Azure Virtual Networking

# Objective

To do research, documenting and applying the concepts of Azure Virtual Networking (CIDR ranges, subnets and VNet peering) by:

Create two subnets in the same virtual network (VNet), deploy two virtual machines: Linux in one subnet and Windows in the other and make them able to interact. Next, create another VNet and peer it with the first VNet and switch on VM interaction between VNets.

This is the introductory practical lab in learning about Azure networking, communications between different VMs and connectivity between VNets.

# Introduction

The Microsoft Azure core networking is based on Azure Virtual Network (VNet). It is an isolated and secure platform where one can deploy and manage resources in azure. Just like the typical on-premises networks, VNets will allow complete control of IP address ranges, DNS configuration, security policies, and routing.

In our documentation, we are going to cover several ways of working with VNets, how to plan IP ranges with CIDR notation, how to segment network with subnets, and how to enable communication among VNets with VNet peering: These are building blocks of cloud infrastructure and are essential for secure cloud environments.

# Key Concepts

## Azure Virtual Network (VNet)

Azure Virtual Network is the foundational concept which helps in implementing private networking in Azure. It serves like a conventional on-premises network but yet hosted in Azure cloud. You can use it to:

* Link the Azure resources such as Virtual Machine (VMs), Azure App Services, and so on.
* To apply security controls for network traffic protection, Helps manage IP address allocation and to configure DNS settings.
* Allow sharing of information between resources across Azure or with other networks including on-premises data centers through either VPN or ExpressRoute.

VNet will provide you with isolation, implying that the resources in it cannot be accessed by other VNets or other cloud tenants unless they are specifically interconnected by means of peering or gateways.

## Subnets

Logical partitions of Azure Virtual Network(VNet) are called subnets. It enables you to subdivide the VNet into small addresses space to effectively operate and isolate resources in the same network boundary.

Defining a subnet is in a way reserving a specific part of the VNet’s IP address range (defined using the CIDR notation) to a single set of resources e.g. virtual machines, databases or containers.

## Key Characteristics:

* CIDR-based Addressing:

Every subnet is described with a CIDR block which is required to fully fall within that VNet network space. For example, a valid subnet could be 10.0.1.0/24 only then if a VNet is using 10.0.0.0/16.

* Isolation and Traffic Flow:

The resources can communicate freely if within the same subnet. Nevertheless, the process of assigning Network Security Groups (NSGs) to subnets gives you the control to manage inbound and outbound traffic thus enabling safe separation of workloads.

* Resource Assignment:

All network interfaces (NICs) that are having connection to a virtual machine or to other resources in Azure must be a part of that single subnet. Multiple subnets cannot be covered by a NIC.

* Support of integration with Azure Services:

Other Azure services (such as Application Gateway, Bastion, or Azure Firewall) have requirements to have dedicated subnets with special naming convention (e.g. AzureFirewallSubnet).

# Design:

* Subnet Sizing:

Always design subnets on the current and future capacity. Remember that the Azure reserves 5 IP addresses in each subnet, so small subnets might not be viable.

* Non-Overlapping Ranges:

Each of the subnets must have assigned a different and non-overlapping CIDR block in the VNet. When overlapping IP ranges are allocated to two subnets it may cause failure in deployment of the subnets and communication issues in the network.

* Security Boundaries:

Subnets are the boundaries for the security rules. Placement of sensitive workloads in the isolated subnets and stringent NSG rules keeps you less exposed to the threats.

* Custom Routing:

Subnets can also be linked to User Defined Routes (UDRs) in order to regulate network traffic direction. This is usually deployed in case of incorporation with firewalls or local networks.

| **Resource** | **Address Range** | **Purpose** |
| --- | --- | --- |
| VNet | 10.0.0.0/16 | Main network space |
| Subnet-Linux | 10.0.1.0/24 | Linux VM subnet |
| Subnet-Windows | 10.0.2.0/24 | Windows VM subnet |

# CIDR

CIDR or Classless Inter-Domain Routing is an IP address range allocation system that is flexible and is scalable. Azure identifies both the Virtual Networks (VNets) and subnets within the same spaces using the CIDR notation to specify its IP address space.

CIDR is important in the design of the architecture of any cloud network. With the knowledge of deploying the concept of CIDR, you could prevent any IP overlap, effectively promote the communication between various resources, and establish a structure of networks that can expand with your system.

CIDR in Virtual Networks(VNet):

During the process of developing VNet, the user is required to provide it with an IP address space in CIDR notation. This is how many different IP addresses in total are available in that network to be used privately.

