

Understanding Azure Virtual Network Peering And configuring Peering Between Two Virtual Machines

What Is Azure Virtual Network Peering

Virtual network peering enables you to seamlessly connect two or more virtual networks in Azure using Microsoft's backbone infrastructure. Traffic between the peered network stays private and never travels over the public internet, allowing resources in one virtual network to communicate with resources in another as if they were in the same network.

Supported Peering in Azure

- Virtual Network Peering: Connecting virtual networks within the same region.
- Global Virtual Network Peering: Connecting virtual networks across Azure regions.

Step 1: Create a Virtual Network

In our previous article we covered what a virtual network is and how to create a virtual network. So, in this article I will be creating a second virtual network called "TechbysaraVNET2". This VNet will be in the same resource group as our previous VNet.

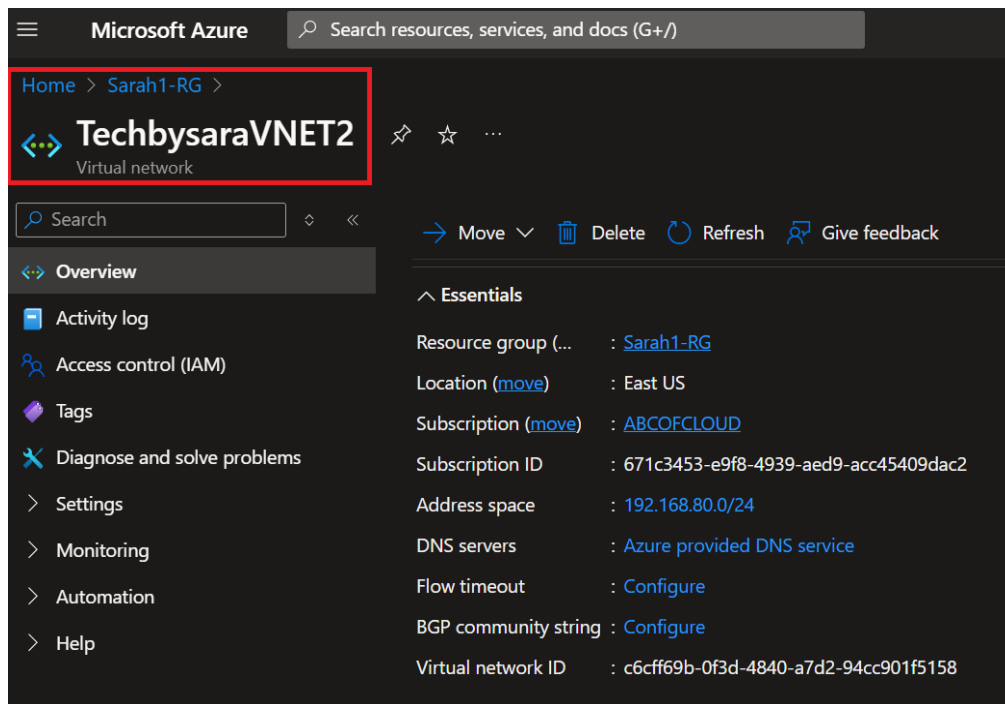
1. In the [Azure portal](#) > Search for **Virtual Network** > Select from **Marketplace**.

Project details:

2. Subscription > **ABCOFCLOUD**.
3. Resource group > **Sarah1-RG**.

Instance details:

4. Virtual network name > **TechbysaraVNET2**.
5. Region > **East US**.
6. IP address space > **192.168.8.0/24**
7. Subnet name > **HR-Subnet**.
8. Starting address > **192.168.80.0**
9. Size > **/26 (64 addresses)**.
10. Select > **Review + Create**, and when validation passes > select **Create**.



Step 2: Create a Virtual Machine

In our previous article, we explained what a virtual machine and virtual network are and how to create these resources on Azure. So, in this article I will be creating two virtual machines called “TechbysaraVM1” and “TechbysaraVM2”. These VMs will be located in the same resource group.

Note: RDP port 3389 is exposed to the internet. This is only recommended for testing. For production environments, it is recommended to use VPN or private connection.

Project details:

1. Subscription > **ABCOFCLOUD**.
2. Resource group > **Sarah1-RG**.

Instance details > for both VMS:

3. Virtual machine name > **TechbysaraVM1 and TechbysaraVM2**.
4. Region > **East US**.
5. Availability options > **Availability zone**.
6. Availability zone > **Zone 1**.
7. Image > **Windows Server 2019 Datacenter - Gen2**.
8. Size > **Standard B2ms (2 vcpus, 8 GiB memory)**.

