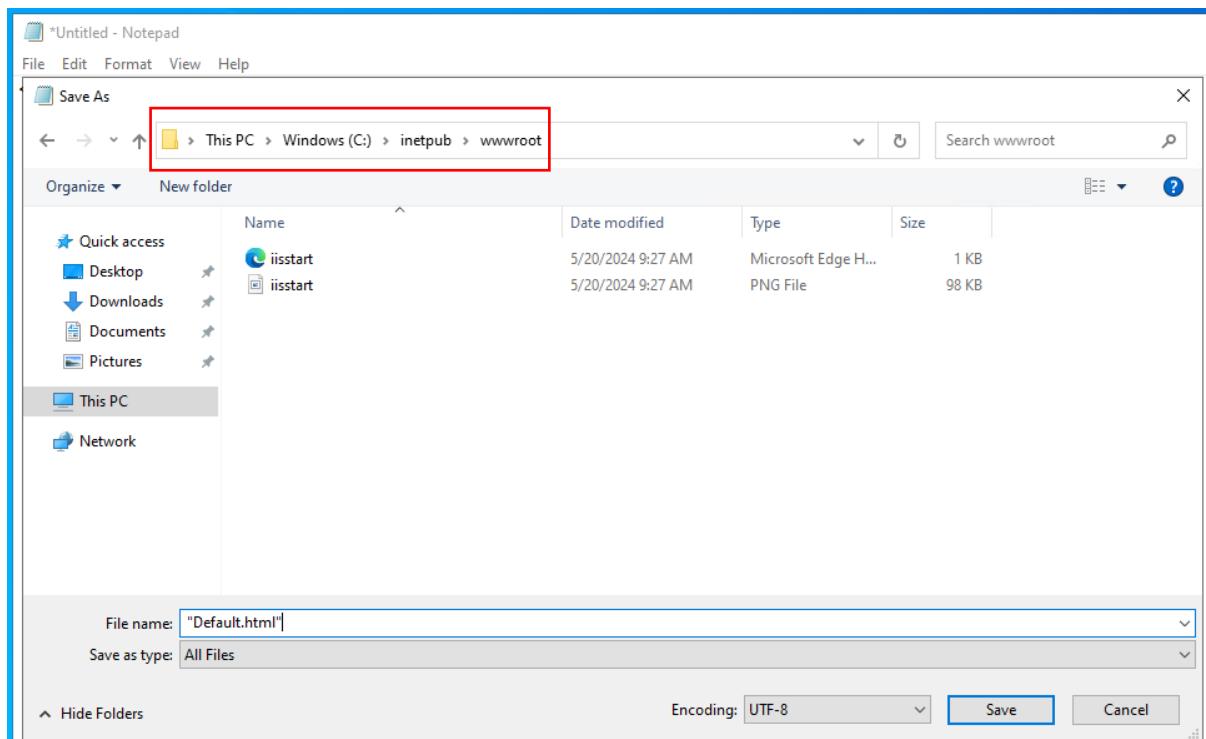
13. After that move to the installation and install it.



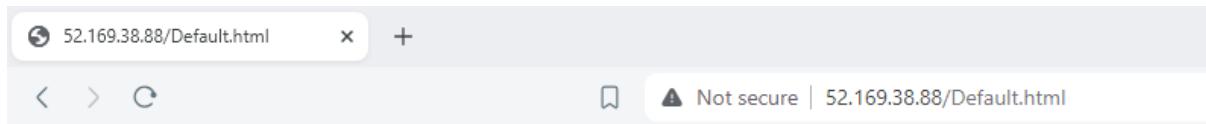
14. Once the installation is completed then you have to open the Note Pad in your VM and you have to write this.

```
*Untitled - Notepad
File Edit Format View Help
<h1>This is Load-VM 1</h1>
```

15. And you have to save this file in this highlighted path as shown in the snapshot.



16. Now to see that it is working as expected you have to copy the Public IP address of your VM and paste it in a new tab and append it with the file name which you saved in your VM. You will see that result as expect.



17. After that you are going to create a new Windows VM and do the same thing as you did in the first VM.

18. Again choose your resource group then give your VM a name then choose availability set in availability options.

Subscription * ⓘ Azure Pass - Sponsorship

Resource group * ⓘ new-grp [Create new](#)

Instance details

Virtual machine name * ⓘ loadVM2

Region * ⓘ (Europe) North Europe

Availability options ⓘ Availability set

A screenshot of the Azure portal 'Create a new virtual machine' form. It shows the subscription 'Azure Pass - Sponsorship', resource group 'new-grp', and virtual machine name 'loadVM2'. The region is set to '(Europe) North Europe' and the availability option is 'Availability set'.

