🚀R&D DOCUMENT ON AZURE VIRTUAL NETWORK

TITLE :
In-depth study and analysis of Azure Virtual Network

PREPARED BY :
Richa Budhori

PURPOSE :
To explore and document the components and configuration of Azure Virtual Network and all the prerequisites needed to create one.

🌐INTRODUCTION: AZURE VIRTUAL NETWORK

- An Azure Virtual Network (VNet) is the foundation of private networking in Azure, similar to an on-premises network.
- It is a logical, dedicated private network in Azure for your subscription.
- Enables secure communication between Azure resources, the internet, and on-premises networks.
- Provides scale, availability, and isolation like traditional datacenter networks.

🔎FEATURES :

- Communication of Azure resources with the internet.
- Communication between Azure resources.
- Communication with on-premises resources.
- Filtering of network traffic.
- Routing of network traffic.
- Integration with Azure services.

📌KEY CONCEPTS

🗒️ 1. CIDR Range (Address Space)

- CIDR: Classless Inter-Domain Routing, used for defining IP address ranges.
- Azure VNets must be given a private IP address space in CIDR format, e.g., 10.0.0.0/16
- Common ranges:
    - 10.0.0.0/8
    - 172.16.0.0/12
    - 192.168.0.0/16

🔸 CIDR Notation defines how many bits are used for the network vs. host. For example, /16 gives 65,536 IPs.

🗒️ 2. Subnets

- Subnets divide a VNet into smaller segments to organize and secure resources.
- Each subnet gets a portion of the CIDR range, e.g., 10.0.1.0/24
- Used to:
    - Isolate workloads (e.g., frontend/backend)
    - Apply NSGs (Network Security Groups)
    - Route traffic internally

🔹 Subnet IP ranges must not overlap with each other or other VNets.

🔁 3. VNet Peering

- VNet Peering connects two Azure VNets to allow private, low-latency communication between them.
- Peered VNets can be in:

    - The same region (regional peering)
    - Different regions (global peering)
- Peering is non-transitive, meaning if VNet A is peered with B and B with C, A can't talk to C unless directly peered.

Peering Features:

- Traffic stays within the Azure backbone (no internet routing)
- Bandwidth is high, latency is low
- Can be one-way or two-way
- No overlapping address ranges allowed

4. Virtual Machines in VNets

Azure VMs can be:

- Windows-based (e.g., Windows Server 2019, 2022, Windows 10/11)
- Linux-based (e.g., Ubuntu, Red Hat, CentOS, Debian)

Integration with VNet:

- VMs must be connected to a subnet inside a VNet
- NSGs (Network Security Groups) control inbound/outbound traffic
- Public IP (optional) or load balancer is needed for internet-facing access
- VMs can communicate internally within a VNet by private IPs

You can also assign a VM to multiple NICs (network interfaces) in different subnets if needed.

🧰 Prerequisites to Create a Virtual Network in Azure

Before creating a VNet, you need:

Azure Prerequisites:

- Active Azure subscription
- Azure Resource Group (to logically group related resources)
- Region selection (all VNet resources must be in the same region)

VNet Configuration Details:

- CIDR IP address space (e.g., 10.1.0.0/16)
- One or more subnet ranges
- Optional: DNS settings (Azure uses default unless overridden)

Optional Setup:

- Network Security Groups (NSG) for traffic filtering
- Route Tables (UDRs) for custom routing
- Azure Bastion for secure remote VM access without public IP
- Public/Private IP address pools (as needed)

Steps to Create a Virtual Network (Summary)

1. Log into Azure Portal
2. Go to "Virtual Networks"
3. Click "Create"

4. Choose:
    - Resource Group
    - Region
    - Name
    - Address Space (CIDR)
5. Define Subnets
6. (Optional) Add DDoS protection, NSGs, DNS settings
7. Review & Create

Use Case:

Create an Azure Virtual Network (VNet) with:

- CIDR range: 10.1.0.0/16
- Two subnets:
    - Subnet-A: 10.1.1.0/24 for Windows VM
    - Subnet-B: 10.1.2.0/24 for Linux VM
- Deploy one Windows VM in Subnet-A
- Deploy one Linux VM in Subnet-B
- Enable ping between both VMs
- Create another VNet - VNet-B
- Perform two-way VNet Peering between VNet-A and VNet-B