Administration account:

9. Username > **Enter your preferred username. For this demo we will use (Techbysara) as the username.**

10. Password and confirm password > **Provide a secure password of at least 12 characters long.**

Inbound port rules:

11. Public inbound ports > **Allow selected ports.**
12. Select inbound ports > **RDP (3389)**

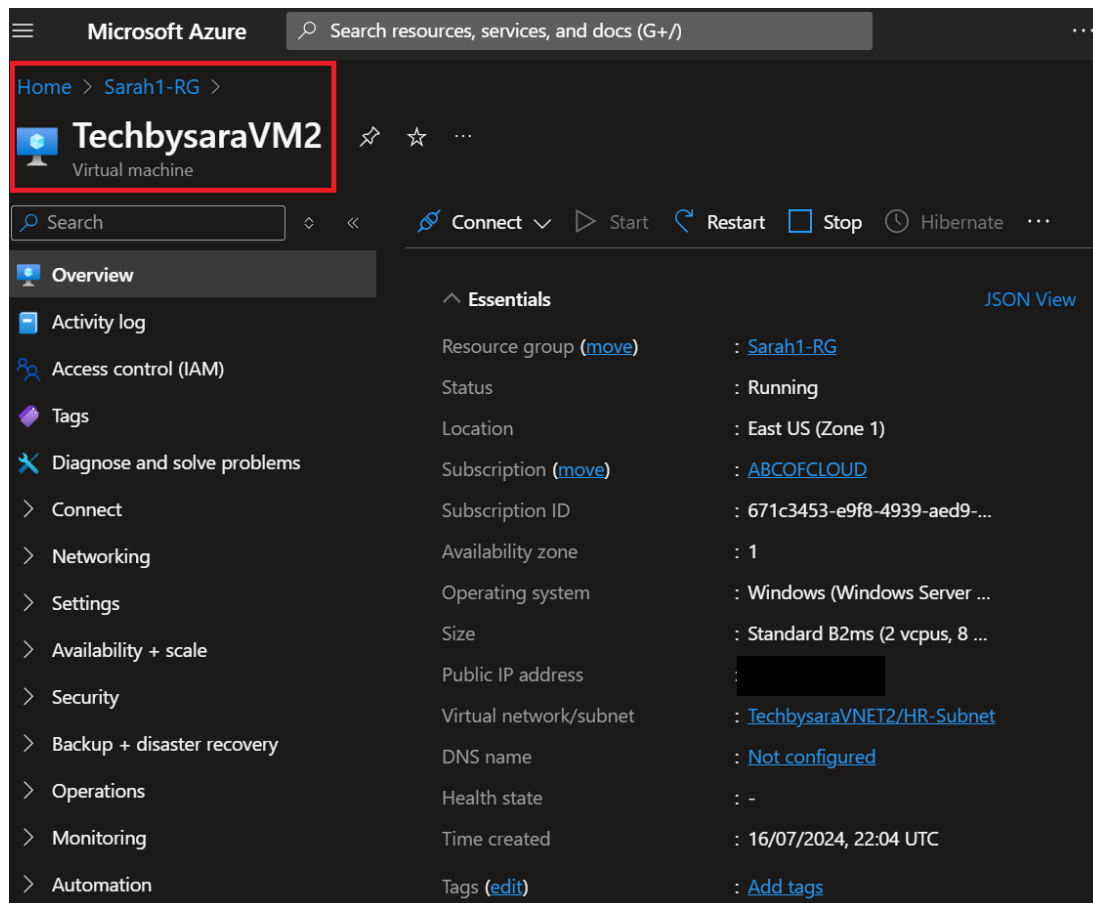
Networking tab > Network interface:

1. Virtual network > **TechbysaraVNET and TechbysaraVNET2.**
2. Subnet > **Project-Subnet(192.168.70.40/29) and HR-Subnet (192.168.80.0/26).**
3. Public IP > **Azure created a Public IP for both VMs.**
4. Nic network security group > **Basic.**
5. Public inbound ports > **Allow selected ports.**
6. Select inbound ports > **RDP (3389).**
7. Select > **Review + create**, and when validation passes, select > **Create.**