19. Then choose your availability set, after that choose your image and size. Then give your machine a username and password. Also, choose HTTP as your inbound rule.

Availability set * ⓘ

Create new

Security type ⓘ

Image * ⓘ

Windows Server 2022 Datacenter - x64 Gen2
[See all images](#) | [Configure VM generation](#)

This image is compatible with additional security features. [Click here to swap to the Trusted launch security type.](#)

Size * ⓘ

[See all sizes](#)

Enable Hibernation ⓘ

Hibernate does not currently support Availability Sets. [Learn more](#) ↗

Administrator account

Username * ⓘ

Password *

Confirm password *

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports * ⓘ

None
 Allow selected ports

Select inbound ports *

All traffic from the internet will be blocked by default. You will be able to change inbound port rules in the VM > Networking page.

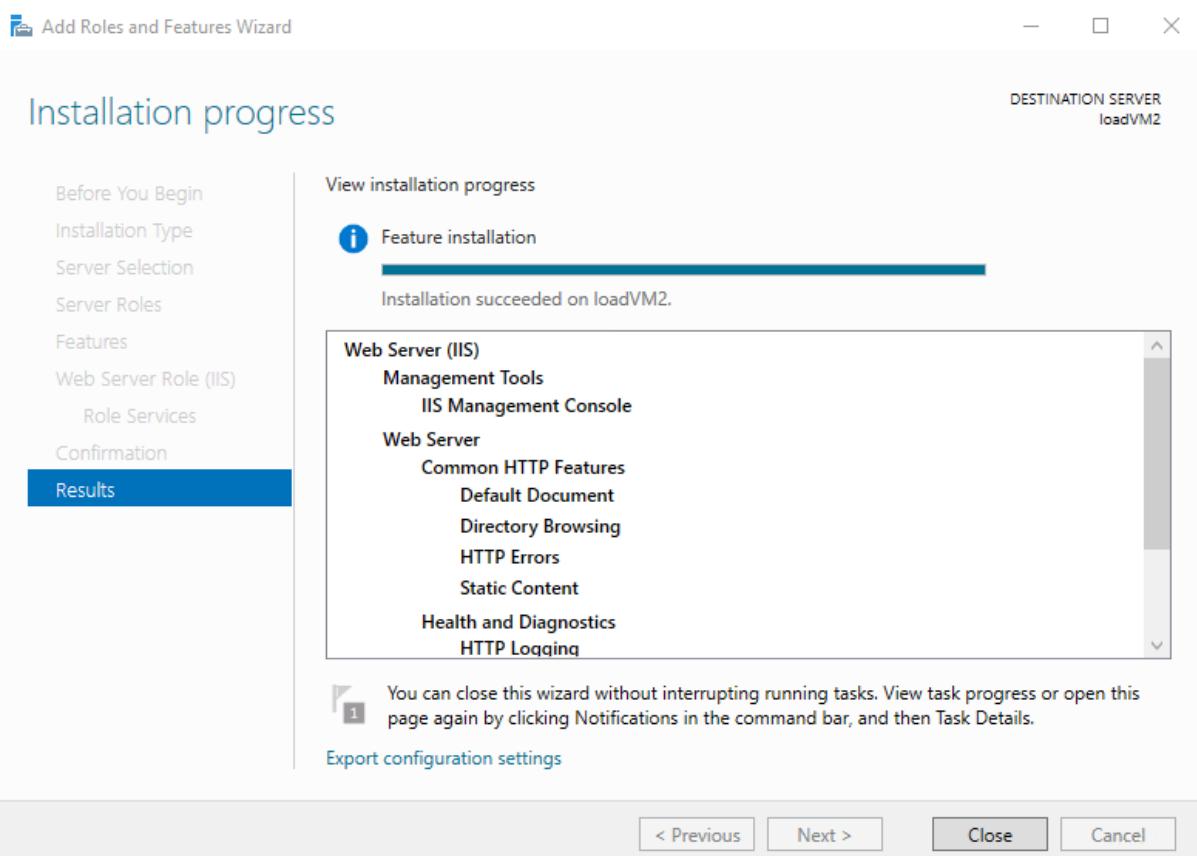
20. Then in the network interface choose the same virtual network, the same subnet, and the new public IP address. After that just move to the Review page and create your Virtual machine.

