**Introduction**

**What is an Azure Virtual Network?**

* An Azure Virtual Network (VNet) is a representation of a network in the Azure Cloud. It allows users to create their network in Azure and control how resources communicate with each other.
* A VNet can be used to create a secure and isolated environment for Azure resources, such as virtual machines, web apps, and databases.
* It can also be used to connect on-premises networks to Azure, allowing for hybrid scenarios and extending the organization's existing network to the cloud.
* VNets can be configured with security features like network security groups and Azure firewalls to control inbound and outbound traffic.
* VNets can be connected to other VNets, allowing for the creation of complex network topologies and secure communication between resources in different VNets.

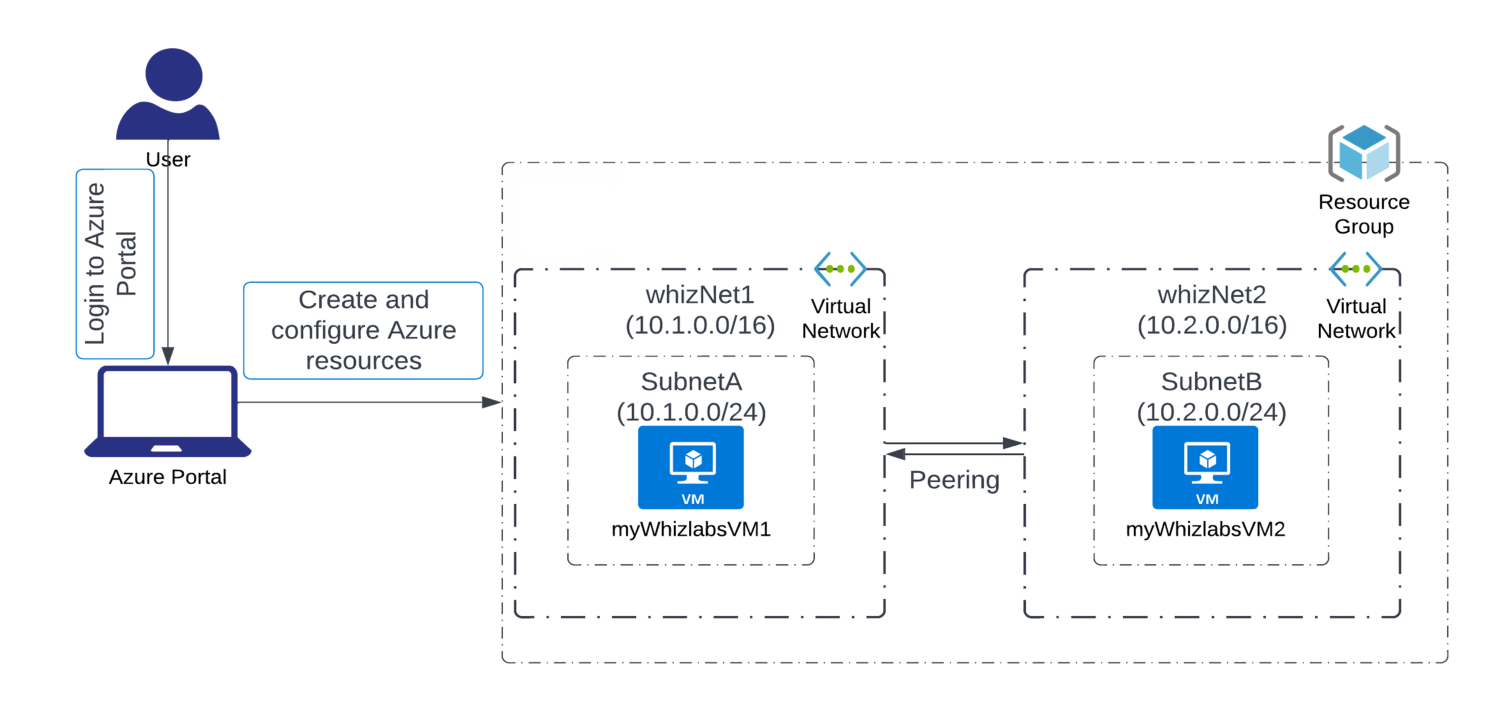
**What is VNet Peering in Azure?**

* Azure VNet peering is a feature that allows for direct communication between two Azure Virtual Networks (VNets) in the same Azure region.
* VNet peering creates a high-speed, low-latency connection between the peered VNets, allowing resources in one VNet to communicate with resources in another VNet as if they were in the same network.
* VNet peering can be used to connect VNets within the same Azure subscription or across different subscriptions.
* VNet peering supports both IPv4 and IPv6 addresses, and also allows for transitive routing, meaning that resources in one VNet can communicate with resources in another VNet that is peered with the first VNet.
* VNet peering can be used to create a hub-and-spoke network topology, where a central hub VNet has peered with multiple spoke VNets.

**What is Azure Virtual Machine?**

* Azure Virtual Machine (VM) is a cloud-based service provided by Microsoft Azure that enables users to deploy and manage virtual machines in a scalable and flexible manner.
* With Azure VMs, users have the freedom to choose from a wide range of virtual machine sizes and configurations, allowing them to meet specific workload requirements and optimize resource allocation.
* Azure VMs support multiple operating systems, including Windows and various Linux distributions, providing users with the flexibility to run their preferred applications and workloads in a familiar environment.
* Azure VMs integrate seamlessly with other Azure services, such as Azure Virtual Network and Azure Storage, enabling users to build comprehensive and interconnected cloud solutions that leverage the full capabilities of the Azure platform.

