

# **R&D Document: CIDR Ranges of VNet, Subnet, VNet Peering and Its Types with Azure Use Case**

**Title:** CIDR Ranges of a VNet, Subnet, Vnet Peering and it's Types.

**Presented to:** Celebal Technologies

**Prepared by:** Tejasvi Avhad – Intern (Cloud Infra and Security)

## **1. Introduction**

In cloud computing, networks are created virtually. Microsoft Azure provides something called a Virtual Network (VNet) which acts like a personal network in the cloud. Inside a VNet, we can create smaller divisions called Subnets. These help in organizing and securing our resources better.

We also use CIDR ranges to assign IP addresses in the network. Additionally, we can connect two different VNets using a concept called VNet Peering.

This document explains all of this in a simple way and also includes the steps performed by me practically on the Azure Portal.

## **2. CIDR**

CIDR stands for Classless Inter-Domain Routing. It is a way to write IP address ranges. For example:

- 10.0.0.0/16 means we are giving a big range of IPs.
- 10.0.1.0/24 means a smaller range inside the bigger one.

CIDR helps Azure know how many IP addresses we want to reserve.

## **3. VNet (Virtual Network)**

A VNet is like a private network that we create inside Azure. It is completely isolated and helps us connect different services securely. For example, we can launch virtual machines inside it and make them talk to each other.

## **4. Subnet**

A Subnet is a smaller part of a VNet. We create subnets to group different types of resources. For example, we can keep Windows VMs in one subnet and Linux VMs in another.

Each subnet also has its own IP range, which comes from the VNet's CIDR range.

## **5. VNet Peering**

Sometimes, we need two different VNets to talk to each other. For this, we use VNet Peering. It connects two VNets so that resources inside them can communicate as if they are inside the same network.

There are two types:

- Same region peering – Both VNets are in the same Azure region.
- Global peering – The VNets are in different regions.

## **6. Prerequisites to Perform This Use Case**

Before we begin, we need:

- A valid Azure subscription
- Internet connection
- Basic knowledge of how to use the Azure portal

## **7. Use Case Overview**

Create two VNets. Inside the first VNet, create two subnets. Launch a Windows VM in the first subnet and a Linux VM in the second. Then, create a second VNet and connect it with the first using VNet peering. Finally, ensure that the VMs can ping each other.

## **8. Steps to Perform the Use Case (on Azure Portal)**

### **Step 1: Create a Resource Group**

- Go to Azure Portal
- Search and open Resource Groups
- Click on + Create
- Name the group ( MyResourceGroup)
- Select Region
- Click Create

The screenshot shows the Microsoft Azure Resource groups interface. The left sidebar lists 'MyResourceGroup' under 'Resource groups'. A message indicates a new version of the Browse experience is available. The main area displays the 'Overview' tab for 'MyResourceGroup', which contains sections for Activity log, Access control (IAM), Tags, Resource visualizer, Events, Settings, Cost Management, Monitoring, Automation, and Help. On the right, there's a table for managing resources, with columns for Name, Type, and Location. A search bar at the top allows filtering by resource type.

## Step 2: Create First VNet and Two Subnets

- Search and open Virtual Networks
- Click + Create
- Name it VNet1
- Set address space to 10.0.0.0/16
- Create two subnets:
  - Subnet1: 10.0.1.0/24 (for Windows VM)
  - Subnet2: 10.0.2.0/24 (for Linux VM)
- Click Create

The screenshot shows the Microsoft Azure Virtual Network Overview page for 'VNet1-1750861681046'. It displays deployment details: Deployment name: VNet1-1750861681046, Subscription: Azure for Students, Resource group: MyResourceGroup. The status is 'Your deployment is complete'. It includes sections for Deployment details and Next steps, with a 'Go to resource' button. The right side features promotional cards for Cost management, Microsoft Defender for Cloud, and Free Microsoft tutorials.

The screenshot shows the Microsoft Azure portal interface for managing subnets in a virtual network. The left sidebar has a 'Subnets' section selected under 'Virtual network'. The main area displays a table of subnets with columns for Name, IPv4, IPv6, Available IPs, Delegated to, Security group, and Route table. Subnet1 and Subnet2 are selected, while the default subnet is not.