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network *	load-VN
	Create new
Subnet *	default (10.0.0.0/24)
	Manage subnet configuration
Public IP	(new) loadVM2-ip
	Create new

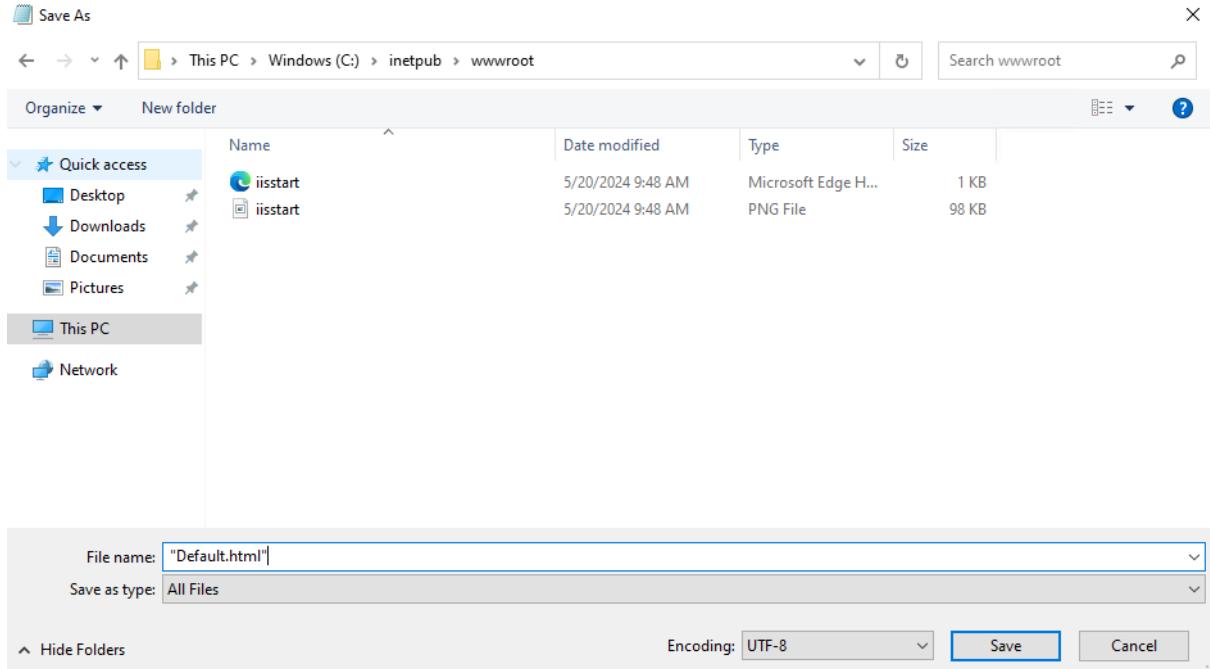
21. Once the deployment is complete for your VM then just RDP into it and install the web server (IIS) as you did for the first VM.
22. Once your Web Server is installed then open your Note Pad.



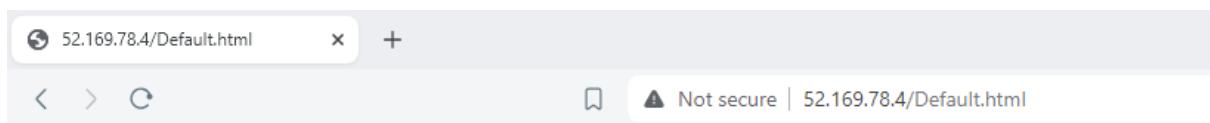
23. Then write this on your pad and save it in the same location as before.

```
*Untitled - Notepad
File Edit Format View Help
<h1> This is Load-VM 2</h1>
```

24. Below is the location where you have to save it.



25. Then copy the public IP of your 2nd VM and append it with the file name.



This is Load-VM 2



Basic Load Balancer Deployment

In this process, we're refining the network configuration by removing public IP addresses from the virtual machines and consolidating network security groups (NSGs) to enhance security and streamline management. The end goal is to deploy an Azure Load Balancer with basic SKU, configuring it to evenly distribute incoming traffic to the backend pool of virtual machines. This setup aims to improve application availability, reliability, and scalability by efficiently managing incoming traffic across multiple instances.