**Architecture Diagram**

****

**Task Details**

1. Sign in to Azure Portal
2. Create Virtual Networks
3. Peer the Virtual Networks
4. Create Virtual Machines
5. Establish Communication between Virtual Machines
6. Validation test
7. Delete the resources

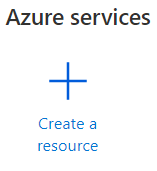
**Lab Steps**

**Task 1: Sign in to Azure Portal**

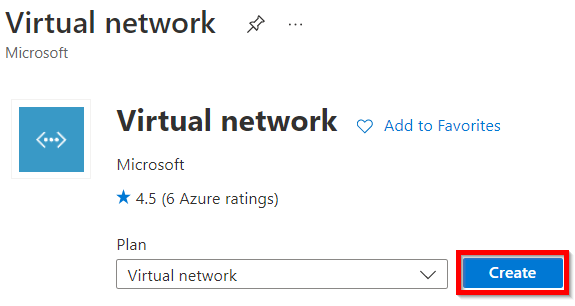
**Task 2: Create Virtual Networks**

In this lab, you will create virtual networks in Azure, enabling you to build isolated and secure network environments for your applications and resources,   
  
with the ability to configure network settings and establish connectivity.

Click on **Create a Resource** button



1. In the search box, enter **Virtual Network**



Select **Create**and enter the following values in the **Basics** tab.

Resource group: Select **rg\_westeurope\_XXXXX**  
Instance details:

Virtual Network Name: Enter **whizNet1**

Region: Select **West Europe**

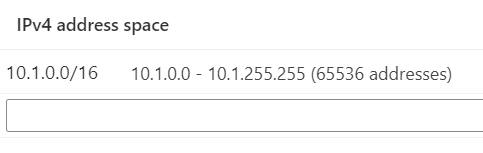


Click on the **Next: IP Addresses >**button and enter or select the following details:

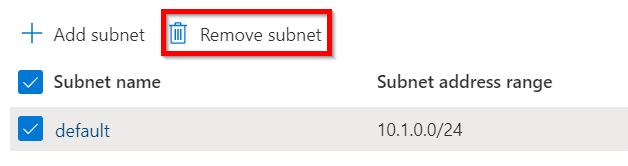
IPv4 address space: Enter **10.1.0.0/16**

Click on the **Next: IP Addresses >**button and enter or select the following details:

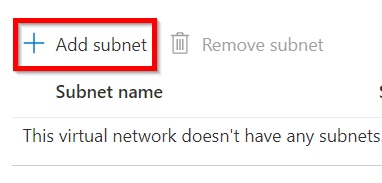
IPv4 address space: Enter **10.1.0.0/16**



Check the box on the left of the **default** subnet, and click on the **Remove Subnet**button.

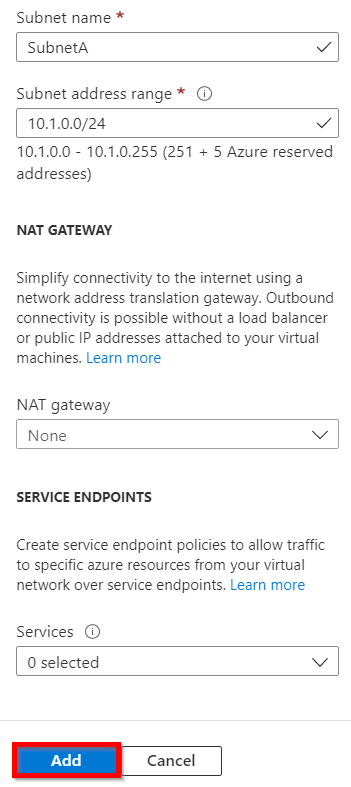


Now, Click on the **Add Subnet**button.

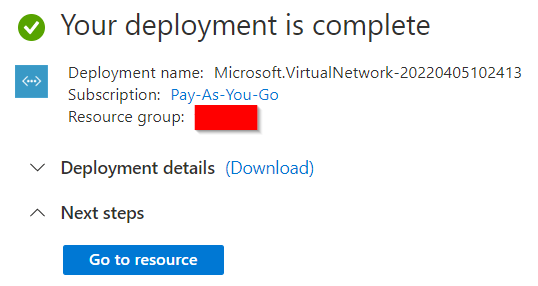


On the Add Subnet page, enter or select the following details and click on **Add.**

* Subnet Name: Enter **SubnetA**
* Subnet Address range: Enter **10.1.0.0/24**
* NAT gateway: Leave the defaults
* Service gateway: Leave the defaults



Select **Review + Create**and then select **Create.**



Repeat the above steps to create the second Virtual Network with the following details:

Basics tab

Resource group: Select **rg\_westeurope\_XXXXX**

Instance details:

Virtual Network Name: Enter **whizNet2**

Region: Select **West Europe**

IP Addresses tab

IPv4 address space: Enter **10.2.0.0/16**

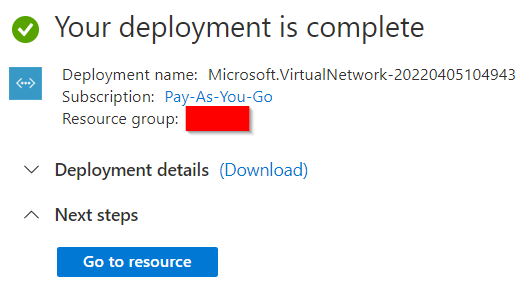
Subnet details:

Subnet Name: Enter **SubnetB**

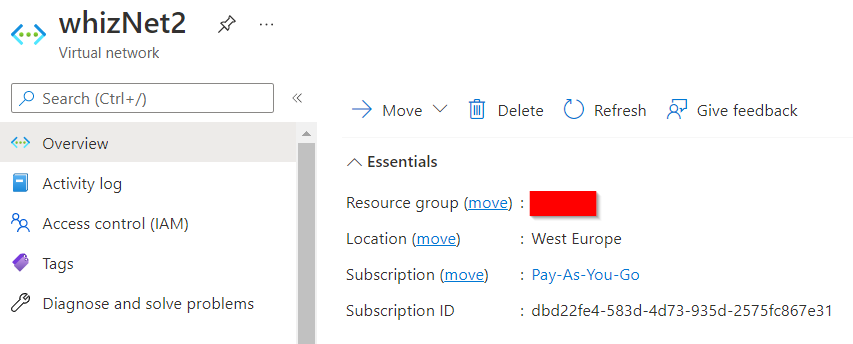
Subnet Address range: Enter **10.2.0.0/24**

NAT gateway: Leave the defaults

Service gateway: Leave the defaults



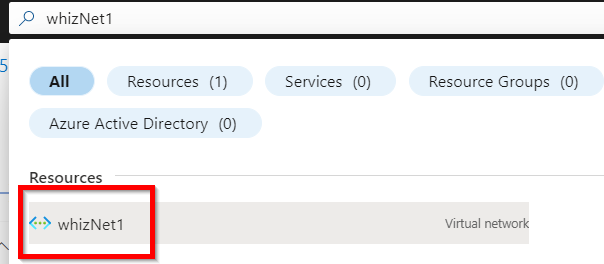
Click on **Go To Resources.**



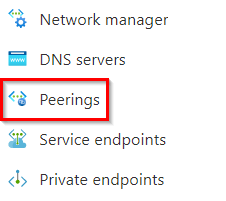
**Task 3: Peer the Virtual Networks**

You will learn how to establish network peering between virtual networks in Azure, enabling secure communication and resource sharing between the interconnected networks, and providing a seamless and scalable network architecture for your applications and services.

1. In the search box at the top of Azure Portal, search for **whizNet1** and select it from the list.



1. From the left menu, scroll down under the **Settings** section and select **Peerings.**



Click on **Add**button and enter or select the following details and click on **Add.**

Remote virtual Virtual Network summary:

Peering Link Name: Enter **whizNet2-whizNet1**

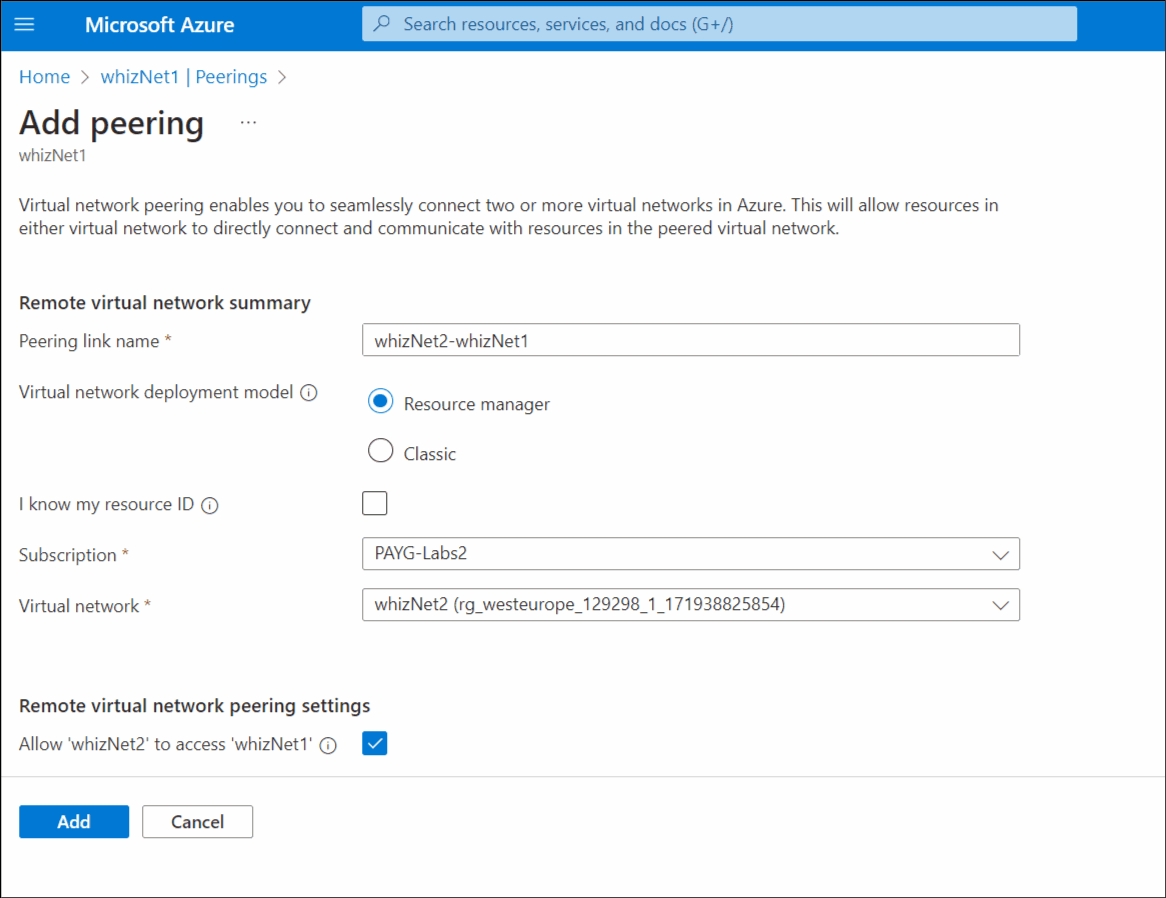
Local Virtual Network peering setting:

Peering Link Name: Enter **whizNet1-whizNet1**

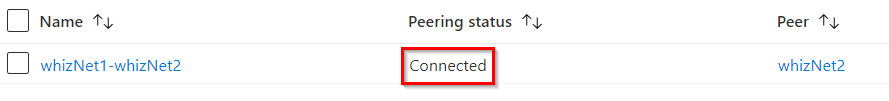
Subscription: Select your Subscription

Virtual Network: Select **whizNet2**

**NOTE:**After that step, it might take 5-10 min for the vnet to get displayed, try refreshing the page once or twice in 2-3mins.