Name	IPv4	IPv6	Available IPs	Delegated to	Security group	Route table
default	10.0.0.0/24	-	251	-	-	-
Subnet1	10.0.1.0/24	-	251	-	-	-
Subnet2	10.0.2.0/24	-	251	-	-	-

### Step 3: Create Windows and Linux Virtual Machines

- Go to Virtual Machines click + Create
- Choose Windows Server 2019 for the first VM
- Place it in Subnet1 of VNet1
- Choose Ubuntu Linux for the second VM
- Place it in Subnet2 of VNet1
- Set login credentials
- Allow required ports (RDP for Windows, SSH for Linux)

The screenshot shows the Microsoft Azure portal interface for a completed deployment named 'CreateVm-MicrosoftWindowsServer.WindowsServer-201-20250625202440'. The 'Overview' tab is selected. The deployment status is shown as complete. The 'Deployment details' table lists four resources: WindowsVM01, windowsvm01350, WindowsVM01-nsg, and WindowsVM01-ip, all in 'OK' status. A sidebar on the right provides links for Cost Management, Microsoft Defender for Cloud, and Free Microsoft tutorials.

**Properties**

**Virtual machine**

Computer name	LinuxVM01
Operating system	Linux (ubuntu 22.04)
VM generation	V2
VM architecture	x64
Agent status	Ready
Agent version	2.14.0.1
Hibernation	Disabled
Host group	-
Host	-

**Networking**

Public IP address	4.213.167.20 ( Network interface )
Public IP address (IPv6)	-
Private IP address	10.0.0.5
Private IP address (IPv6)	-
Virtual network/subnet	VNet1/default
DNS name	Configure

**Size**

Size	Standard D2s v3
------	-----------------

## Step 4: Allow Ping (ICMP) in Network Security Group

- Go to each VM's Networking settings
- Click on Network Security Group
- Add a rule to allow ICMP (Ping)

**Inbound security rules**

Priority	Name	Port	Source	Destination	Action
300	SSH	22	Any	Any	Allow
1000	Allow-Ping-ICMP	Any	Any	Any	Allow
65000	AllowVnetInBound	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalanc...	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Deny

## Step 5: Create Second VNet

- Go to Virtual Networks click on + Create
- Name it VNet2
- Set address space to 10.1.0.0/16
- Create one subnet inside it: 10.1.1.0/24
- Click Create

The screenshot shows the Microsoft Azure portal with the URL [https://portal.azure.com/#blade/Microsoft\\_Azure\\_VirtualNetworks/ResourceGroupDeploymentBlade/resourceId=%2Fsubscriptions%2F00000000-0000-0000-0000-000000000000%2FresourceGroups%2FMyResourceGroup%2Fproviders%2FMicrosoft.Network%2FvirtualNetworks%2FVNet2%2Fsubnets%2FSubnet2&resourceType=Microsoft.Network/virtualNetworks/subnets&resourceName=Subnet2&resourceGroup=MyResourceGroup&operationName=Deployment2-1750914373958&bladeType=ResourceGroupDeploymentBlade](#). The page title is "VNet2-1750914373958 | Overview". A deployment summary indicates "Deployment succeeded" for "Deployment 'VNet2-1750914373958' to resource group 'MyResourceGroup' was successful." Deployment details show the name as "VNet2-1750914373958", subscription as "Azure for Students", and resource group as "MyResourceGroup". Deployment time is 26/6/2025, 10:32:04 am. Correlation ID is bc0f9d2-fcac-48eb-9081-fd48... The "Deployment details" section includes a "Deployment details" link and a "Next steps" section with a "Go to resource" button. The right sidebar features links for "Cost management", "Microsoft Defender for Cloud", and "Free Microsoft tutorials".

The screenshot shows the Microsoft Azure portal with the URL [https://portal.azure.com/#blade/Microsoft\\_Azure\\_VirtualNetworks/SubnetBlade/resourceId=%2Fsubscriptions%2F00000000-0000-0000-0000-000000000000%2FresourceGroups%2FMyResourceGroup%2Fproviders%2FMicrosoft.Network%2FvirtualNetworks%2FVNet2%2Fsubnets%2FSubnet2&resourceType=Microsoft.Network/virtualNetworks/subnets&resourceName=Subnet2&resourceGroup=MyResourceGroup](#). The page title is "VNet2 | Subnets". It lists two subnets: "default" (IPv4: 10.1.0.0/24, IPv6: -, Available IPs: 251, Delegated to: -, Security group: -, Route table: -) and "Subnet2" (IPv4: 10.1.1.0/24, IPv6: -, Available IPs: 251, Delegated to: -, Security group: -, Route table: -). The left sidebar shows navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Settings, Address space, Connected devices, Subnets (selected), and Bastion.