1. We're going to make some changes first. I'm going to ensure that we don't have a public IP address assigned to both machines. We'll remove that because that's not required. We'll also just have one NSG in place and one network security group.
2. Now in your VM you need to go to network settings and then go to network interface.

loadVM2 | Network settings

Virtual machine

Search

This is a new experience. [Please provide feedback](#)

Connect

- Connect
- Bastion
- Windows Admin Center

Networking

- Network settings** (highlighted with a red box)
- Load balancing
- Application security groups

Network interface / IP configuration
loadvm2877 (primary) / ipconfig1 (primary)

Essentials

Network interface	: loadvm2877
Virtual network / subnet	: load-VN / default
Public IP address	: 52.169.78.4
Private IP address	: 10.0.0.5

- Then in the network interface go to IP configuration and the open the ip config which you will see inside it.

loadvm2877 | IP configurations

Network interface

IP Settings

Enable IP forwarding

Overview

Activity log

Access control (IAM)

Tags

Settings

IP configurations (highlighted with a red box)

DNS servers

Network security group

Properties

Locks

Monitoring

Automation

Help

Virtual network: load-VN

Gateway load balancer: None

Subnet: default (10.0.0.0/24) 249 free IP addresses

Add Make primary Delete

Name	IP Version	Type	Private IP Address	Public IP Address
ipconfig1	IPv4	Primary	10.0.0.5 (Dynamic)	52.169.78.4 (loadVM2-ip)

- Then you have to uncheck the associate public IP address part and then click on save.

Public IP address settings

i Unchecking "Associate Public IP address" will disassociate the public IP address from this network interface card 'ipconfig1'. [Learn more](#)

Associate public IP address

Save

Cancel

- Then you have to do the same for the other VM to remove the Public IP address.

- Now if you go to virtual machines, you will see that the Public IP address has been removed.

Name	Type	Subscription	Resource group	Location	Status	Operating system	Size	Public IP address	Disk
loadVM1	Virtual machine	Azure Pass - Sponsorship	NEW-GRP	North Europe	Running	Windows	Standard_DS1_v2	-	1
loadVM2	Virtual machine	Azure Pass - Sponsorship	NEW-GRP	North Europe	Running	Windows	Standard_DS1_v2	-	1

- Now go to the All Resources tab and delete your public IP address because we don't need it.

Name	Type	Resource group	Location	Subscription
loadVM1-ip	Public IP address	new-grp	North Europe	Azure Pass - Sponsorship
loadVM2-ip	Public IP address	new-grp	North Europe	Azure Pass - Sponsorship

- Now we are going to create a new Network Security Group (NSG), for that in the marketplace search for it.
- You just have to choose your resource group and create your NSG.

Create network security group

Basics Tags Review + create

Project details

Subscription * Azure Pass - Sponsorship

Resource group * new-grp
Create new

Instance details

Name * subnet-ns

Region * North Europe

- Now in a new tab you have to open all resources and filter out the NSGs. Then open them both in a new tab separately.

All resources ...

Default Directory (pulkitkumar2711@gmail.onmicrosoft.com)

+ Create Manage view Refresh Export to CSV Open query | Assign tags Delete

loadvm nsg Subscription equals all Resource group equals all Type equals all Location equals all Add filter

0 Recommendations Changed resources Unsecure resources

Name ↑↓	Type ↑↓	Resource group ↑↓	Location ↑↓
loadVM1-nsg	Network security group	new-grp	North Europe
loadVM2-nsg	Network security group	new-grp	North Europe

11. Then go to inbound rules and remove both of them.

loadVM1-nsg | Inbound security rules ...

Network security group

Search Add Hide default rules Refresh Delete Give feedback

Network security group security rules are evaluated by priority using the combination of source, source port, destination, destination port, and protocol to allow or deny the traffic. A security rule can't have the same priority and direction as an existing rule. You can't delete default security rules, but you can override them with rules that have a higher priority. [Learn more](#)

Priority ↑↓	Name ↑↓	Port ↑↓	Protocol ↑↓	Source ↑↓	Destination ↑↓	Action ↑↓
300	RDP	3389	TCP	Any	Any	Allow
320	HTTP	80	TCP	Any	Any	Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInbound	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny

12. Then go to network interfaces and dissociate it. Then move to overview and delete your NSG.

loadVM1-nsg | Network interfaces ...

Network security group

Search Associate Refresh Dissociate

Network security group security rules are evaluated by priority using the combination of source, source port, destination, destination port, and protocol to allow or deny the traffic. A security rule can't have the same priority and direction as an existing rule. You can't delete default security rules, but you can override them with rules that have a higher priority. [Learn more](#)

Name ↑↓	Public IP address ↑↓	Private IP address ↑↓	Virtual machine ↑↓
loadvm1331	-	10.0.0.4	loadVM1

13. You have to do the same for other VM also.

14. After that you have to open the New NSG and add two inbound rules in it one for HTTP and other one for RDP.

subnet-nsg | Inbound security rules ...