CIDR Format:

<IP address>/<prefix length>

For example:

10.0.0.0/16- It means Azure will allocate a total of 65,536 IP addresses.

This range can also be divided into many subnets as per your design.

Key points:

* There should be use of private IPs, as recommended by RFC 1918, like:
* 10.0.0.0 - 10.255.255.255
* 172.16.0.0 - 172.31.255.255
* 192.168.0.0 - 192.168.255.255

Azure supports many CIDR blocks to be added in one VNet so that in case you require expanding the network in the future, it would be available with ease.

* There should not be overlapping of CIDR blocks in a VNet and it must also not conflict with the address space of the VNet you plan to peer with.

CIDR in Subnets

VNet subdivides itself into subnets. Every subnet has its own CIDR range assigned to it, and this address space should fall within the total address space of VNet.

For example:

If VNet is making use of (10.0.0.0/16), the valid subnets will be:

* 10.0.1.0/24 (Linux VMs)
* 10.0.2.0/24 (windows machines VMs)

Important considerations:

* The CIDR block of each subnet should be unique as well as it should be non-overlapping and distinct from any other of the subnets in the same VNet.
* In each of the subnet the Azure will reserve 5 IP addresses :
* 1st : Network address
* 2nd: Default gateway
* 3rd 4th: Internal use only
* Last: Broadcast (not used, and yet reserved)
* This implies that a /24 subnet (which provides 256 total IP numbers), has 251 usable IPs.
* Unless there are free IPs in the subnet, you cannot deploy a VM, and the subnet size ought to be selected based upon the present and future capacity.

# VNet Peering

The Azure Virtual Network (VNet) Peering can be used to connect two VNets together directly and safely with the help of the Azure backbone network. Coming to VNet peering, resources (such as VMs) within one VNet can communicate with other ones via the use of binding IP addresses, similarly to the communication as these resources belong to the same network.

Peer eliminates use of gateways or public IP as a way of communication between VNets

Characteristics:

* There is low-latency and high bandwidth communication.
* It did not require any VPN gateways or public IPs.
* Non transitive: A peered with B, B peered with C is not equal to A is able to talk to C.
* CIDR range had to be non-overlapping in every VNet.
* In case of usage of remote gateway options, it should be peer to peer or it should be configured in both directions.

Types of VNet Peering

There are two types of peering which is supported by Azure:

1. **Intra-region VNet Peering**

Intra-region peering means to connect two VNets within the same Azure region, thus enabling direct private communication between them with the help of the internal backbone network of Azure.

Key Points:

* There is Extremely low latency and high bandwidth because the traffic remains in the same area.
* There is no extra data transfer fee (since egress charges may be minimal in some cases).
* It is suitable to use for microservices or multi-tier applications deployed into a single region.

Use case:

* A front–end web interface and backend API deployed in different VNets in the East US.
* A database deployed in a secure subnet in a different VNet in the East us.

Example CIDRs:

* VNet-Frontend: 10.0.0.0/16
* VNet-Backend: 10.1.0.0/16

Benefits:

* Makes architecture easy when it comes to applications which require network isolation but at the same time need communication.
* It would be so great to split it into departments or environments logically (e.g. dev vs. test).

1. **Global VNet Peering**

Global peering allows privately connecting VNets in separate Azure regions, but they are still located on the Microsoft backbone.

Key Points:

* It allows cross region networking without the use of VPNs or gateways.
* More latency (with regard to intra-region peering) because of distance.
* There could be charges on data transfer (depending on the source and destination region).

Example of a Use Case:

* The company is global and has a major application in the East US and a disaster recovery facility in West Europe.
* Log management servers in a hub area that is shared by many remote VNets across the world.

Example CIDRs:

* VNet-EastUS: 10.2.0.0/16
* VNet-WestEurope: 10.3.0.0/16

Benefits:

* It is perfect for geographically distributed regions.
* Reliable and secure connectivity will be there without exposing it to the internet.

Prerequisites:

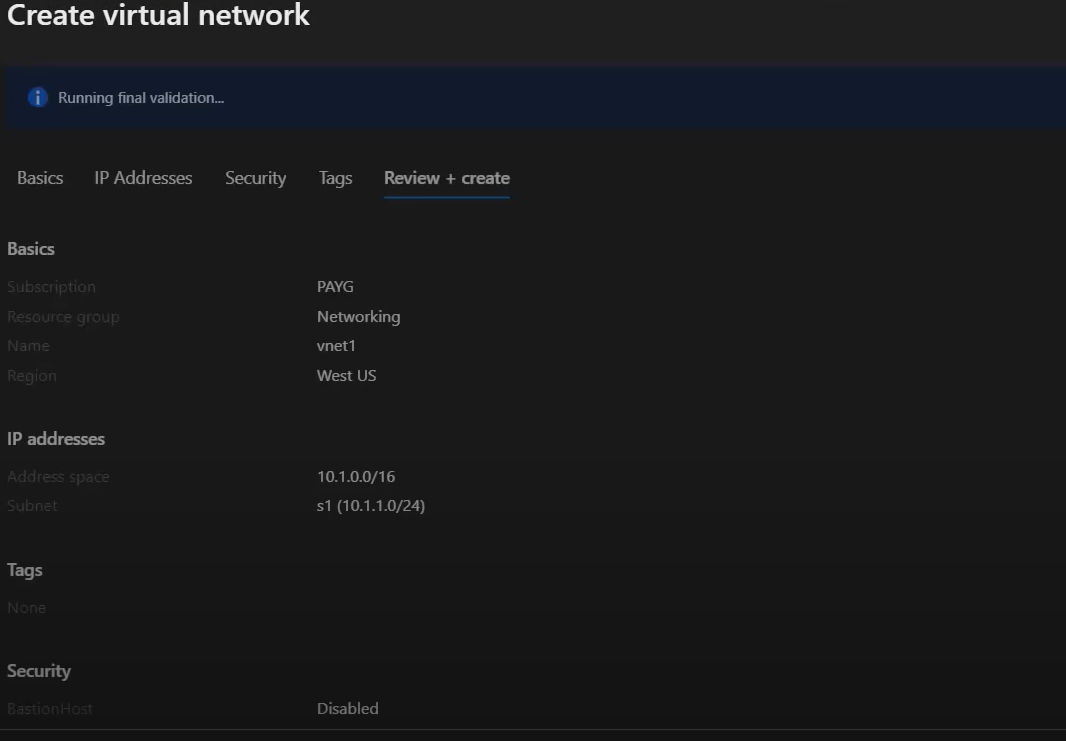
Prior to the provision of the actual implementation of the Azure Virtual Network setting, the following requirements were attained:

* Azure Subscription: an active account in Microsoft Azure with access to the free or paid plan.
* Resource Group: A resource group VNetLab-RG was created in order to contain and coordinate the associated resources.
* Deployment:
* The deployment of vnet1 occurred in West US
* The deployment of vnet2 occurred in East US
* IP Planning: Each VNet had a Non-overlapping CIDR range and its subnets in order for routing to be compatible.
* Environment Access: All deployments were done with the use of the Azure Portal GUI. Neither CLI nor scripting.
* General Knowledge of Networking: There was a need to have an idea of the basic concepts like, the IP addressing, the CIDR format, subnets, and routing.

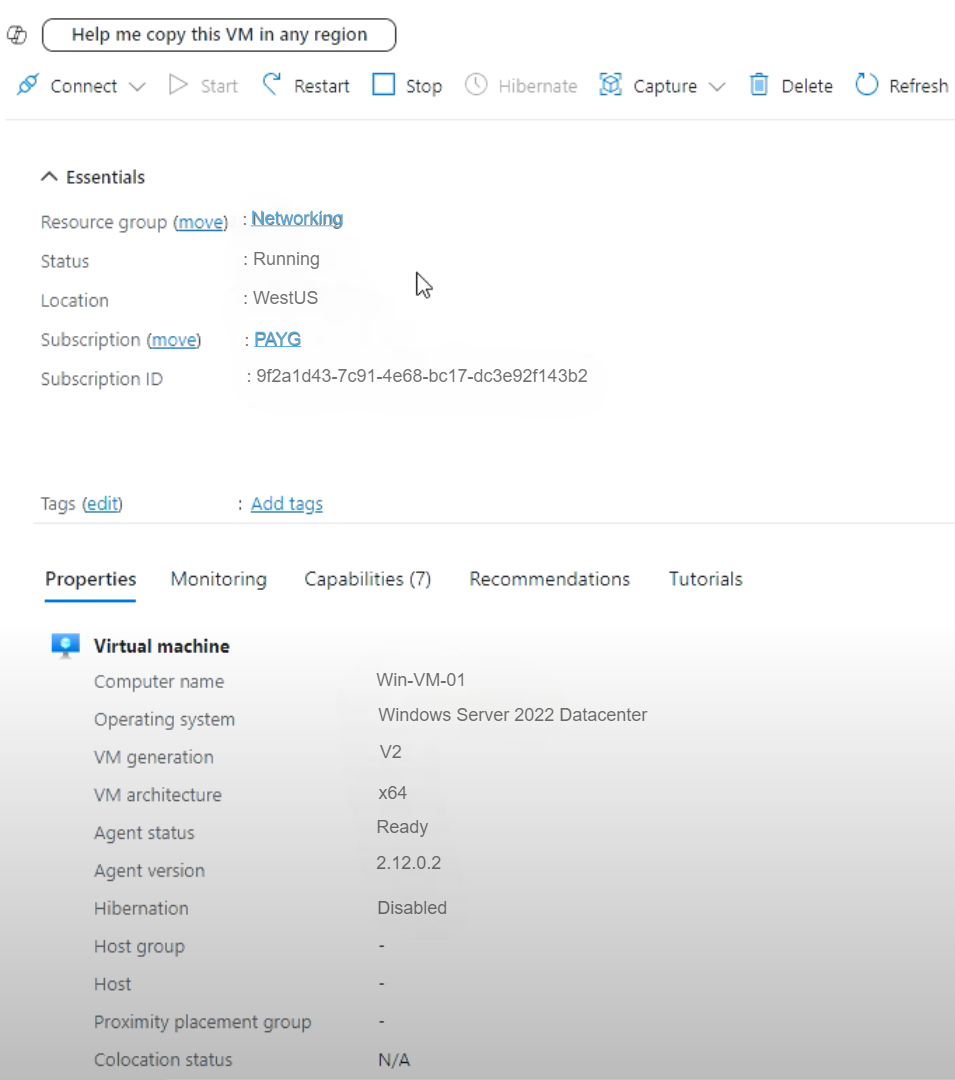
Use case:

In this R&D project, the emphasis will be placed on the planning and implementing secure, geographically dispersed cloud network on Microsoft Azure. The goal is to create two distinct virtual networks (vnet1 and vnet2), and in each of them there is one subnet and one virtual machine. Windows VM is running in VNet1 and the Linux VM is in VNet2. Once it is configured then the VNets peer with each other through Azure VNet peering option to enable secure and low-latency traffic at a cross-regional level. The solution has been successful as both VMs were able to ping each other using the private IP.

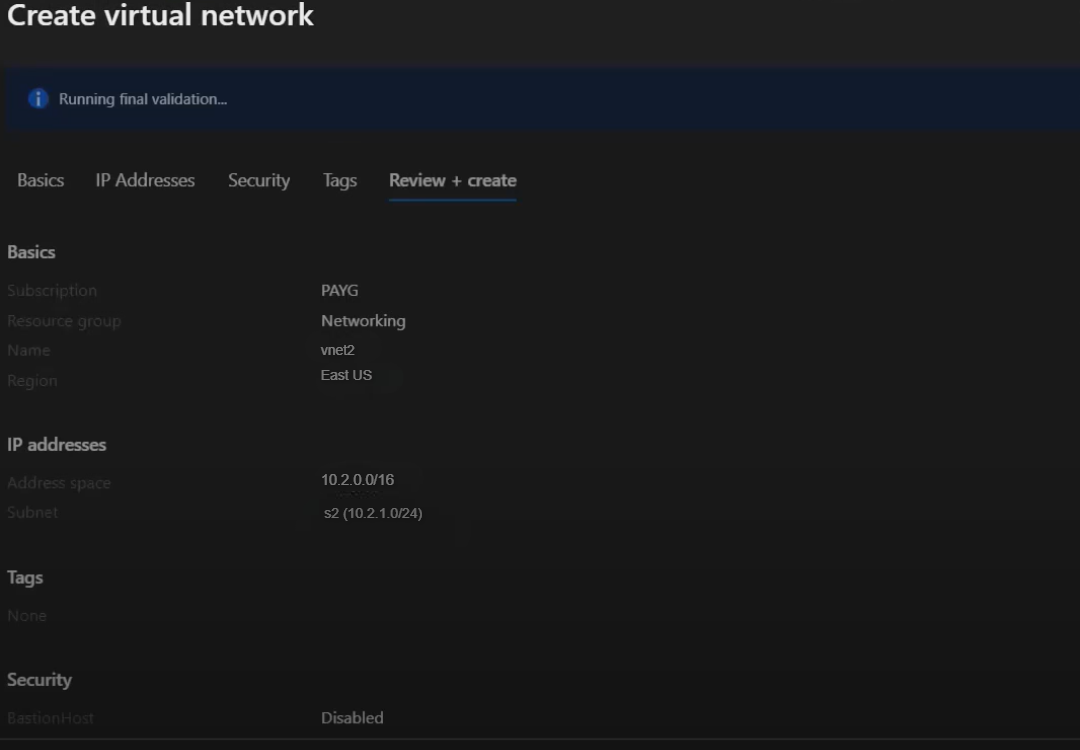
Implementation:



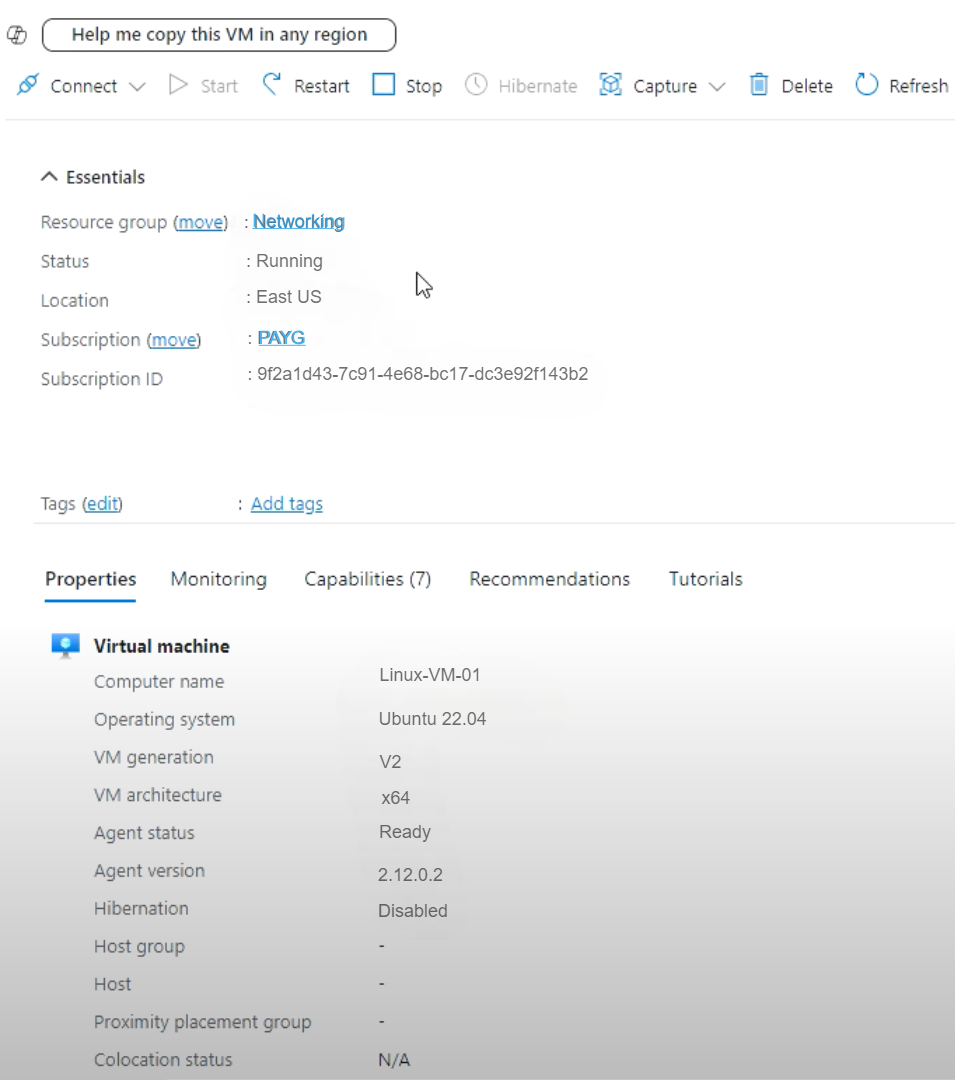
Virtual Network vnet1 has been created in the West US with using address space 10.1.0.0/16, and also containing subnet s1 (10.1.1.0/24) to host Windows VM.



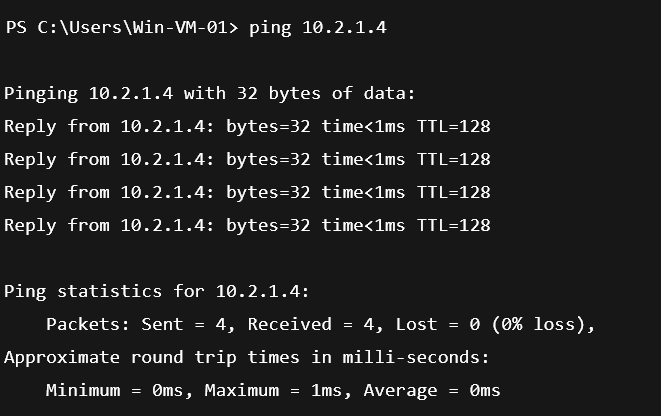
A Windows Server VM with name of Win-VM-01 has been deployed in subnet s1 of VNet1. Private IP from 10.1.1.0/24 has been used by the VM.