## Step 6: Configure VNet Peering

From VNet1 to VNet2:

- Go to VNet1 > Peerings > + Add
- Give name: VNet1-To-VNet2
- Select VNet2 as the peer
- Enable "Allow VNet access"
- Click Add

From VNet2 to VNet1:

- Go to VNet2 > Peerings > + Add
- Give name: VNet2-To-VNet1
- Select VNet1 as the peer
- Enable "Allow VNet access"
- Click Add

The screenshot shows the Microsoft Azure portal interface for managing virtual network peerings. The top navigation bar includes 'Microsoft Azure', a search bar, and user information. Below the navigation is the title 'VNet1 | Peerings' and a sub-header 'Virtual network'. On the left, a sidebar lists various network components: DDoS protection, Firewall, Microsoft Defender for Cloud, Network manager, DNS servers, Peering (which is selected), Service endpoints, Private endpoints, Properties, Locks, and Monitoring. The main content area displays a table of peerings. The table has columns for Name, Peering sync status, Peering ID, Remote VNet, Virtual network, and Cross-tenant. One row is visible: 'VNet1-to-VNet2' with status 'Fully Synchronized', 'Connected' under Peering ID, 'VNet2' under Remote VNet, 'Disabled' under Virtual network, and 'No' under Cross-tenant. There are also buttons for 'Add', 'Refresh', 'Export to CSV', 'Delete', and 'Sync'.

Name	Peering sync status	Peering ID	Remote VNet	Virtual network	Cross-tenant
VNet1-to-VNet2	Fully Synchronized	VNet1-to-VNet2	VNet2	Disabled	No

The screenshot shows the Microsoft Azure portal interface. At the top, there's a navigation bar with 'Microsoft Azure', a search bar, and various icons like Copilot, Refresh, Export to CSV, Delete, Sync, and Help. The user is signed in as 'tejasviyahad@gmail.com'.

The main content area is titled 'VNet2 | Peerings'. On the left, a sidebar lists options: Bastion, DDoS protection, Firewall, Microsoft Defender for Cloud, Network manager, DNS servers, Peering (which is selected), Service endpoints, Private endpoints, and Properties. A note at the bottom of the sidebar says 'Add or remove favorites by pressing Ctrl+Shift+F'.

The main pane displays a table of peers. The columns are: Name, Peering sync status, Peering sync status, Remote IP, Virtual network, and Cross-tenant. One row is shown: 'VNet1-to-VNet2', 'Fully Synchronized', 'Connected', 'VNet1', 'Disabled', and 'No'.