Network security group

Search Add Hide default rules Refresh Delete Give feedback

Network security group security rules are evaluated by priority using the combination of source, source port, destination, destination port, and protocol to allow or deny the traffic. A security rule can't have the same priority and direction as an existing rule. You can't delete default security rules, but you can override them with rules that have a higher priority. [Learn more](#)

Priority ↑↓	Name ↑↓	Port ↑↓	Protocol ↑↓	Source ↑↓	Destination ↑↓	Action ↑↓
100	AllowAnyHTTPInbound	80	TCP	Any	Any	Allow
110	AllowAnyRDPI inbound	3389	TCP	Any	Any	Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInbound	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny

15. Then go to Subnets and associate your subnet with it.

The screenshot shows the Azure portal interface for managing a Network Security Group (NSG). On the left, there's a navigation sidebar with options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Subnets, Properties, Locks, Monitoring, Automation, and Help. The 'Subnets' section is currently selected. On the right, a modal window titled 'Associate subnet' is open, showing a list of subnets under a virtual network named 'load-VN (new-grp)'. A specific subnet named 'default' is highlighted. At the bottom of the modal is an 'OK' button.

16. Now we are going to deploy our Load Balancer. For that in the marketplace search for the load balancer and choose this service accordingly.

Load Balancer

Microsoft

This screenshot shows the Azure Marketplace listing for the 'Load Balancer' service. It includes the service icon, the title 'Load Balancer', a 'Microsoft | Azure Service' badge, a '4.6 (121 ratings)' rating, and a prominent 'Create' button. Below the main title, there's a 'Plan' section with a dropdown menu set to 'Load Balancer'.

17. First we need to choose our Resource Group.

This screenshot shows the 'Basics' tab of the Azure Load Balancer creation wizard. It displays the subscription as 'Azure Pass - Sponsorship' and the resource group as 'new-grp'. Other tabs available include Frontend IP configuration, Backend pools, Inbound rules, Outbound rules, Tags, and Review + create.

Azure load balancer is a layer 4 load balancer that distributes incoming traffic among healthy virtual machine instances. Load balancers uses a hash-based distribution algorithm. By default, it uses a 5-tuple (source IP, source port, destination IP, destination port, protocol type) hash to map traffic to available servers. Load balancers can either be internet-facing where it is accessible via public IP addresses, or internal where it is only accessible from a virtual network. Azure load balancers also support Network Address Translation (NAT) to route traffic between public and private IP addresses. [Learn more.](#)

Project details

Subscription *: Azure Pass - Sponsorship

Resource group *: new-grp

Create new

18. Then we need to give a name to it and choose our region then we need to choose our SKU which will be basic for the time being. And the type should be Public. Then move to next option.

Instance details

Name * ✓

Region * ✓

SKU * Standard (Recommended) Gateway Basic (Retiring soon)

i Microsoft recommends Standard SKU load balancer for production workloads; Basic SKU will be retired on September 30, 2025. [Learn more](#).

Type * Public Internal

Tier * Regional Global

19. In front-end IP configuration you have to click on Add.

Basics Frontend IP configuration Backend pools Inbound rules Outbound rules Tags Review + create

A frontend IP configuration is an IP address used for inbound and/or outbound communication as defined within load balancing, inbound NAT, and outbound rules.

+ Add a frontend IP configuration

Name ↑↓	IP address ↑↓
Add a frontend IP to get started	

20. After that you need to give it a name then for public IP click on Create new.

Add frontend IP configuration

X

DemoLoadBalancer

Name *

frontend-ip

IP version

IPv4

IPv6

Public IP address *

Choose public IP address

[Create new](#)

Add a public IP address

Name *

load-ip

SKU

Basic

(i) On 30 September 2025, Azure Basic Public IP will be retired. [Learn more.](#)

Tier

Regional

Static IPs are assigned at the time the resource is created and released when the resource is deleted. Dynamic IPs are assigned when associating the IP to a resource and is released when you stop, restart, or delete a resource. Dynamic is only available for Basic SKU.

Assignment

Dynamic

Static

[Save](#)

[Cancel](#)

21. Now we are going to add the backend pool or in simple words our VMs here. For that click on add.

Basics Frontend IP configuration **Backend pools** Inbound rules Outbound rules Tags Review + create

A backend pool is a collection of resources to which your load balancer can send traffic. A backend pool can contain virtual machines, virtual machine scale sets, and containers.

[+ Add a backend pool](#)

Name	Virtual network	Resource Name	Network interface	IP address	Availability zone
Add a backend pool to get started					

22. Then you have to give it a name then choose the virtual network after in the IP configuration you have to click on Add and choose both of the load balancers. Then just click on save.