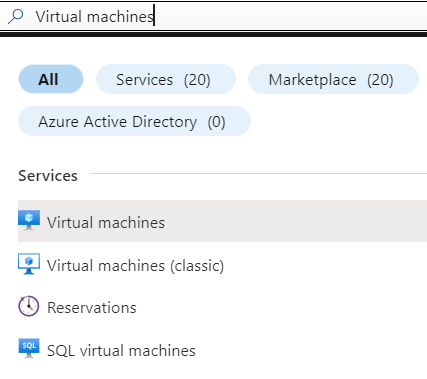


1. Both the VNets have now peered as the Peering Status shows **Connected.**

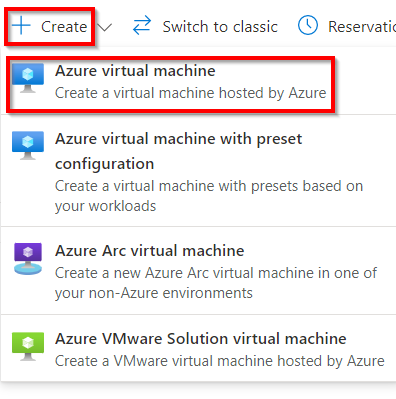


**Task 4: Create Virtual Machines**

1. In the search box at the top of Azure Portal, search for **Virtual Machines** and select it from the list.

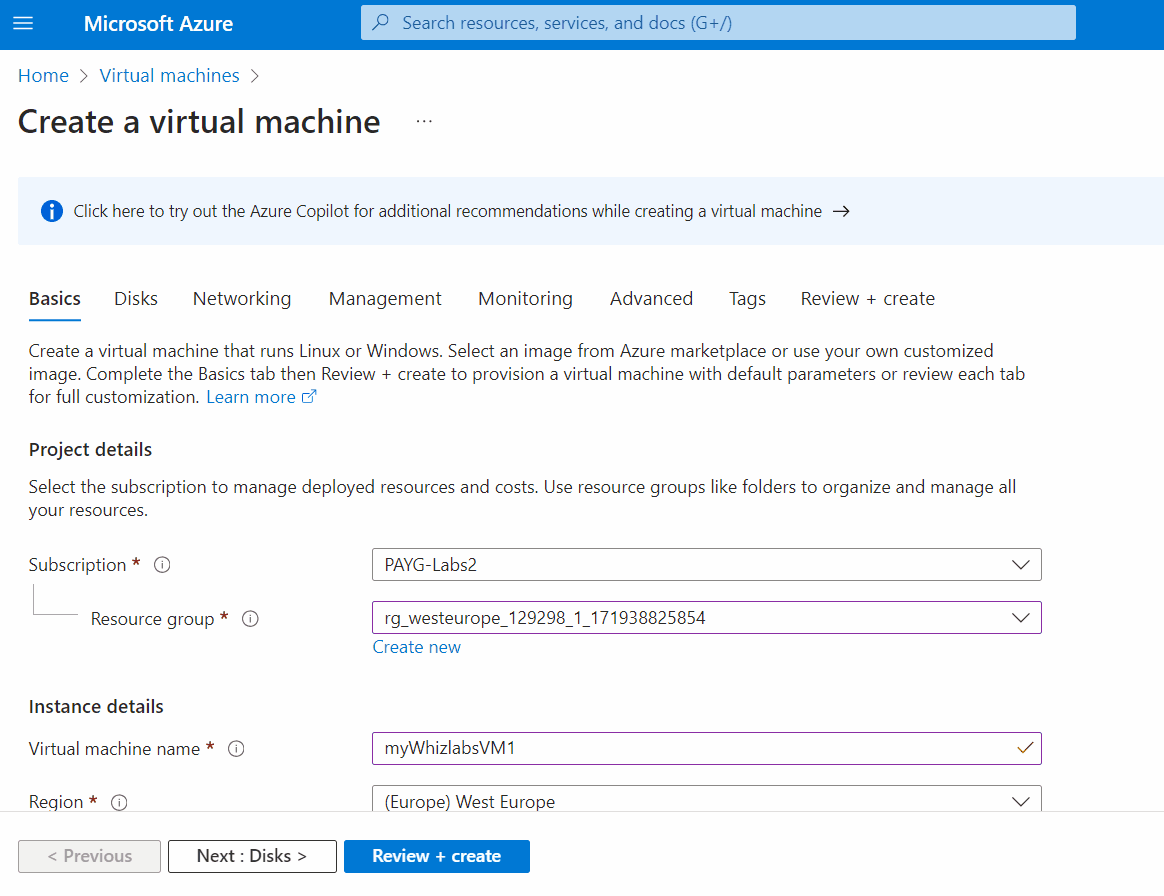


1. Click on **Create**button and select **Azure Virtual Machine.**



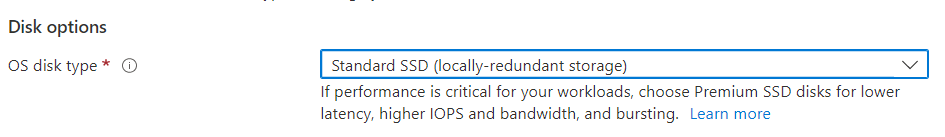
1. On the **Basics** tab, enter or select the following details:

* Resource group: Select **rg\_westeurope\_XXXXX**
* Instance details :
* Virtual Machine Name: Enter **myWhizlabsVM1**
* Region: Select **West Europe**
* Availabilty zone: Select **No infrastructure redundancy required**
* Security type: Select **Standard**
* Image: Select **Windows Server 2019 Datacenter - Gen2**
* Azure Spot instance: Leave the default unchecked.
* Size: Select **Standard\_B2s**
* Administrator Account :
* Username: Enter **vm1**
* Password: Enter a password
* Confirm password: Re-enter password
* Inbound Port rules :
* Public inbound ports: Select **Allow selected ports**
* Select inbound ports: Select **RDP (3389)**



Click on the **Next: Disk >**button and select the following details.

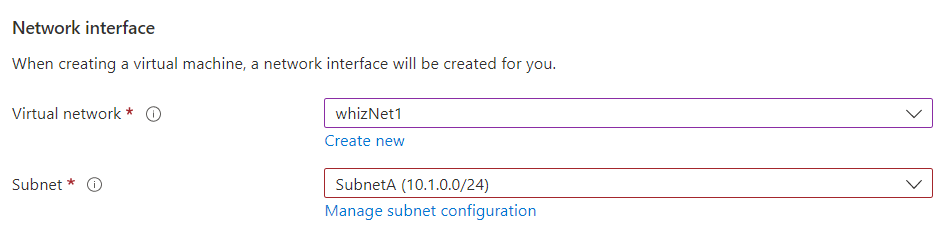
OS disk type: Select **Standard SSD**

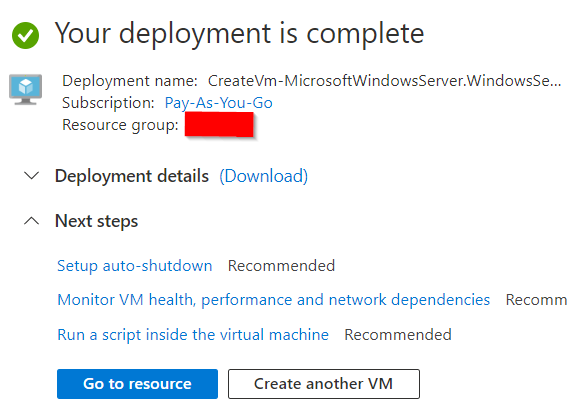


Click on the **Next: Networking >**button and enter or select the following details.

Network Interface:

* + Virtual Network: Select **whizNet1**
  + Subnet: Select **SubnetA**

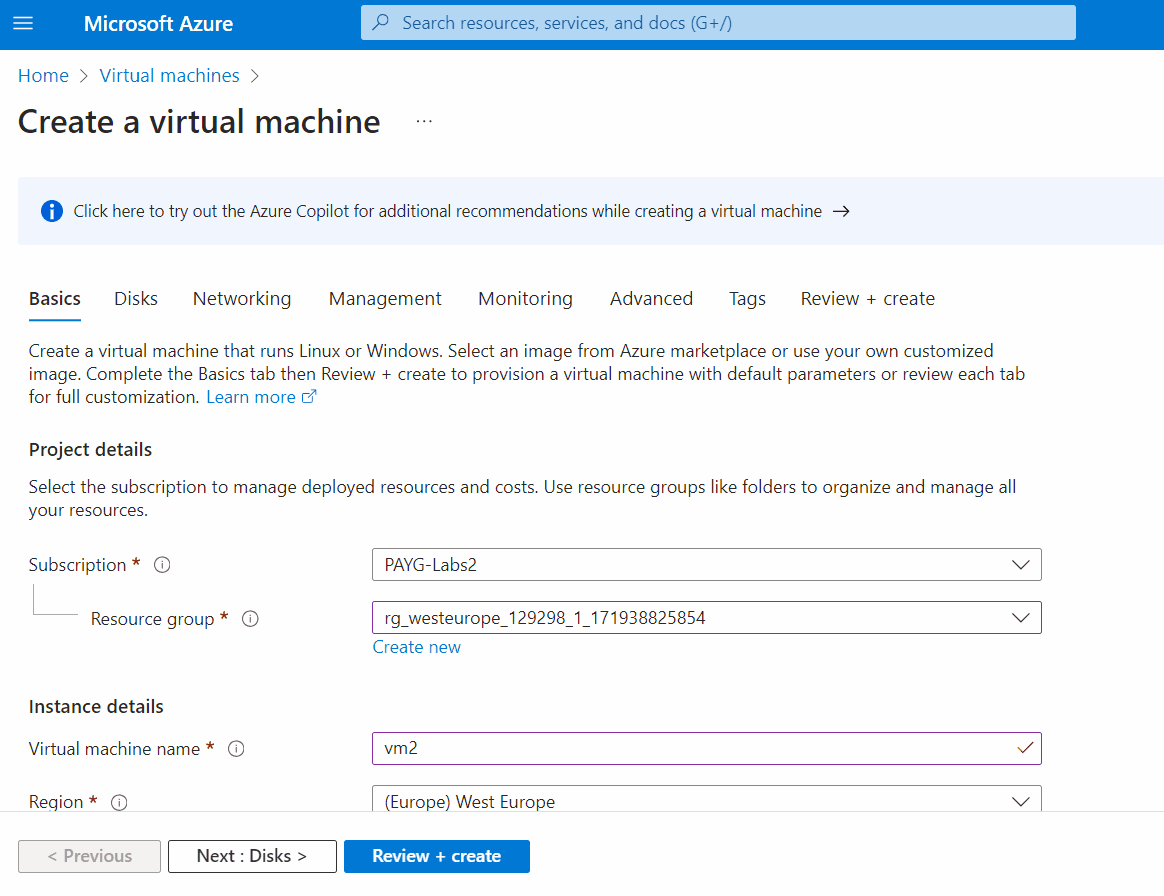




**Repeat steps 1 - 6 to deploy another VM and enter or select the following details.**

Basics tab:

* + Resource group: Select **rg\_westeurope\_XXXXX**
  + Instance details :
    - Virtual Machine Name: Enter **myWhizlabsVM2**
    - Region: Select **West Europe**
    - Availabilty zone: Select **No infrastructure redundancy required**.
    - Security type: Select **Standard**
    - Image: Select **Windows Server 2019 Datacenter - Gen2**
    - Azure Spot instance: Leave the default unchecked.
    - Size: Select **Standard\_B2s**
  + Administrator Account :
    - Username: Enter **vm2**
    - Password: Enter a password
    - Confirm password: Re-enter password
  + Inbound Port rules :
    - Public inbound ports: Select **None**



Disks tab:

OS disk type: Select **Standard SSD**

Networking tab:

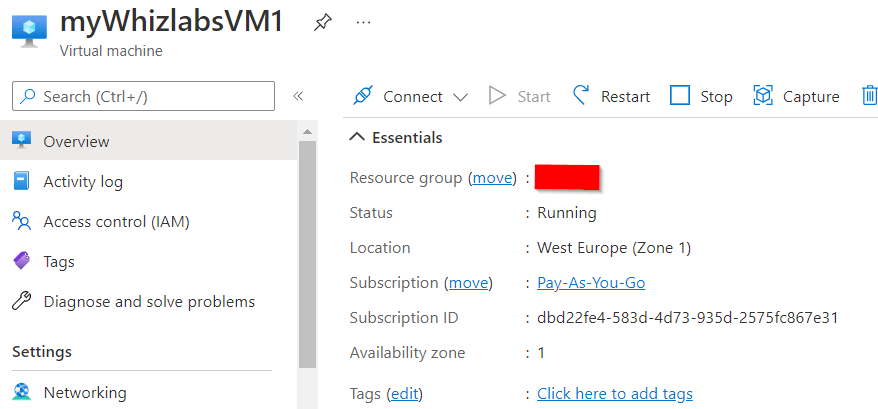
Network Interface:

Virtual Network: Select **whizNet2**

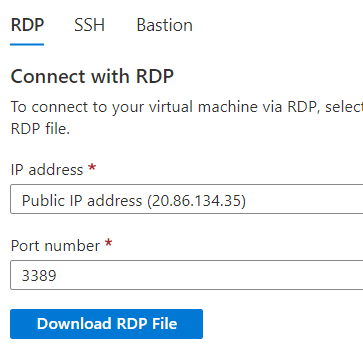
Subnet: Select **SubnetB**

**Task 5: Establish Communication between Virtual Machines**

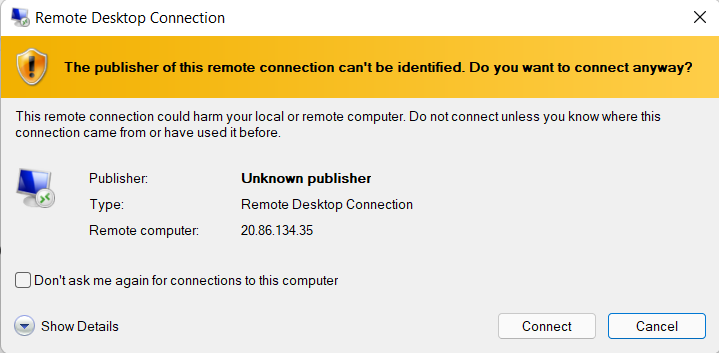
In the search box at the top of Azure Portal, search for **Virtual Machines** and select **myWhizlabsVM1**.



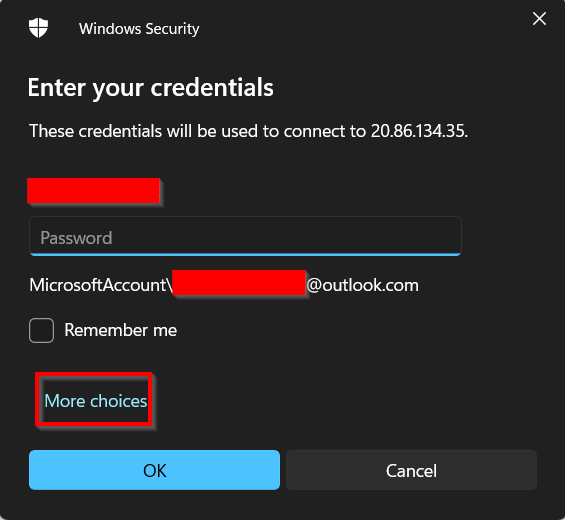
To connect to the virtual machine, select **Connect**and then select **RDP** from the dropdown. Select **Download RDP File** to download the remote desktop file.



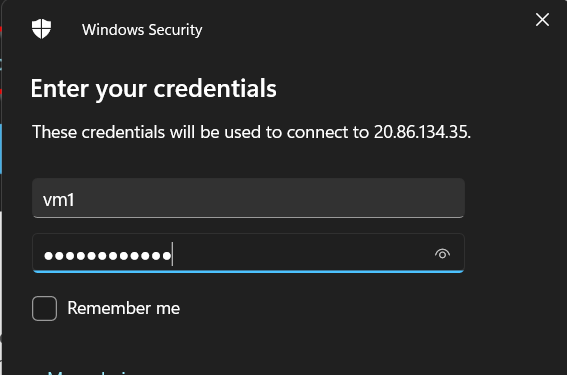
1. Open the download **RDP** file and select **Connect** on the displayed prompt.



