In this document, we will walk through the process of creating two virtual networks (VNETs) in Azure, each with a subnet and a virtual machine (VM). We will then configure VNET peering between these VNETs to ensure the VMs can communicate with each other.

**Prerequisites**

* An active Azure subscription
* Basic knowledge of Azure portal and networking concepts

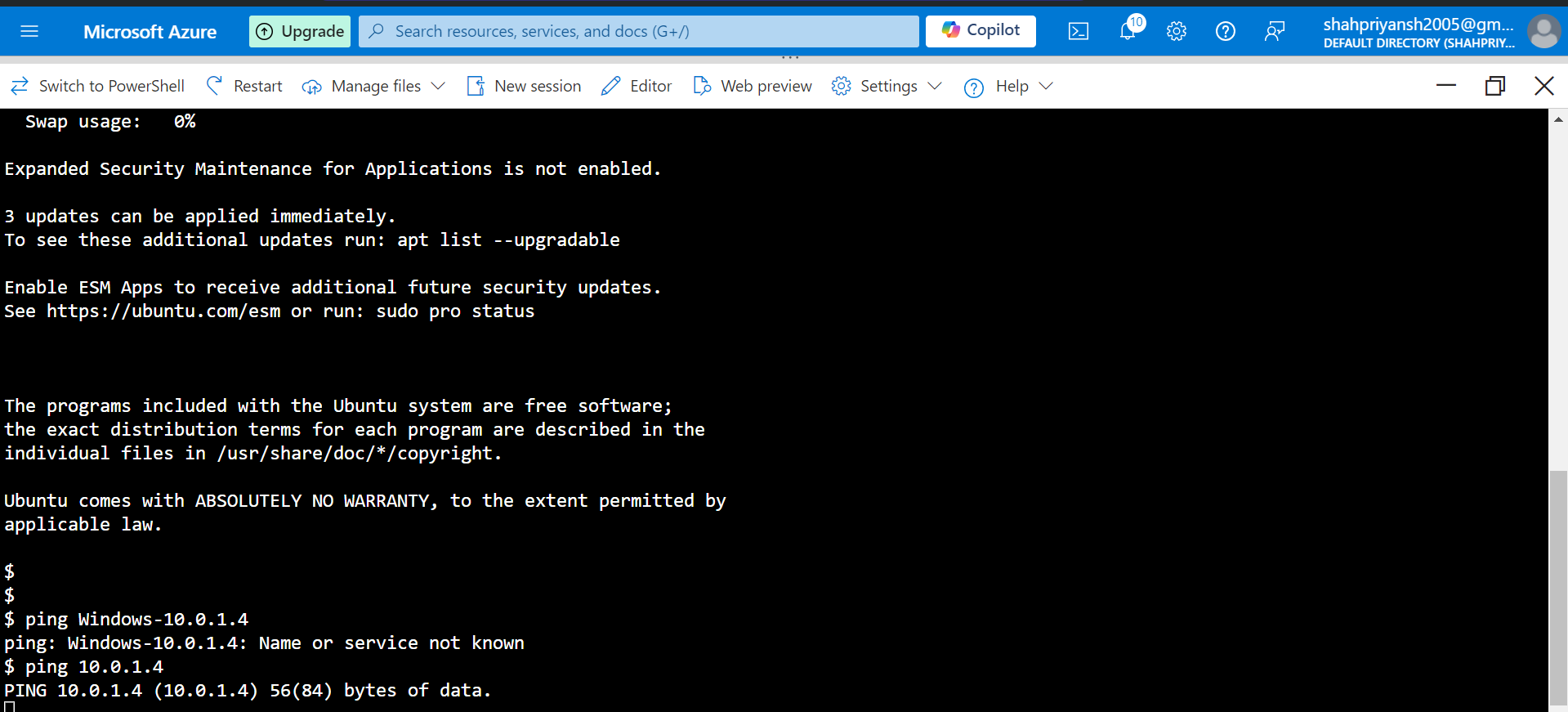
**Step-by-Step Process**

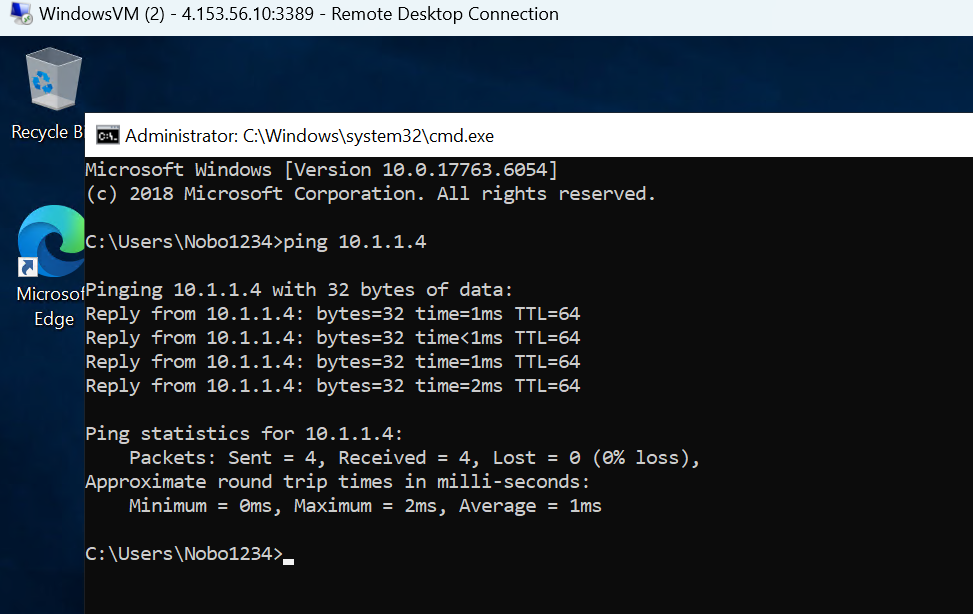
1. **Create VNET1 and Subnet1**
   * **Navigate to the Azure portal** and select **Create a resource** > **Networking** > **Virtual network**.
   * **Configure VNET1**:
     + Name: VNET1
     + Address space: 10.0.0.0/16
     + Resource group: Select or create a new one
     + Location: Choose a region
   * **Add Subnet1**:
     + Subnet name: Subnet1
     + Subnet address range: 10.0.1.0/24
   * **Review and create**.
2. **Create VNET2 and Subnet2**
   * **Navigate to the Azure portal** and select **Create a resource** > **Networking** > **Virtual network**.
   * **Configure VNET2**:
     + Name: VNET2
     + Address space: 10.1.0.0/16
     + Resource group: Select or create a new one
     + Location: Choose a region
   * **Add Subnet2**:
     + Subnet name: Subnet2
     + Subnet address range: 10.1.1.0/24
   * **Review and create**.
3. **Launch Windows VM in Subnet1 (VNET1)**
   * **Navigate to the Azure portal** and select **Create a resource** > **Compute** > **Virtual machine**.
   * **Configure Windows VM**:
     + Name: WindowsVM
     + Region: Same as VNET1
     + Image: Windows Server 2019