Add backend pool ...

Name * PoolA
Virtual network ⊞ load-VN (new-grp)

IP configurations

IP configurations associated to virtual machines and virtual machine scale sets must be in same location as the load balancer and be in the same virtual network.

+ Add | X Remove

Resource Name	Resource group	Type	IP configuration	IP Address	Availability set
LOADVM1	NEW-GRP	Virtual machine	ipconfig1	10.0.0.4	LOADSET
loadVM2	new-grp	Virtual machine	ipconfig1	10.0.0.5	LOADSET

23. After that move to the review page and create your load balancer. Once the deployment is complete click on go to resources.

Microsoft.LoadBalancer-20240520162523 | Overview

Deployment

Search | Delete | Cancel | Redeploy | Download | Refresh

Overview

- Inputs
- Outputs
- Template

Your deployment is complete

Deployment name : Microsoft.LoadBalancer-20240520162523
Subscription : Azure Pass - Sponsorship
Resource group : new-grp

Start time : 20/5/2024, 4:36:00 pm
Correlation ID : 77344004-38ca-4f8b-98d4-2a9ea86d064f

> Deployment details

< Next steps

Go to resource

24. This is the dashboard of the load balancer you can explore it if you want.

DemoLoadBalancer | Load balancer

Move | Delete | Refresh | Give feedback

Overview

- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Settings
- Monitoring
- Automation
- Help

DemoLoadBalancer is on Basic SKU which is deprecating soon. Learn more about standard SKU and migration steps →

Essentials

Resource group (move) : new-grp	Backend pool : PoolA (2 virtual machines)
Location : North Europe	Load balancing rule : -
Subscription (move) : Azure Pass - Sponsorship	Health probe : -
Subscription ID : 3541d15a-44aa-4f6e-a120-1b7a6d5925bf	NAT rules : 0 inbound
SKU : Basic	Tier : Regional
Tags (edit) : Add tags	

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](#)

25. In the front-end IP configuration, we can see that we have our public IP address.

DemoLoadBalancer | Frontend IP configuration

Load balancer

Search | Add | Refresh | Give feedback

Frontend IP configuration

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Settings

Filter by name...

Name ↑↓	IP address ↑↓	Rules count ↑↓
frontend-ip	load-ip	0

26. And in the back-end pools we have our VMs running.

Backend pool	Resource Name	IP address	Network interface	Availability zone	Rules count	Resource Status
PoolA	loadVM1	10.0.0.4	loadvm1331	-	0	Running
PoolA	loadVM2	10.0.0.5	loadvm2877	-	0	Running



Basic Load Balancer Configuration

In this process, we're configuring an Azure Load Balancer to enhance application reliability and optimize resource usage. We're adding health probes to ensure traffic is directed only to healthy backend instances, defining load balancing rules to distribute incoming requests efficiently, and setting up inbound NAT rules for remote access to individual virtual machines. The end goal is to establish a robust load-balancing setup that improves application availability and facilitates remote management of backend resources.

1. Now in this we will add some configurations in our Load balancer.
2. Inside of your load balancer click on heath probes and click on add.

