**What is Azure Virtual Network?**

Azure Virtual Network (VNet) is the fundamental building block for your private network in Azure. VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks. VNet is similar to a traditional network that you'd operate in your own data center, but brings with it additional benefits of Azure's infrastructure such as scale, availability, and isolation.

**VNet concepts**

* **Address space:** When creating a VNet, you must specify a custom private IP address space using public and private (RFC 1918) addresses. Azure assigns resources in a virtual network a private IP address from the address space that you assign. For example, if you deploy a VM in a VNet with address space, 10.0.0.0/16, the VM will be assigned a private IP like 10.0.0.4.
* **Subnets:** Subnets enable you to segment the virtual network into one or more sub-networks and allocate a portion of the virtual network's address space to each subnet. You can then deploy Azure resources in a specific subnet. Just like in a traditional network, subnets allow you to segment your VNet address space into segments that are appropriate for the organization's internal network. This also improves address allocation efficiency. You can secure resources within subnets using Network Security Groups. For more information, see [Network security groups](https://docs.microsoft.com/en-us/azure/virtual-network/security-overview).
* **Regions**: VNet is scoped to a single region/location; however, multiple virtual networks from different regions can be connected together using Virtual Network Peering.
* **Subscription:** VNet is scoped to a subscription. You can implement multiple virtual networks within each Azure [subscription](https://docs.microsoft.com/en-us/azure/azure-glossary-cloud-terminology?toc=/azure/virtual-network/toc.json#subscription) and Azure [region](https://docs.microsoft.com/en-us/azure/azure-glossary-cloud-terminology?toc=/azure/virtual-network/toc.json#region).

**Best practices**

As you build your network in Azure, it is important to keep in mind the following universal design principles:

* Ensure non-overlapping address spaces. Make sure your VNet address space (CIDR block) does not overlap with your organization's other network ranges.
* Your subnets should not cover the entire address space of the VNet. Plan ahead and reserve some address space for the future.
* It is recommended you have fewer large VNets than multiple small VNets. This will prevent management overhead.
* Secure your VNet's by assigning Network Security Groups (NSGs) to the subnets beneath them.

**Communicate with the internet**

All resources in a VNet can communicate outbound to the internet, by default. You can communicate inbound to a resource by assigning a public IP address or a public Load Balancer. You can also use public IP or public Load Balancer to manage your outbound connections. To learn more about outbound connections in Azure, see [Outbound connections](https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-outbound-connections), [Public IP addresses](https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-public-ip-address), and [Load Balancer](https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview).

**Note**

When using only an internal [**Standard Load Balancer**](https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-standard-overview), outbound connectivity is not available until you define how you want [**outbound connections**](https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-outbound-connections) to work with an instance-level public IP or a public Load Balancer.

**Communicate between Azure resources**

Azure resources communicate securely with each other in one of the following ways:

* **Through a virtual network**: You can deploy VMs, and several other types of Azure resources to a virtual network, such as Azure App Service Environments, the Azure Kubernetes Service (AKS), and Azure Virtual Machine Scale Sets. To view a complete list of Azure resources that you can deploy into a virtual network, see [Virtual network service integration](https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-for-azure-services).
* **Through a virtual network service endpoint**: Extend your virtual network private address space and the identity of your virtual network to Azure service resources, such as Azure Storage accounts and Azure SQL Database, over a direct connection. Service endpoints allow you to secure your critical Azure service resources to only a virtual network. To learn more, see [Virtual network service endpoints overview](https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-service-endpoints-overview).
* **Through VNet Peering**: You can connect virtual networks to each other, enabling resources in either virtual network to communicate with each other, using virtual network peering. The virtual networks you connect can be in the same, or different, Azure regions. To learn more, see [Virtual network peering](https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-peering-overview).

**Communicate with on-premises resources**

You can connect your on-premises computers and networks to a virtual network using any combination of the following options:

* **Point-to-site virtual private network (VPN):** Established between a virtual network and a single computer in your network. Each computer that wants to establish connectivity with a virtual network must configure its connection. This connection type is great if you're just getting started with Azure, or for developers, because it requires little or no changes to your existing network. The communication between your computer and a virtual network is sent through an encrypted tunnel over the internet. To learn more, see [Point-to-site VPN](https://docs.microsoft.com/en-us/azure/vpn-gateway/point-to-site-about?toc=/azure/virtual-network/toc.json).
* **Site-to-site VPN:** Established between your on-premises VPN device and an Azure VPN Gateway that is deployed in a virtual network. This connection type enables any on-premises resource that you authorize to access a virtual network. The communication between your on-premises VPN device and an Azure VPN gateway is sent through an encrypted tunnel over the internet. To learn more, see [Site-to-site VPN](https://docs.microsoft.com/en-us/azure/vpn-gateway/design?toc=/azure/virtual-network/toc.json#s2smulti).
* **Azure ExpressRoute:** Established between your network and Azure, through an ExpressRoute partner. This connection is private. Traffic does not go over the internet. To learn more, see [ExpressRoute](https://docs.microsoft.com/en-us/azure/expressroute/expressroute-introduction?toc=/azure/virtual-network/toc.json).

**Filter network traffic**

You can filter network traffic between subnets using either or both of the following options:

* **Network security groups:** Network security groups and application security groups can contain multiple inbound and outbound security rules that enable you to filter traffic to and from resources by source and destination IP address, port, and protocol. To learn more, see [Network security groups](https://docs.microsoft.com/en-us/azure/virtual-network/security-overview#network-security-groups) or [Application security groups](https://docs.microsoft.com/en-us/azure/virtual-network/security-overview#application-security-groups).
* **Network virtual appliances:** A network virtual appliance is a VM that performs a network function, such as a firewall, WAN optimization, or other network function. To view a list of available network virtual appliances that you can deploy in a virtual network, see [Azure Marketplace](https://azuremarketplace.microsoft.com/marketplace/apps/category/networking?page=1&subcategories=appliances).