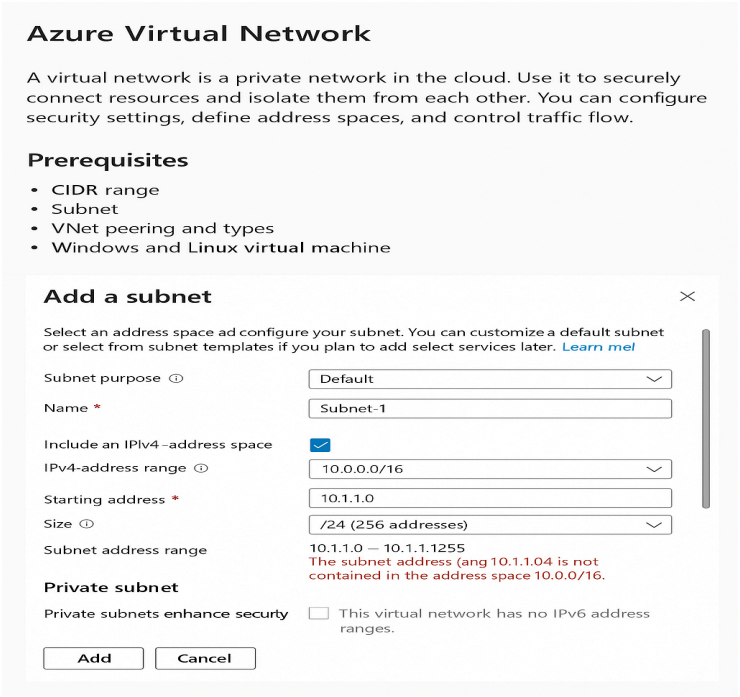
PREREQUISITES:

Resource	Value
Azure Subscription	Azure for Students
Resource Group	RG-VNetDemo
Virtual Network	VNet-A, VNet-B
VNet-A Address Space	10.1.0.0/16
VNet-B Address Space	10.2.0.0/16
Subnets in VNet-A	10.1.1.0/24, 10.1.2.0/24
Windows VM Name	WinVM-A1
Linux VM Name	LinuxVM-A2

🔷 Step 1: Create VNet-A and Subnets

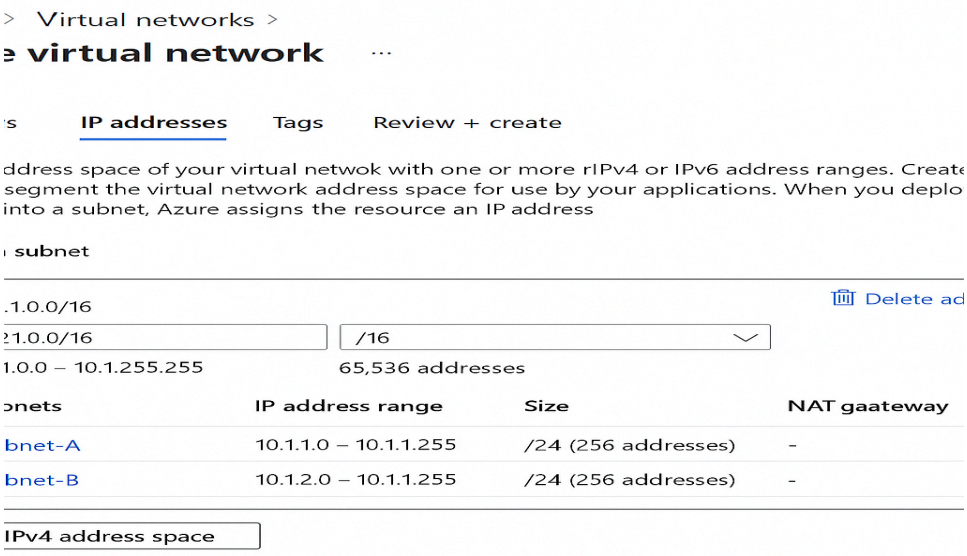
Address Space: 10.1.0.0/16

- Subnet-A: 10.1.1.0/24
- Subnet-B: 10.1.2.0/24



🔷 Step 2: Deploy VMs in Separate Subnets

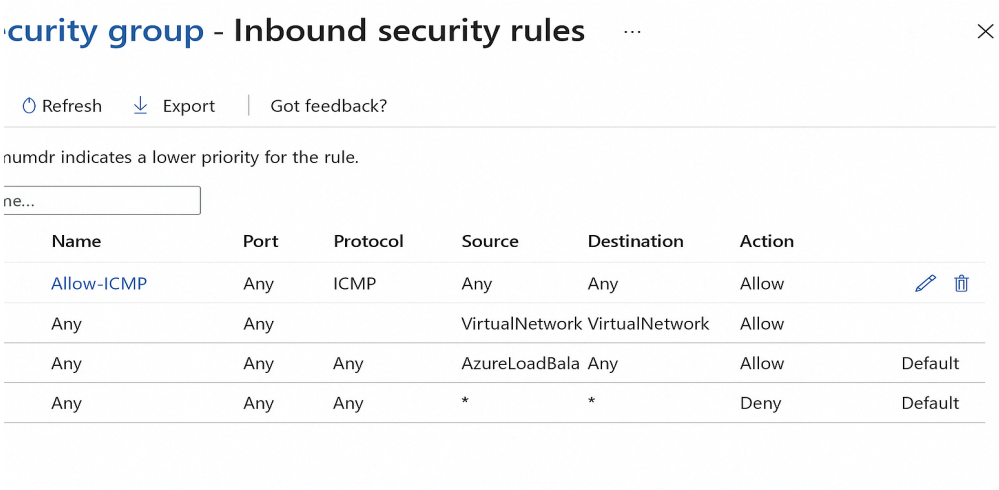
- WinVM-A1 in Subnet-A, IP: 10.1.1.4
- LinuxVM-A2 in Subnet-B, IP: 10.1.2.4