DemoLoadBalancer | Health probes

Load balancer

Search Add Refresh Give feedback

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Settings

Frontend IP configuration Backend pools

Health probes

3. Then in there you need to give it a name and choose the same configuration as shown below.

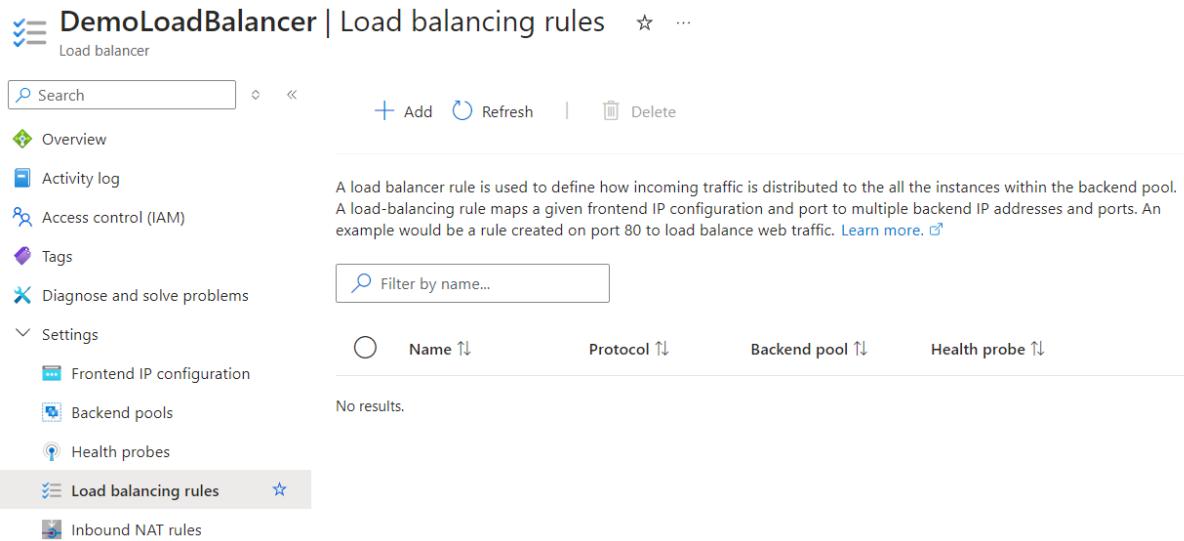
Add health probe

DemoLoadBalancer

Health probes are used to check the status of a backend pool instance. If the health probe fails to get a response from a backend instance then no new connections will be sent to that backend instance until the health probe succeeds again.

Name *	ProbeA
Protocol *	TCP
Port *	80
Interval (seconds) *	5
Used by *	Not used

4. Now we have to define load balancing rules. For that click on add. Moreover, the load balancing rule will understand how to direct requests onto the backend machines.

Screenshot of the Azure Load Balancer Overview page for 'DemoLoadBalancer'. The left sidebar shows navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings (Frontend IP configuration, Backend pools, Health probes), Load balancing rules (selected), and Inbound NAT rules. The main content area displays a table header for 'Load balancing rules' with columns: Name ↑, Protocol ↑, Backend pool ↑, and Health probe ↑. A note states: 'A load balancer rule is used to define how incoming traffic is distributed to all the instances within the backend pool. A load-balancing rule maps a given frontend IP configuration and port to multiple backend IP addresses and ports. An example would be a rule created on port 80 to load balance web traffic.' A search bar at the top right says 'Filter by name...'. Below the table, it says 'No results.'

- Now here you have to give it a name first then choose your frontend IP address and backend pool. Then the port and backend port are both 80. After that choose your health probe and click on save.

Add load balancing rule

DemoLoadBalancer

Name *

IP Version *

IPv4
 IPv6

Frontend IP address * ⓘ

frontend-ip (Dynamic)

Backend pool * ⓘ

PoolA

Protocol

TCP
 UDP

Port *

80

Backend port * ⓘ

80

Health probe * ⓘ

ProbeA (TCP:80)

Create new

- Now you have to go to frontend IP configuration and here you will see that the Public IP address has been assigned. Now copy this IP and paste it in a new tab or browser.

DemoLoadBalancer | Frontend IP configuration

Load balancer

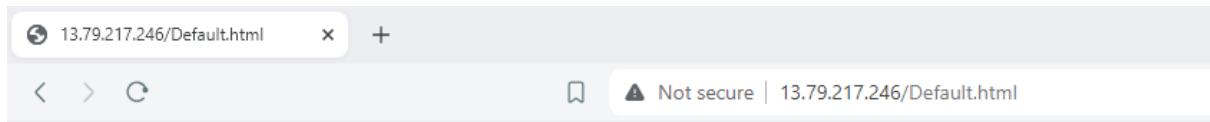
Search Add Refresh Give feedback

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Settings

Frontend IP configuration Backend pools

Name ↑↓	IP address ↑↓	Rules count ↑↓
frontend-ip	13.79.217.246 (load-ip)	1

7. Now it has directed the request to Load VM2.
8. So, now the load balancer is directing requests from, let's say my laptop onto loadvm2. If another user tries to again go onto the public IP address of the load balancer, the load balancer might direct the request. Let's say onto loadvm1.



This is Load-VM 2

NAT Rules

1. Now you have to go to Inbound NAT rules and click on Add.

DemoLoadBalancer | Inbound NAT rules

Load balancer

Search Add Refresh Give feedback

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

Name ↑↓	Frontend IP ↑↓	Frontend port/range ↑↓
No results.		

2. Then you have to give it a name and choose the target virtual machine which is load VM1 and then we have to choose network IP and frontend IP. Then the frontend port

is random which we wrote is 5000 and we want to direct our request to the backend port which is 3389. After that just click on save.

Add inbound NAT rule

DemoLoadBalancer

An inbound NAT rule forwards incoming traffic sent to a selected IP address and port combination to a specific virtual machine.