The screenshot displays the Microsoft Azure portal interface. At the top, the navigation bar shows 'Microsoft Azure' and a search bar. Below this, the breadcrumb trail indicates 'Home > Sarah1-RG > TechbysaraVM1'. The main content area is divided into two sections: a left-hand navigation pane and a right-hand details pane. The left pane lists various management options for the virtual machine, including Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Connect, Networking, Settings, Availability + scale, Security, Backup + disaster recovery, Operations, and Monitoring. The right pane, titled 'Essentials', provides key information about the VM: Resource group (Sarah1-RG), Status (Running), Location (East US (Zone 1)), Subscription (ABCOFCLOUD), Subscription ID (671c3453-e9f8-4939-aed9-...), Availability zone (1), Operating system (Windows (Windows Server ...)), Size (Standard B2ms (2 vcpus, 8 ...)), Public IP address (redacted), Virtual network/subnet (TechbysaraVNET/Project-Su...), DNS name (Not configured), Health state (-), and Time created (16/07/2024, 21:57 UTC). A 'JSON View' link is visible in the top right corner of the Essentials section.

Property	Value
Resource group	Sarah1-RG
Status	Running
Location	East US (Zone 1)
Subscription	ABCOFCLOUD
Subscription ID	671c3453-e9f8-4939-aed9-...
Availability zone	1
Operating system	Windows (Windows Server ...)
Size	Standard B2ms (2 vcpus, 8 ...)
Public IP address	[Redacted]
Virtual network/subnet	TechbysaraVNET/Project-Su...
DNS name	Not configured
Health state	-
Time created	16/07/2024, 21:57 UTC

TechbysaraVM1 successfully created.

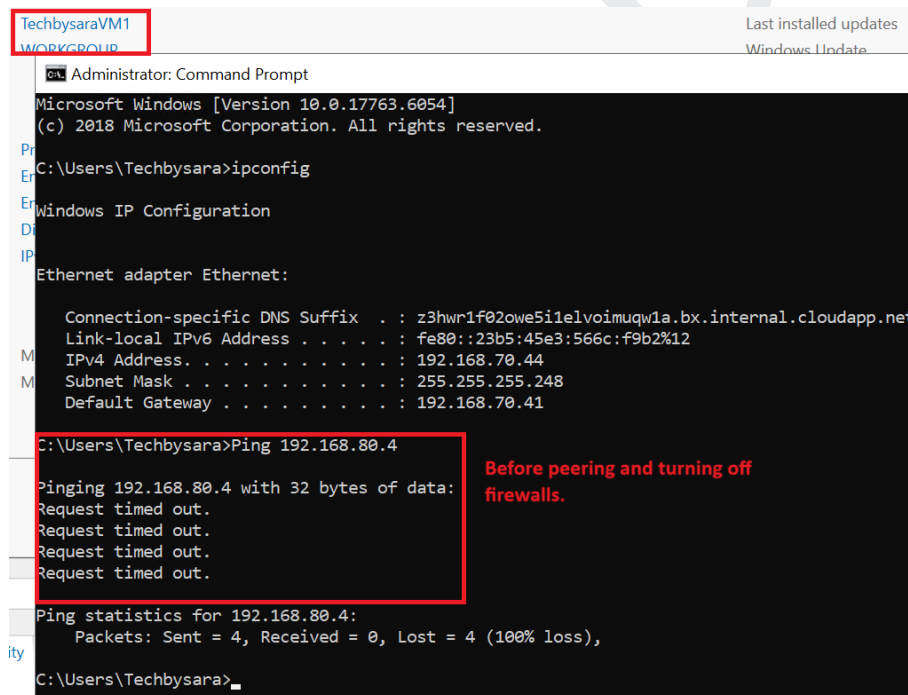
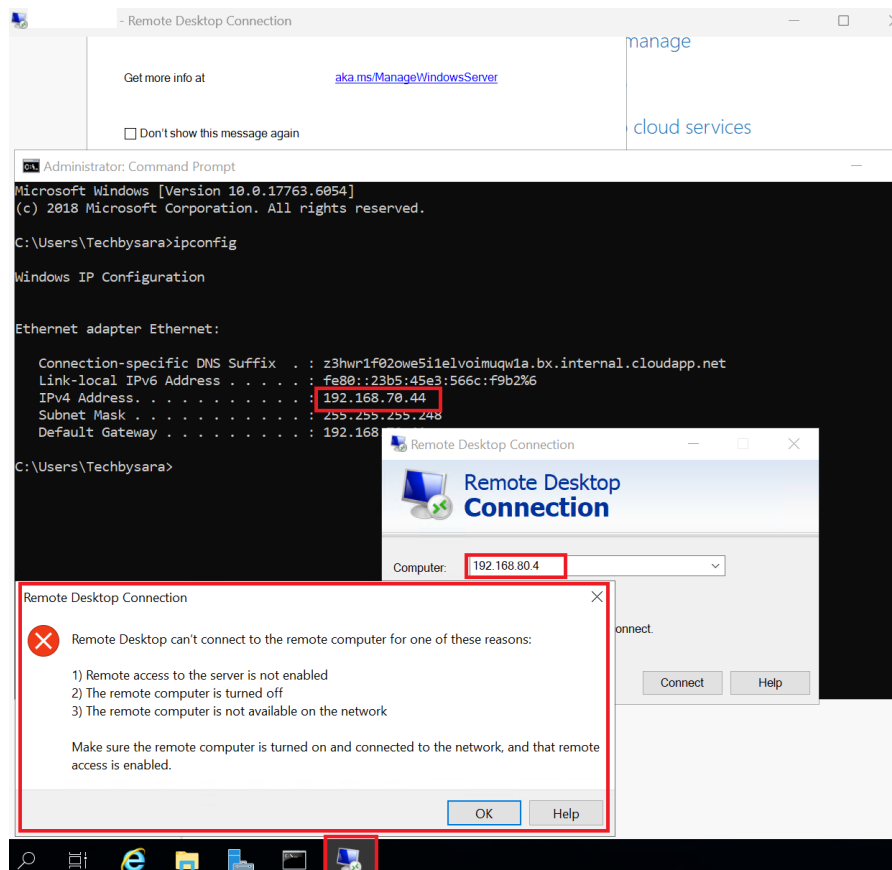