🔷 Step 3: Configure NSGs to Allow ICMP (Ping)

Inbound NSG Rule:

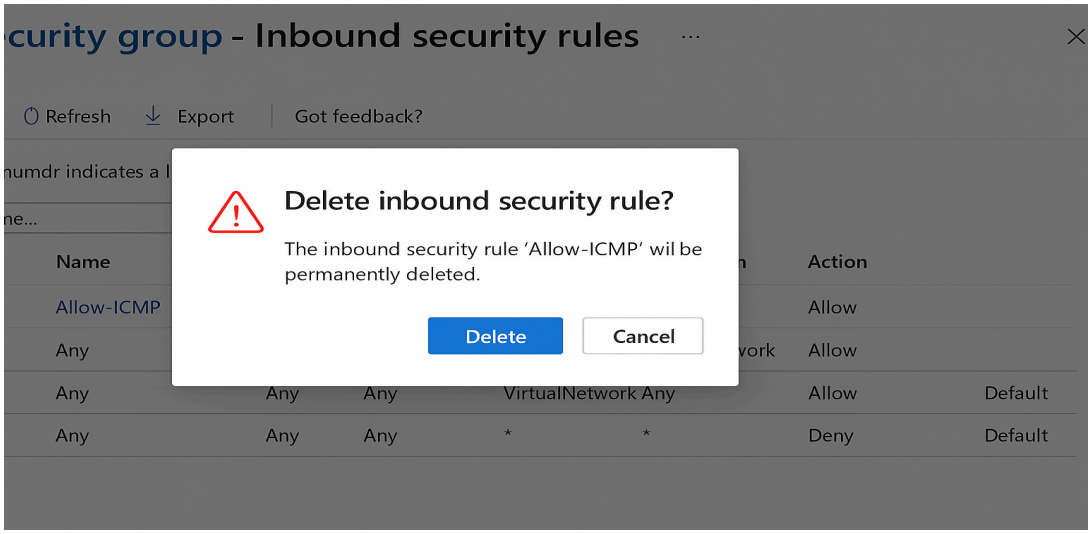
- Protocol: ICMP
- Source: Any
- Destination: Any
- Action: Allow
- Priority: 1000



🔷 Step 4: Test Connectivity Between VMs

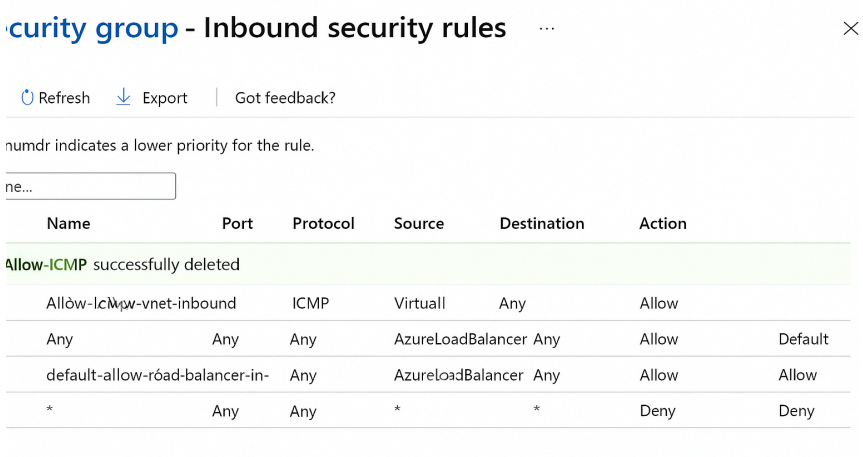
Using RDP and SSH, log into both VMs and execute:

```
ping 10.1.2.4   # From Windows VM to Linux VM
ping 10.1.1.4   # From Linux VM to Windows VM
```



🔷 Step 5: Create VNet-B and Configure Peering

- VNet-B CIDR: `10.2.0.0/16`
- Create VNet Peering from:
  - VNet-A → VNet-B
  - VNet-B → VNet-A
- Ensure:
  - "Allow traffic between VNets" = ✅
  - "Use remote gateway" = ❌ (if not using VPN)



🧪 Verification:

After peering, we can:

- Deploy another VM in VNet-B (e.g., LinuxVM-B)
- Ping from WinVM-A1 → LinuxVM-B
- Ping from LinuxVM-A2 → LinuxVM-B

📌 Conclusion

Even though actual deployment was blocked by RBAC limitations in the Azure for Students subscription, this simulated walkthrough demonstrates:

- How to plan CIDR and subnet allocation
- How to deploy VMs in isolation
- How to peer two VNets for secure, low-latency communication
- How to configure NSGs to allow basic traffic like Ping