Name *	loadVM1
Target virtual machine	LOADVM1
Network IP configuration * ⓘ	ipconfig1 (10.0.0.4)
Frontend IP address * ⓘ	frontend-ip (13.79.217.246)
Frontend Port *	5000
Service Tag *	Custom
Backend port *	3389
Protocol	<input checked="" type="radio"/> TCP <input type="radio"/> UDP

3. After that we do the same for virtual machine 2 and you can copy the configuration from the snapshot below.

Add inbound NAT rule

DemoLoadBalancer

An inbound NAT rule forwards incoming traffic sent to a selected IP address and port combination to a specific virtual machine.

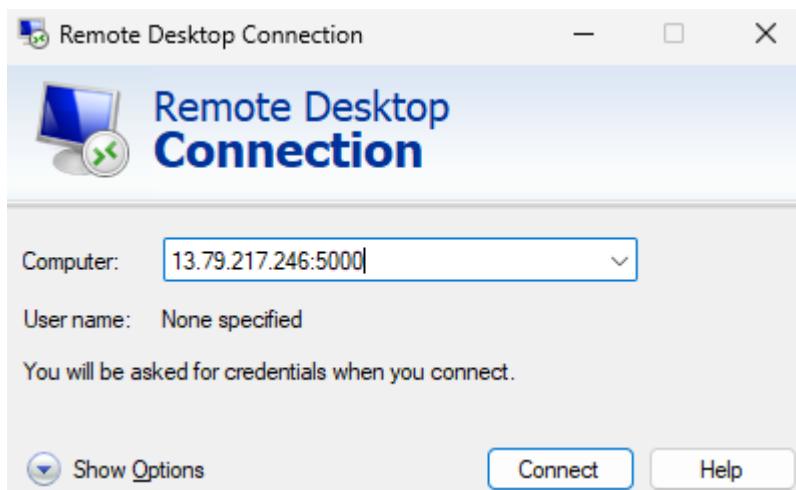
Name *	loadVM2
Target virtual machine	LOADVM2
Network IP configuration * ⓘ	ipconfig1 (10.0.0.5)
Frontend IP address * ⓘ	frontend-ip (13.79.217.246)
Frontend Port *	5001
Service Tag *	Custom
Backend port *	3389

4. Below you can see that both the VMs are added.

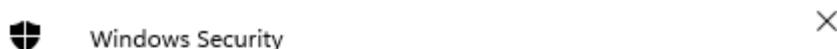
DemoLoadBalancer | Inbound NAT rules

Name	Frontend IP	Frontend port/range	Target	Service
loadVM1	13.79.217.246	5000	loadVM1	RDP (TCP/3389)
loadVM2	13.79.217.246	5001	loadVM2	RDP (TCP/3389)

5. Now you can do the RDP into your machine using the Public IP address from the Frontend IP.
6. Paste the IP address and append it with your front-end port which is 5000 for load VM1.



7. Now you can see that it is asking to enter the VM credentials.



Remember me

OK

Cancel

8. Below you will see that the connection with loadVM1 is going to establish.

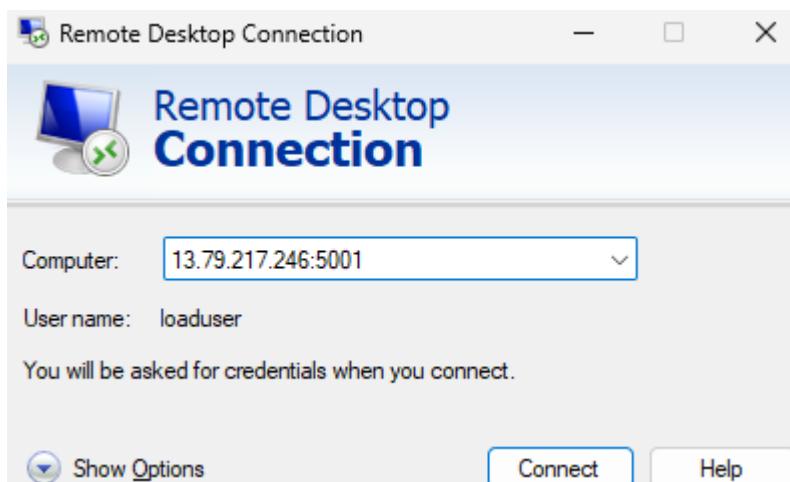


9. And you can see that we are in Load VM1.



10. Now close it and open load VM2 by following the same steps.

11. This time just change the IP with 5001 and click on connect.



12. Below you can see that we are going to connect with Load VM2.



13. Once you are done with all this just delete all of your resources.