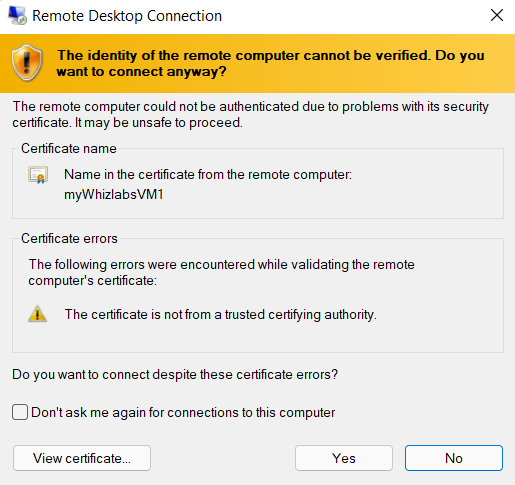
1. On the **Windows Security** prompt, click on **more choices.**



1. Click on **Use a different account** and enter the username and password you specified while creating the Virtual Machine and select **OK**.

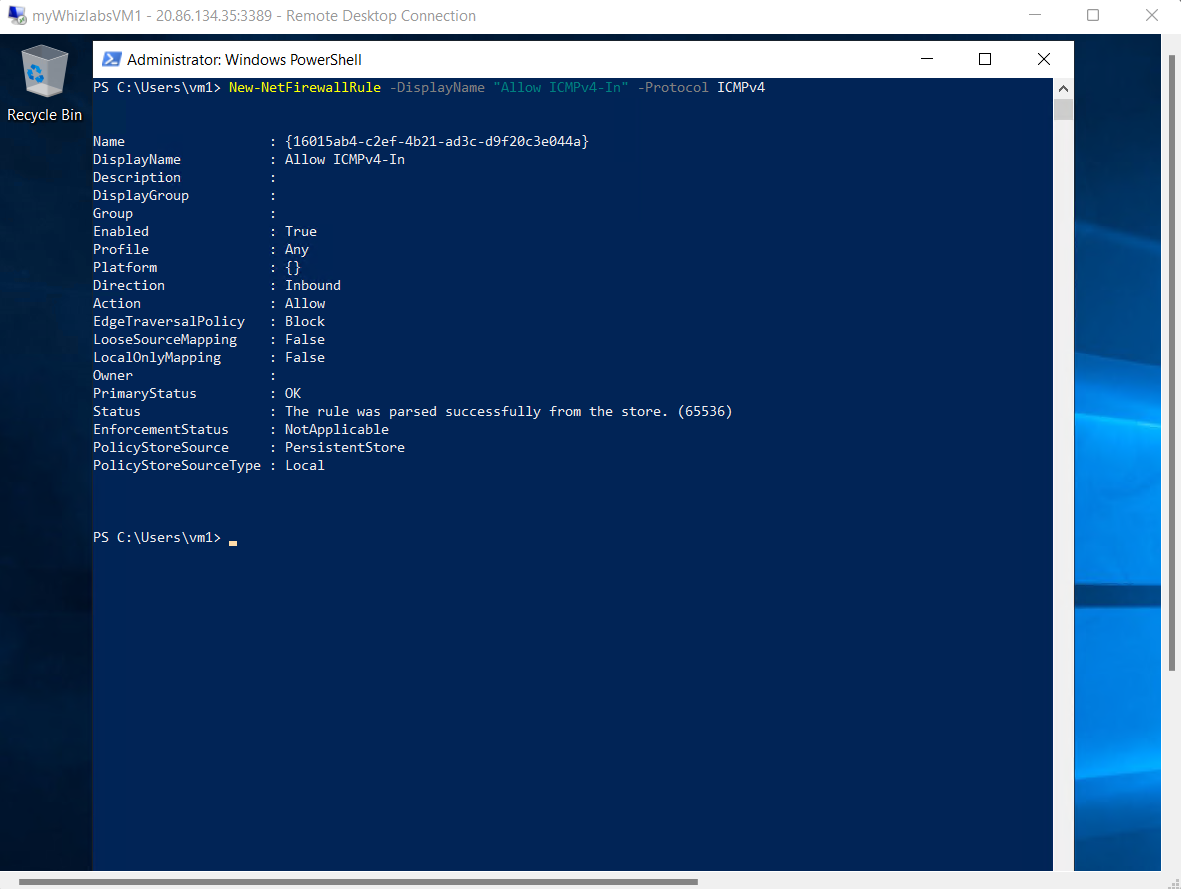


1. You may receive a certificate warning during the sign-in process. Select **Yes** to continue.



In the VM, open Powershell and enter the given command to enable Internet Control Message Protocol (ICMP), which is denied through the Windows firewall by default.