Virtual Network vnet2 created in the East US with the help of non-overlapping CIDR 10.2.0.0/16, and is containing subnet s2 (10.2.1.0/24) for Linux VM.

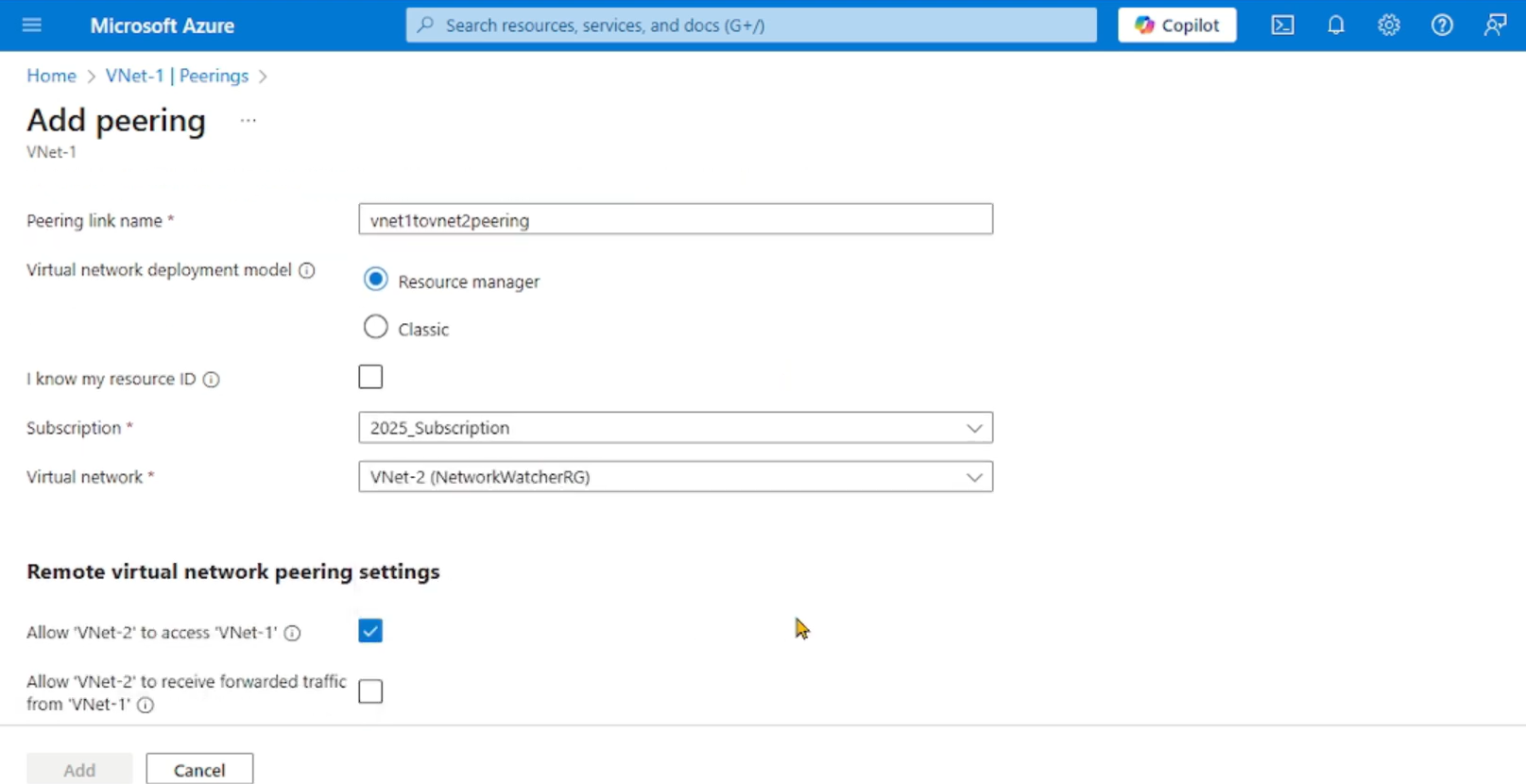


An Ubuntu Linux VM with name of Linux-VM-01 is deployed in subnet s2 of VNet2. Private IP from 10.2.1.0/24 has been used by the VM.