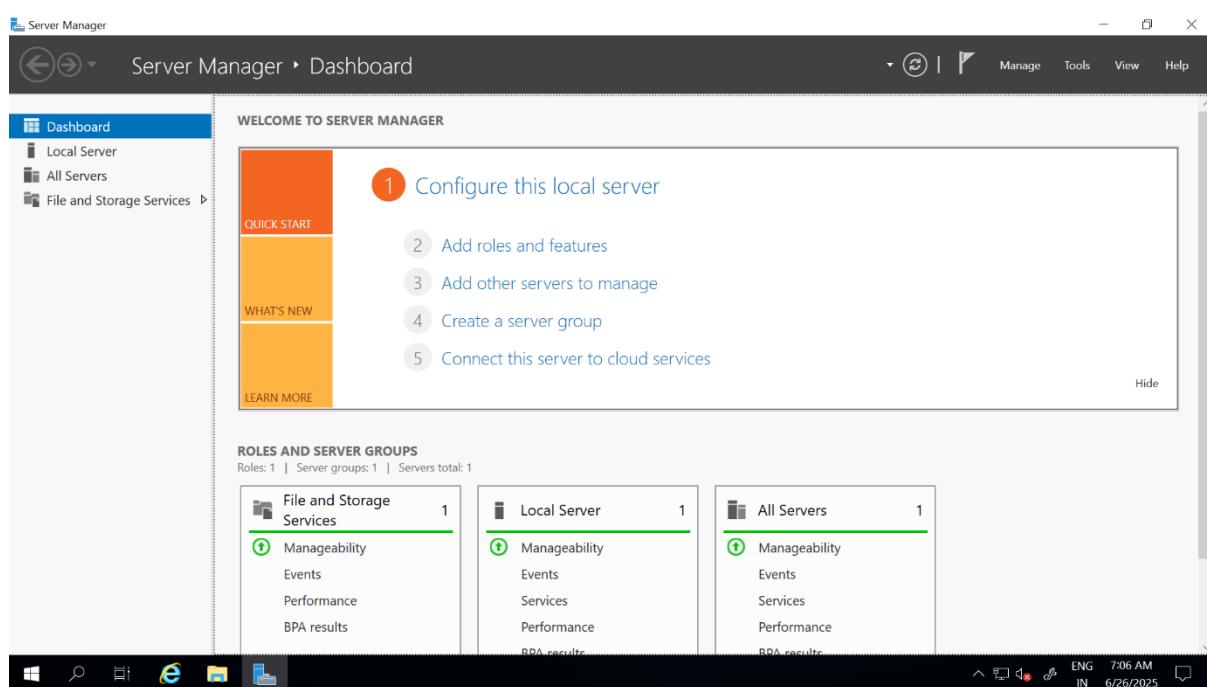
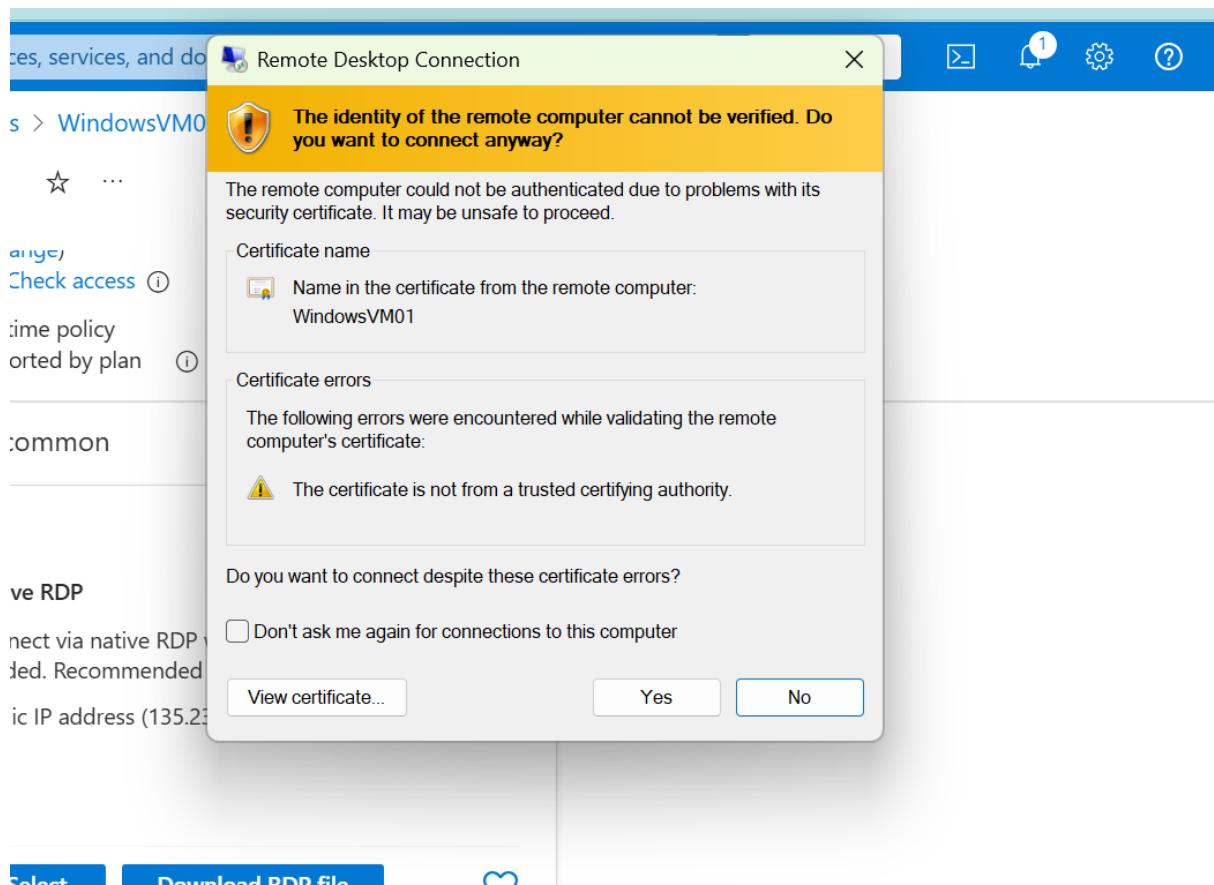
## Step 7: Ping Test Between VMs