TechbysaraVM2 successfully created.

Step 3: Test Traffic Before Peering Between the Two VMs

We are going to connect to TechbysaraVM1 on our Remote desktop, using the public IP of TechbysaraVM1. Then we will attempt to RDP to TechbysaraVM2 from TechbysaraVM1 using the private IP of TechbysaraVM2. Also, we will attempt to establish communication through pinging.

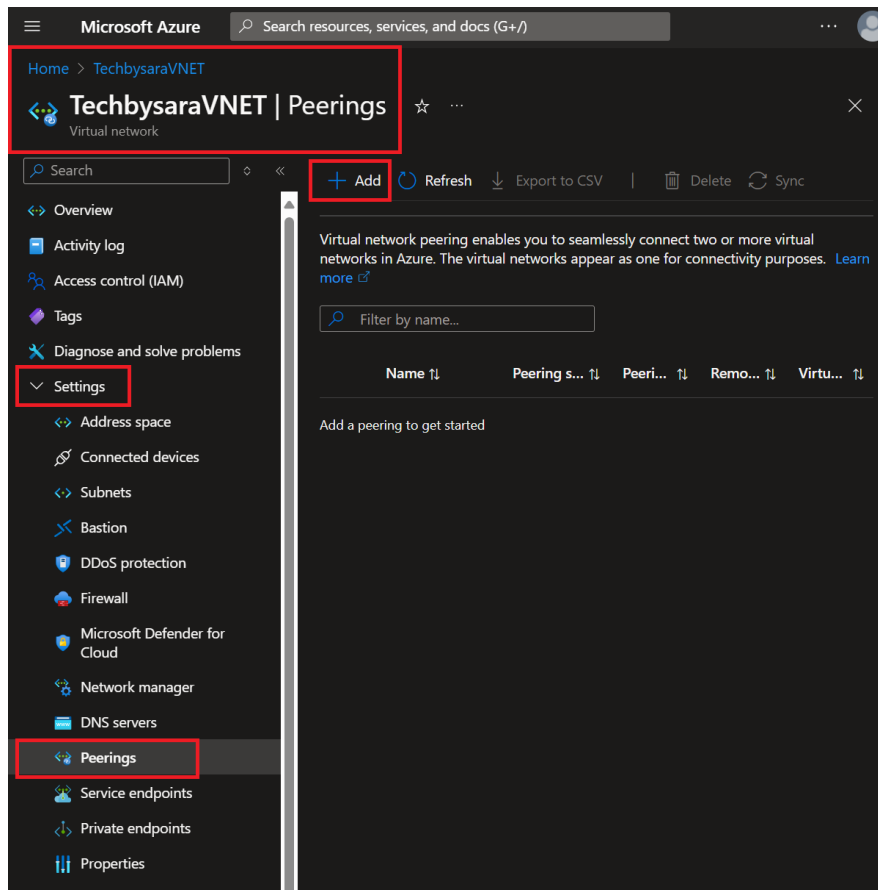
1. From your Resource group > **Sarah1-RG**.
2. Select > **TechbysaraVM1**.
3. On The **Overview page** > select the **Connect button** > Copy your **Public IP address**.
4. In your personal device > search for **Remote Desktop Connection app**.
5. Paste your copied **Public IP address**.