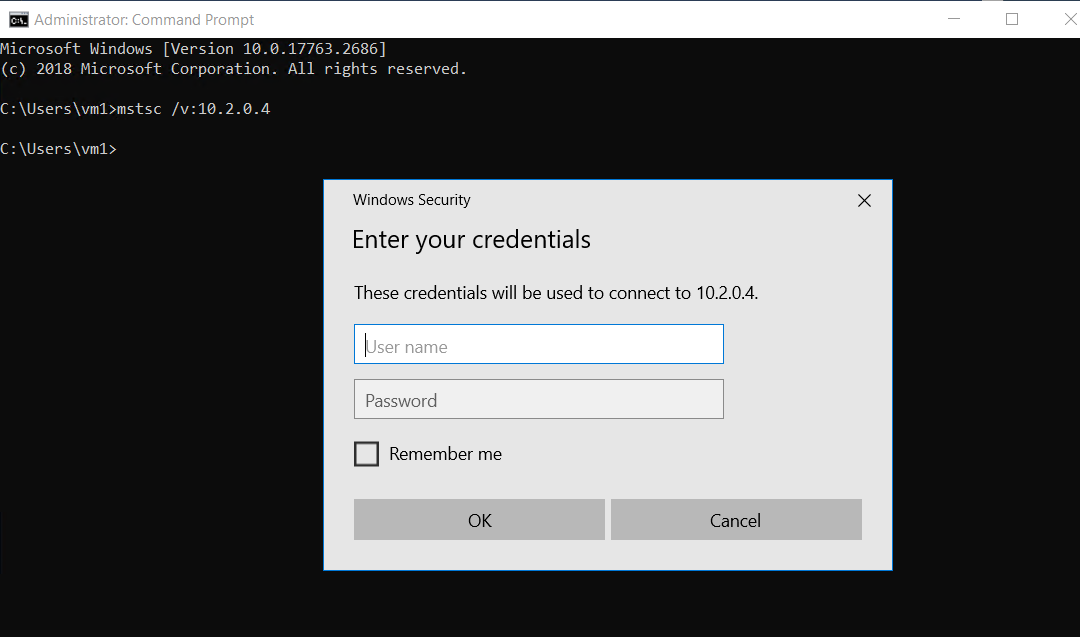
New-NetFirewallRule –DisplayName "Allow ICMPv4-In" –Protocol ICMPv4



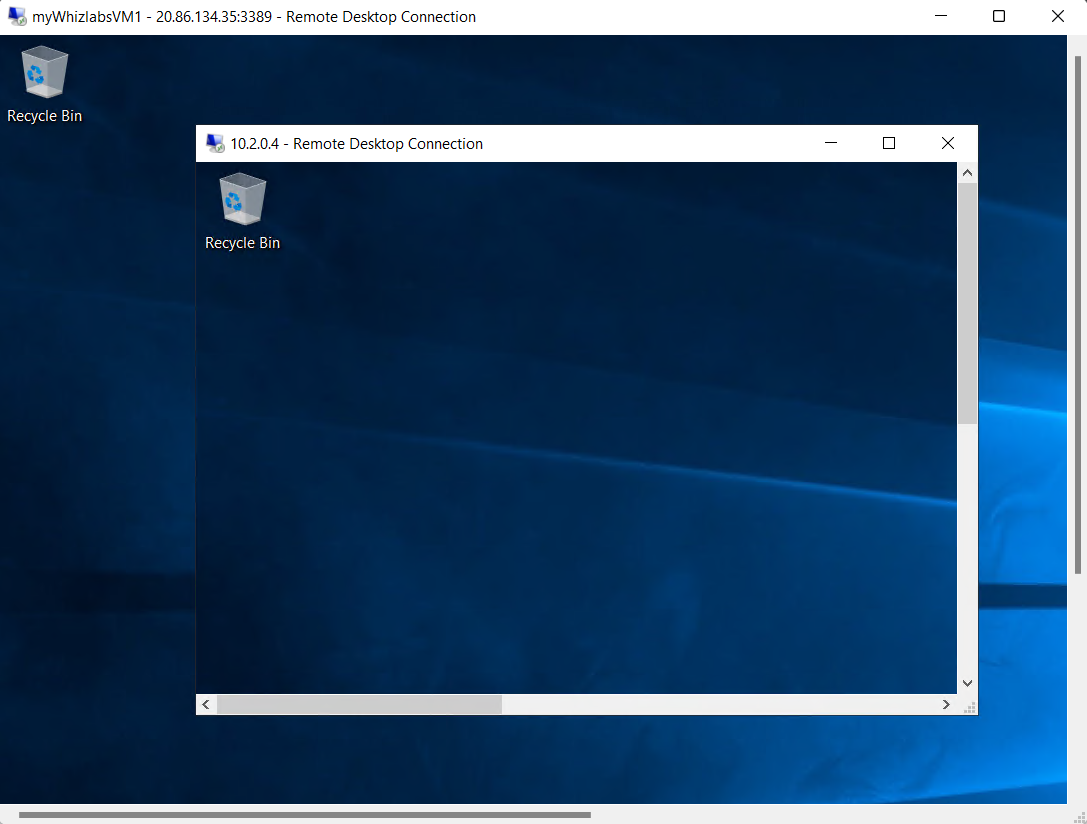
To connect to **myWhizlabsVM2,** enter the following command in the command prompt on **myWhizlabsVM1** and enter the credentials of **myWhizlabsVM2** on the login prompt.

mstsc /v:10.2.0.4

**Note:**10.2.0.4 is the private IP of WhizlabsVM2. If you fail to connect to your VM using this command, go to Azure Portal to check the private IP of your WhizlabsVM2



1. We have successfully logged into **myWhizlabsVM2** from **myWhizlabsVM1**.



### ****Do You Know?****

By implementing Virtual Network peering, you can securely connect virtual networks within same Azure region or across regions, enabling semless communication and expanding your network's capabilities.

**Completion and Conclusions**

1. You have successfully signed in to Azure Portal.
2. You have successfully created two Virtual Networks.
3. You have successfully peered at both Virtual Networks.
4. You have successfully deployed Virtual Machines on both Virtual Networks.
5. You have successfully established communication between the Virtual Machines.
6. You have successfully tested the validation.
7. You have successfully deleted the resources.