     + Username and password: Set your credentials
     + Allow RDP port (3389)
   * **Networking**:
     + Virtual network: VNET1
     + Subnet: Subnet1
   * **Review and create**.
4. **Launch Linux VM in Subnet2 (VNET2)**
   * **Navigate to the Azure portal** and select **Create a resource** > **Compute** > **Virtual machine**.
   * **Configure Linux VM**:
     + Name: LinuxVM
     + Region: Same as VNET2
     + Image: Ubuntu Server 20.04 LTS
     + Username and password: Set your credentials
     + Allow SSH port (22)
   * **Networking**:
     + Virtual network: VNET2
     + Subnet: Subnet2
   * **Review and create**.
5. **Ensure VMs Can Ping Each Other**
   * **Configure NSGs for ICMP**:
     + Add inbound rule for ICMP in NSG associated with Subnet1.
     + Add inbound rule for ICMP in NSG associated with Subnet2.
6. **Create VNET Peering**
   * **VNET1 to VNET2**:
     + Go to VNET1 > Peerings > Add.
     + Configure peering with VNET2.
   * **VNET2 to VNET1**:
     + Go to VNET2 > Peerings > Add.
     + Configure peering with VNET1.

**Verification**

* **Ping from Windows VM to Linux VM**:
  + Connect to the Windows VM using RDP and ping the Linux VM’s private IP address.
* **Ping from Linux VM to Windows VM**:
  + Connect to the Linux VM using SSH and ping the Windows VM’s private IP address.

**Screenshots**

Screenshot of LinuxVm pinging WindowsVM  


Screenshot of WindowsVM pinging LinuxVM  


**Conclusion**

In this document, we successfully created two VNETs, each with a subnet and a VM, and configured VNET peering to ensure the VMs can communicate with each other.