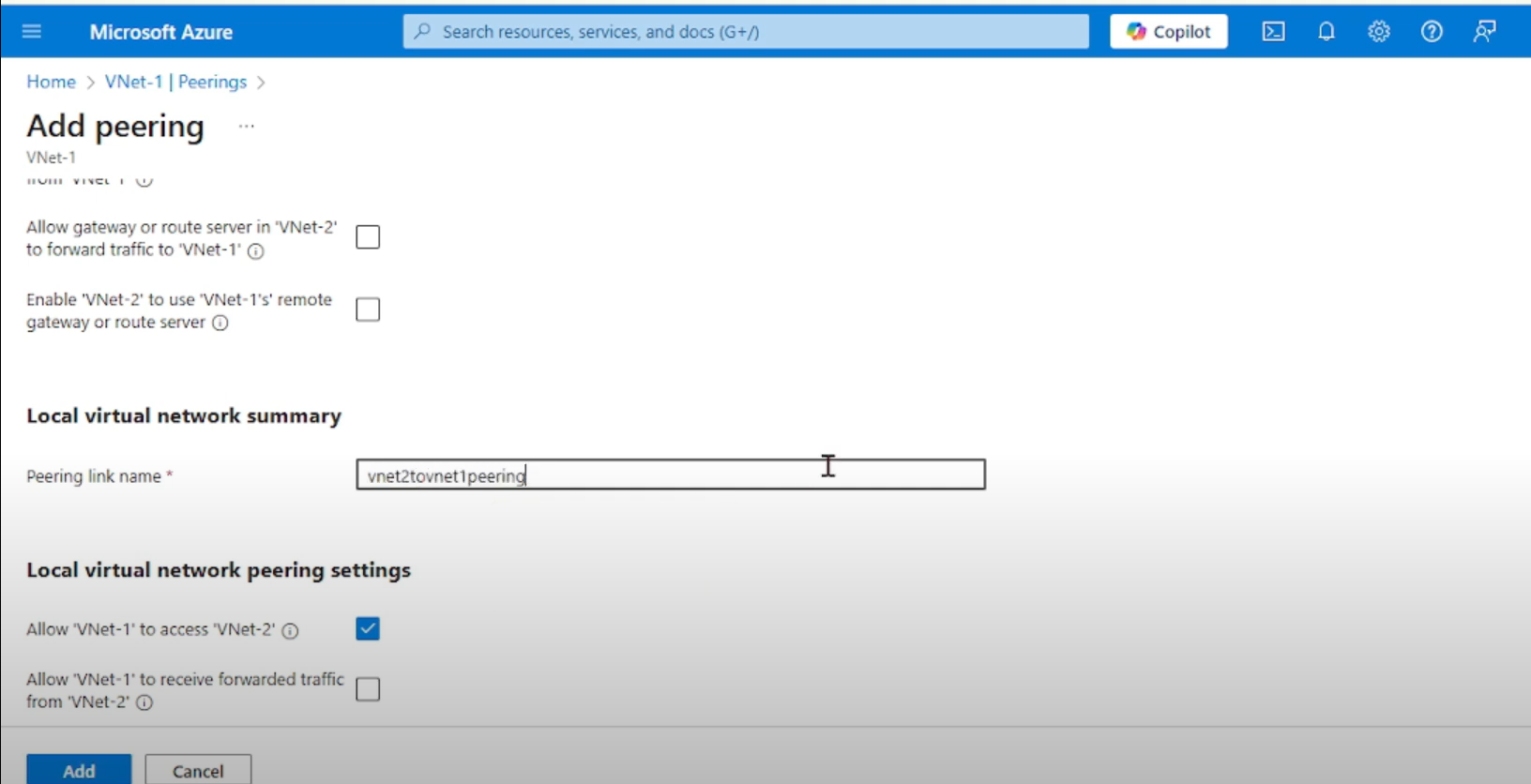
Ping from Win-VM-01 (10.1.1.4) to Linux-VM-01 (10.2.1.4) is showing 100% success, which also means that there is established communication over VNet peering with 0% packet loss.