Remote access was denied because both virtual machines are on a different network. Also communication timed out while pinging.

Step 4: Create a Peering

1. In the Azure portal search box, enter your resource group name > (e.g., **Sarah1-RG**).
2. Virtual networks > select the **virtual network** you want to create a peering with.
3. In the **TechbysaraVNET** page > click on **settings** > scroll down > select **Peerings**.



Add peering > Remote virtual network summary:

4. **Peering link name:** The name of the peering from the local virtual network. The name must be unique within the virtual network > (e.g., **Techbysara-VNET-to-TechbysaraVNET2**).
5. **Virtual network deployment model:** Select the deployment model the virtual network you want to peer with was deployed through > (e.g., Resource manager).
6. **I know my resource ID:** Select this option if you do not have read access to the virtual network or subscription you wish to peer with > (leave unchecked).
7. **Subscription:** Select the subscription of the virtual network you want to peer with (e.g., ABCOFCLOUD).
8. **Virtual Network:** Select the remote virtual network > (e.g., TechbysaraVNET2).

Remote virtual network peering settings:

9. **Allow 'TechbysaraVNET2' to access 'TechbysaraVNET':** Select this option to allow traffic from 'TechbysaraVNET2' to 'TechbysaraVNET'. This setting enables, e.g.,

communication between hub and spoke in hub-spoke network topology and allows a VM in 'TechbysaraVNET2' to communicate with a VM in 'TechbysaraVNET'.

10. **Allow 'TechbysaraVNET2' to receive forwarded traffic from 'TechbysaraVNET':**

Enabling this option will allow 'TechbysaraVNET2' to receive traffic from virtual networks peered to 'TechbysaraVNET'.

Local virtual network summary:

11. Peering link name > (e.g., **TechbysaraVNET2-to-TechbysaraVNET**).

12. Repeat steps 9 and 10.

13. Select **Add**.

Microsoft Azure Search resources, services, and docs (G+)

Home > Sarah1-RG > TechbysaraVNET | Peerings >

Add peering

TechbysaraVNET

Virtual network peering enables you to seamlessly connect two or more virtual networks in Azure. This will allow resources in either virtual network to directly connect and communicate with resources in the peered virtual network.

Remote virtual network summary

Peering link name * TechbysaraVNET2-to-TechbysaraVNET2 4

Virtual network deployment model ⓘ
☒ Resource manager 5
☐ Classic

I know my resource ID ⓘ ☐ 6

Subscription * ABCOFCLOUD 7

Virtual network * TechbysaraVNET2 (Sarah1-RG) 8

Remote virtual network peering settings

Allow 'TechbysaraVNET2' to access 'TechbysaraVNET' ⓘ ☒ 9

Allow 'TechbysaraVNET2' to receive forwarded traffic from 'TechbysaraVNET' ⓘ ☒ 10

Allow gateway or route server in 'TechbysaraVNET2' to forward traffic to 'TechbysaraVNET' ⓘ ☐

Enable 'TechbysaraVNET2' to use 'TechbysaraVNET's' remote gateway or route server ⓘ ☐