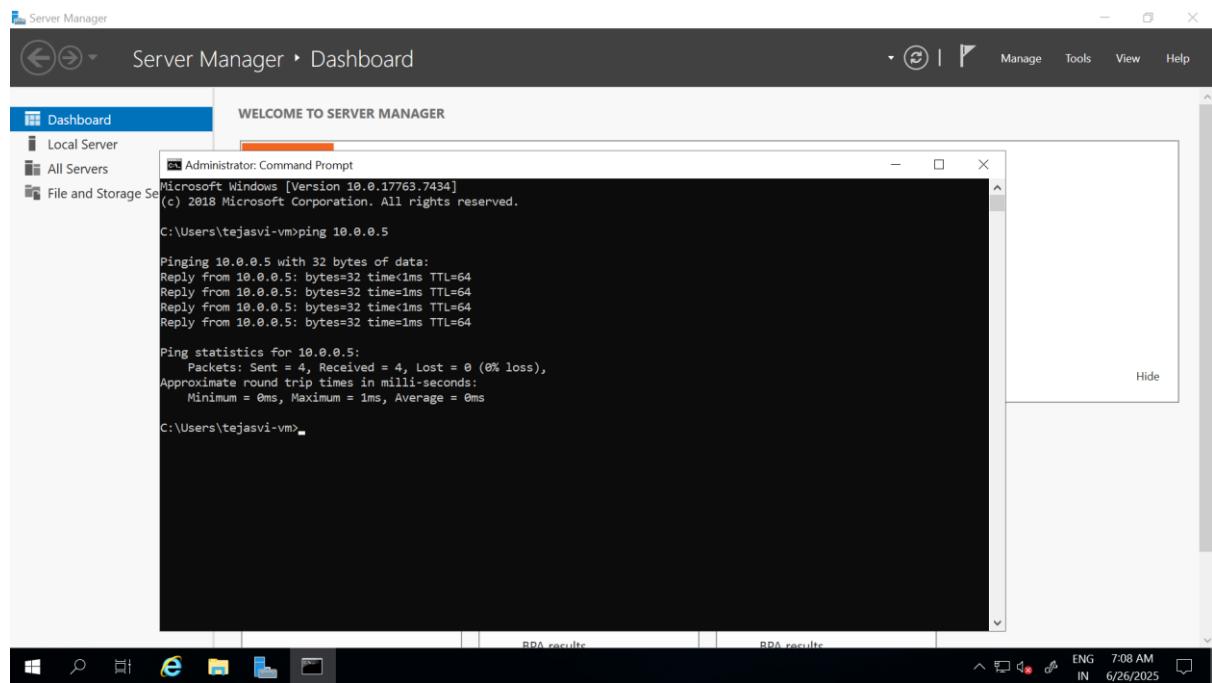
- From Windows VM, open Command Prompt:
  - Run: ping 10.0.0.5 (Linux VM private IP)
- Expected Output:
  - Replies with TTL=64, 0% loss

The screenshot shows the Microsoft Azure portal interface. At the top, there's a navigation bar with 'Microsoft Azure', a search bar, and various icons like Copilot, Refresh, Export to CSV, Delete, Sync, and Help. The user is signed in as 'tejasviyahad@gmail.com'.

The main content area is titled 'WindowsVM01 | Connect'. On the left, a sidebar lists options: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Connect (which is selected), Bastion, Windows Admin Center, and Networking. A note at the bottom of the sidebar says 'Add or remove favorites by pressing Ctrl+Shift+F'.

A modal dialog box is open, titled 'Windows Security'. It asks 'Enter your credentials' and says 'These credentials will be used to connect to 135.235.173.24.' It shows a 'Native RDP' section with a password field containing 'tejasvi-vm'. There's a 'Remember me' checkbox and a 'More choices' link. At the bottom are 'OK' and 'Cancel' buttons.





Peering done successfully.

## 9. Conclusion

This task helped in understanding how networks are managed in Azure. I learned:

- What CIDR is and how IP ranges work
- How to create and divide a VNet into subnets
- How to launch Windows and Linux VMs in different subnets
- How to connect two different VNets using VNet Peering
- How to test communication between VMs

## 9. References

- Azure Virtual Network Overview
- CIDR Notation Basics
- Azure Documentation and Portal