VNet peering in Azure:



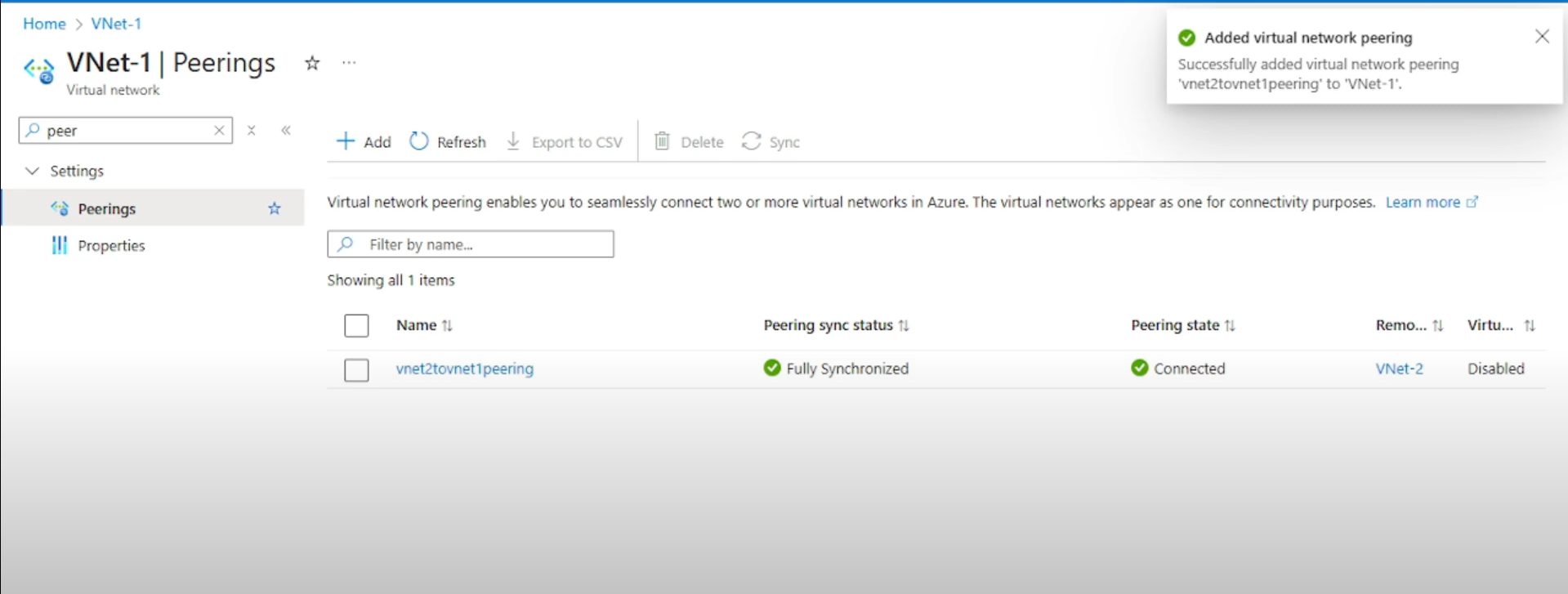
**VNet1 to VNet2 peering:**

VNet-1 to VNet-2 creates the peering link vnet1tovnet2peering with the help of Resource Manager model. Access permission is given to VNet-2, so that it can communicate with VNet-1.



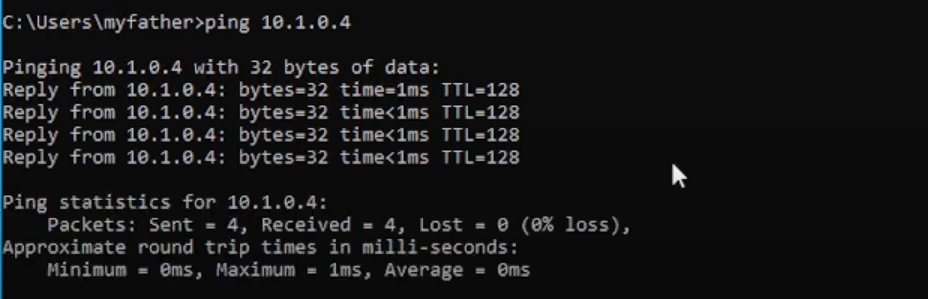
**VNet2 to VNet1 peering:**

Such peering (vnet2tovnet1peering) allows VNet-1 to access to VNet-2 and creates a bidirectional communication between the two networks.



**Successfully established:**

The peering status shows Fully synchronized and connected and hence the two VNets have now been connected and communication between the two is possible.



**Ping Validation:**

There was no 0% loss during ping from a VM in one VNet to another which shows inter-VM connectivity is working.

Conclusion

In this R&D project, the development of Azure Virtual Networks (VNets), subnets and virtual machines through the utilisation of CIDR-based addressing was successfully proved. In an experiential manner, we created two VNets in different regions both with a Windows and Linux VM in respective subnets. The VNet peering helps in efficient connection via the VNets without public IP addresses and/or VPNs. The success of the ping checks confirmed the connectivity among regions using private IP. Overall, this given lab made the basics of Azure network requirements clearer, such as the address planning, network isolation, and the means of secure VM-to-VM contact in the cloud.