Local virtual network summary

Peering link name * TechbysaraVNET2-to-TechbysaraVNET 11

Local virtual network peering settings

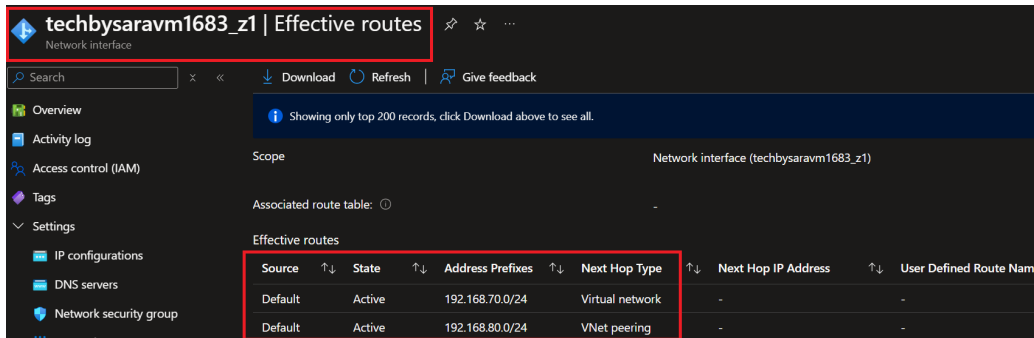
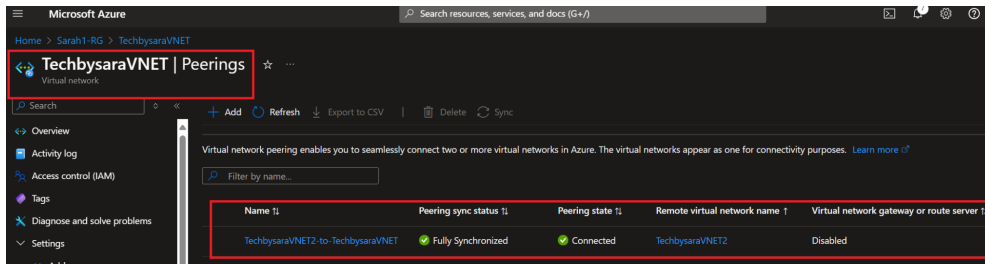
Allow 'TechbysaraVNET' to access 'TechbysaraVNET2' ⓘ ☒ 12

Allow 'TechbysaraVNET' to receive forwarded traffic from 'TechbysaraVNET2' ⓘ ☒ 12

Allow gateway or route server in 'TechbysaraVNET' to forward traffic to 'TechbysaraVNET2' ⓘ ☐

Enable 'TechbysaraVNET' to use 'TechbysaraVNET2's' remote gateway or route server ⓘ ☐

Add 13 Cancel

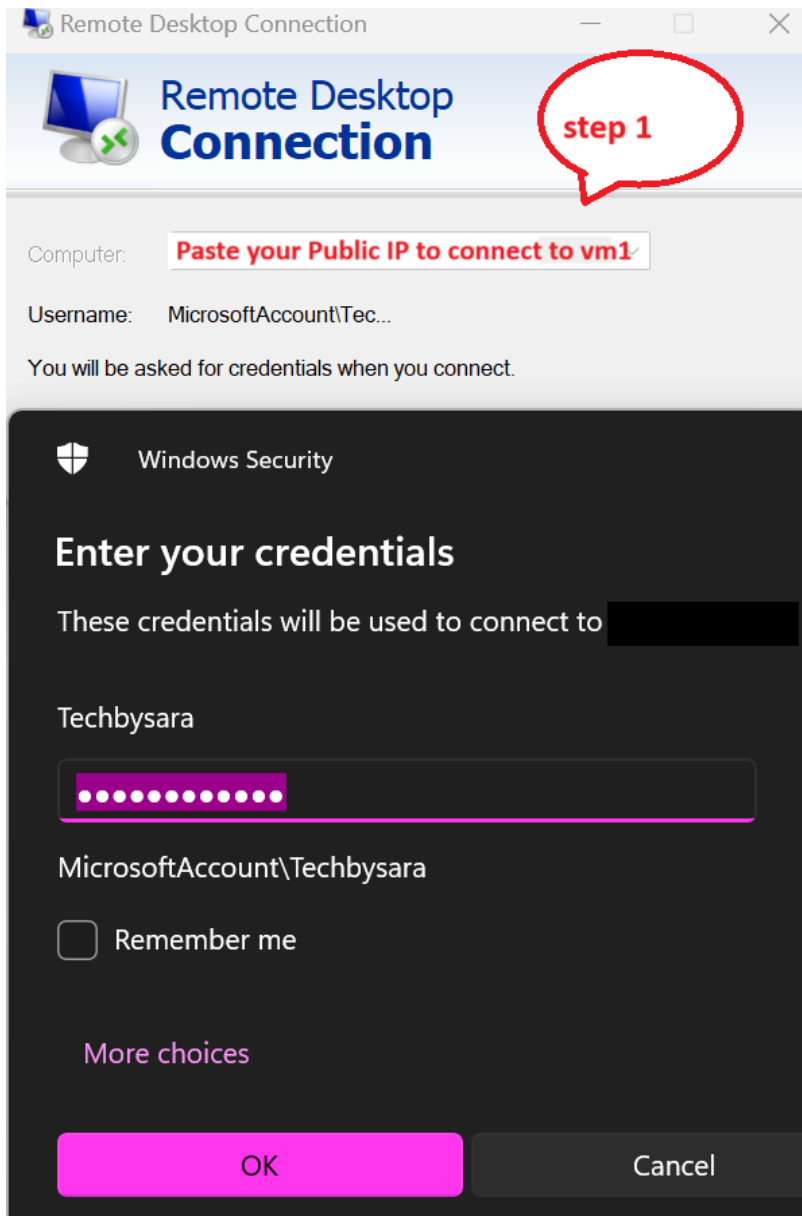


Peering successfully established.

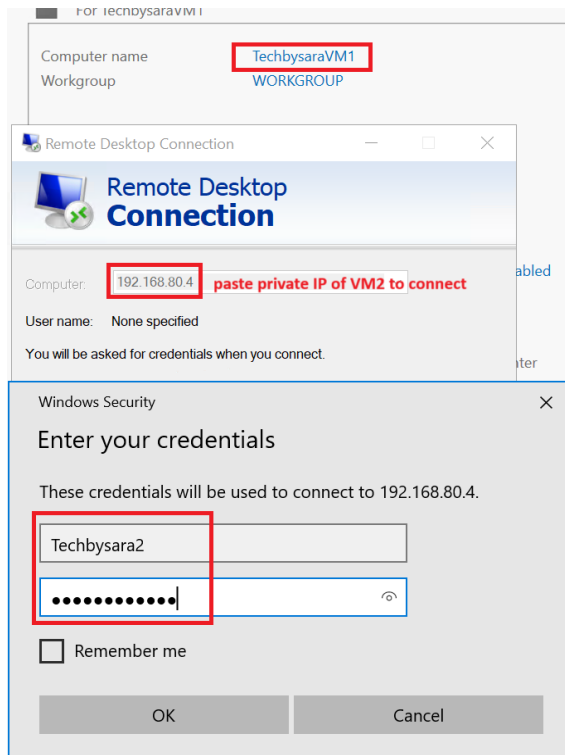
Step 5: Test Traffic After Peering and ping the VMs

Warning: Turning off firewalls increases malware infection risk. Therefore, it is not recommended in production environments. On that note, steps on how to turn off firewalls will not be added in this article.

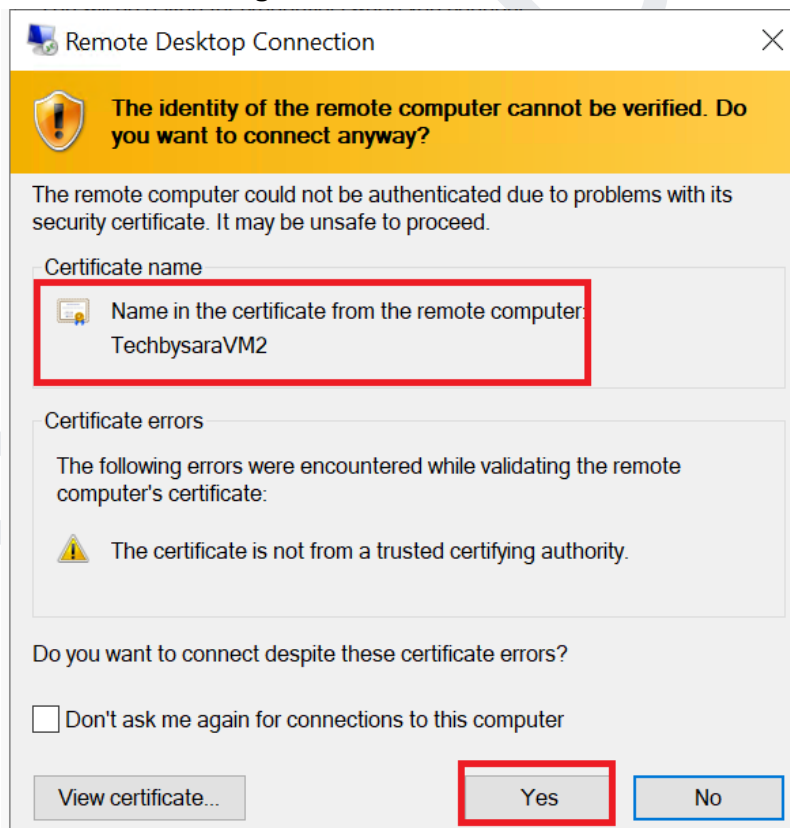
1. Connect to TechbysaraVM1 using Public IP.

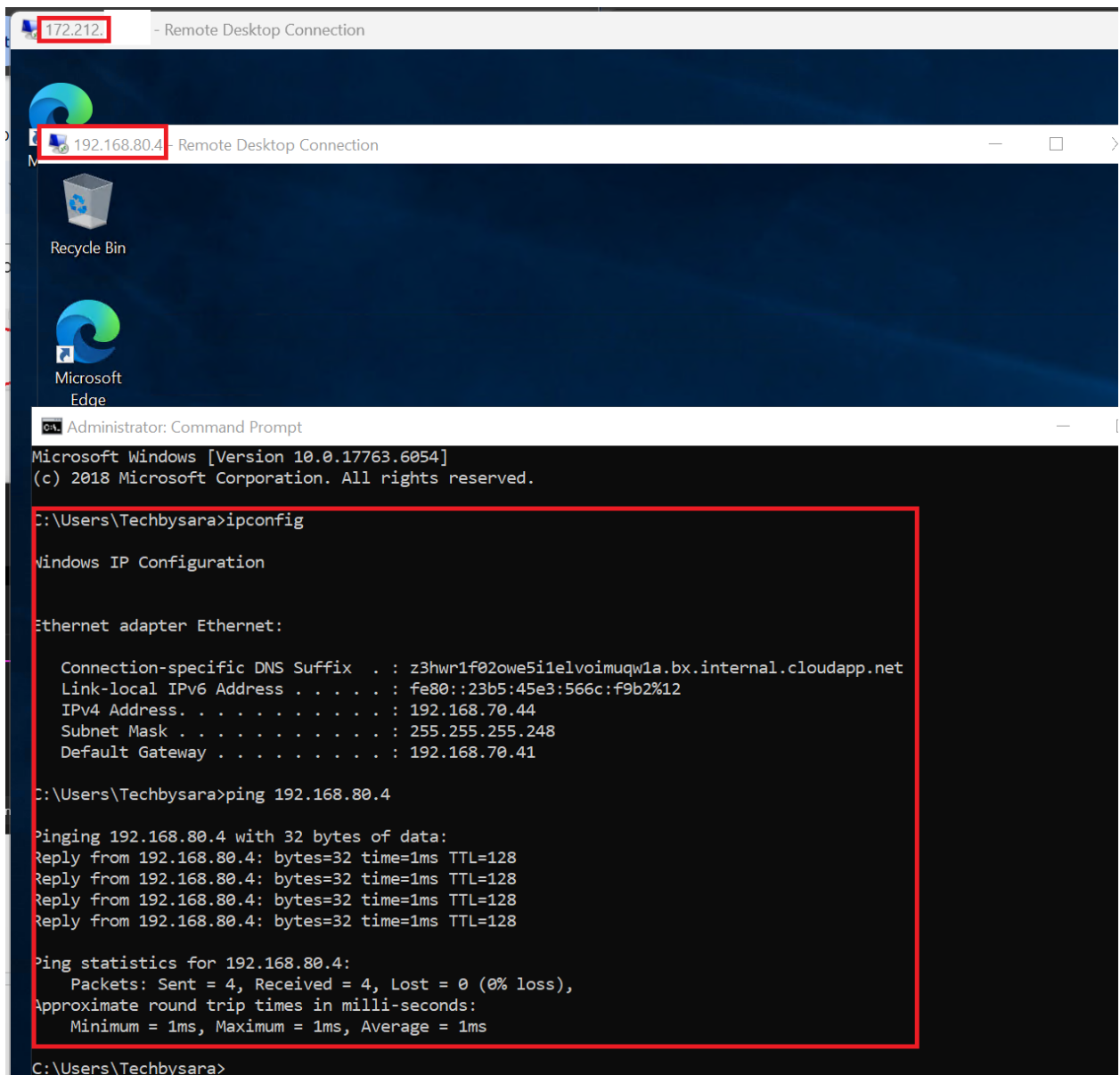


2. From TechbysaraVM1 > RDP to TechbysaraVM2 using private IP



You may receive a certificate warning during the connection process. If you receive the warning, select Yes or Continue to continue the connection.





Peering and communication between TechbysaraVM1 and TechbysaraVM2 successfully established.

THANK YOU!!