**Route network traffic**

Azure routes traffic between subnets, connected virtual networks, on-premises networks, and the Internet, by default. You can implement either or both of the following options to override the default routes Azure creates:

* **Route tables:** You can create custom route tables with routes that control where traffic is routed to for each subnet. Learn more about [route tables](https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-udr-overview#user-defined).
* **Border gateway protocol (BGP) routes:** If you connect your virtual network to your on-premises network using an Azure VPN Gateway or ExpressRoute connection, you can propagate your on-premises BGP routes to your virtual networks. Learn more about using BGP with [Azure VPN Gateway](https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-bgp-overview?toc=/azure/virtual-network/toc.json) and [ExpressRoute](https://docs.microsoft.com/en-us/azure/expressroute/expressroute-routing?toc=/azure/virtual-network/toc.json#dynamic-route-exchange).

**Virtual network integration for Azure services**

Integrating Azure services to an Azure virtual network enables private access to the service from virtual machines or compute resources in the virtual network. You can integrate Azure services in your virtual network with the following options:

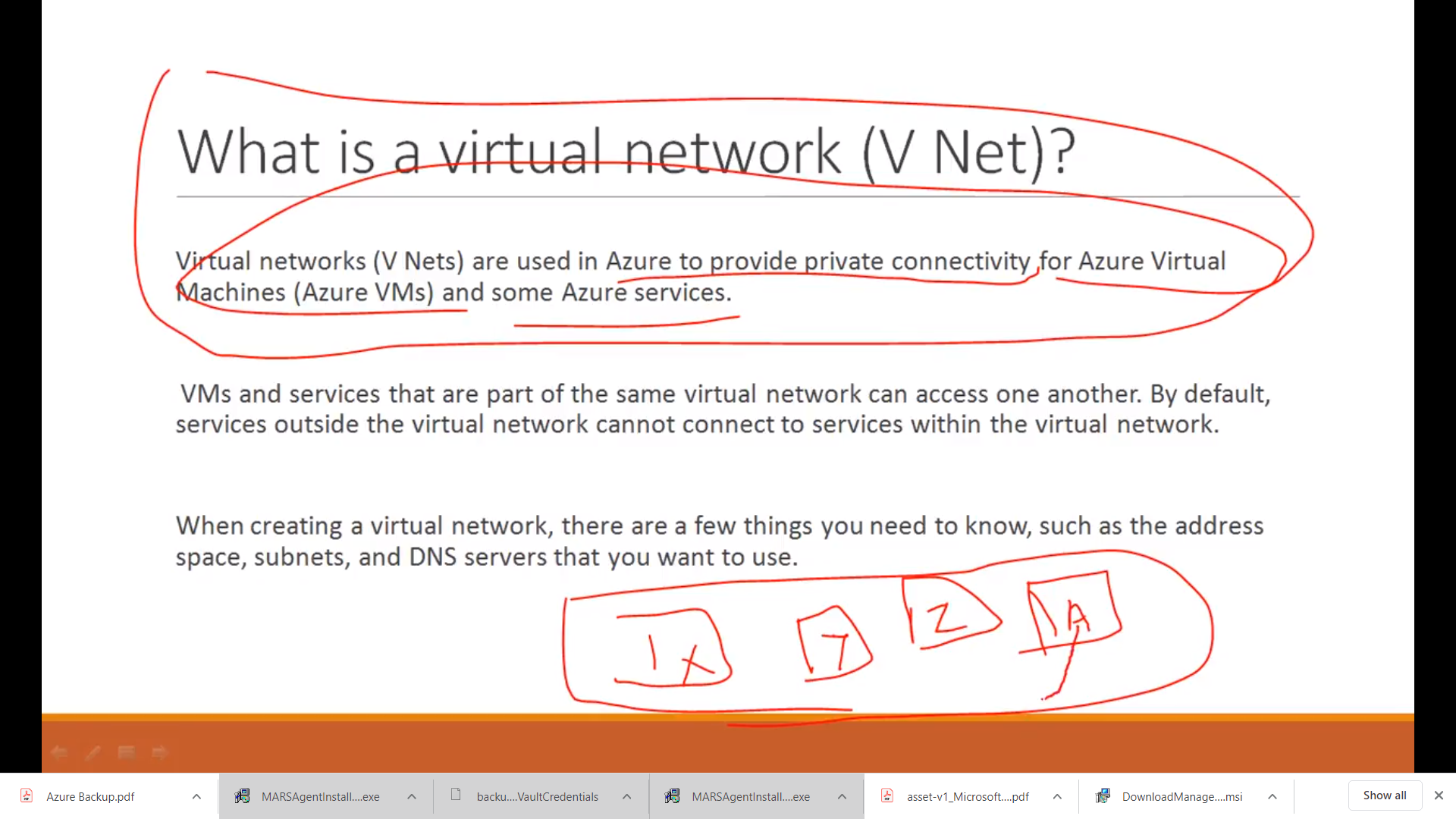
* Deploying [dedicated instances of the service](https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-for-azure-services) into a virtual network. The services can then be privately accessed within the virtual network and from on-premises networks.
* Using [Private Link](https://docs.microsoft.com/en-us/azure/private-link/private-link-overview) to access privately a specific instance of the service from your virtual network and from on-premises networks.
* You can also access the service using public endpoints by extending a virtual network to the service, through [service endpoints](https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-service-endpoints-overview). Service endpoints allow service resources to be secured to the virtual network.

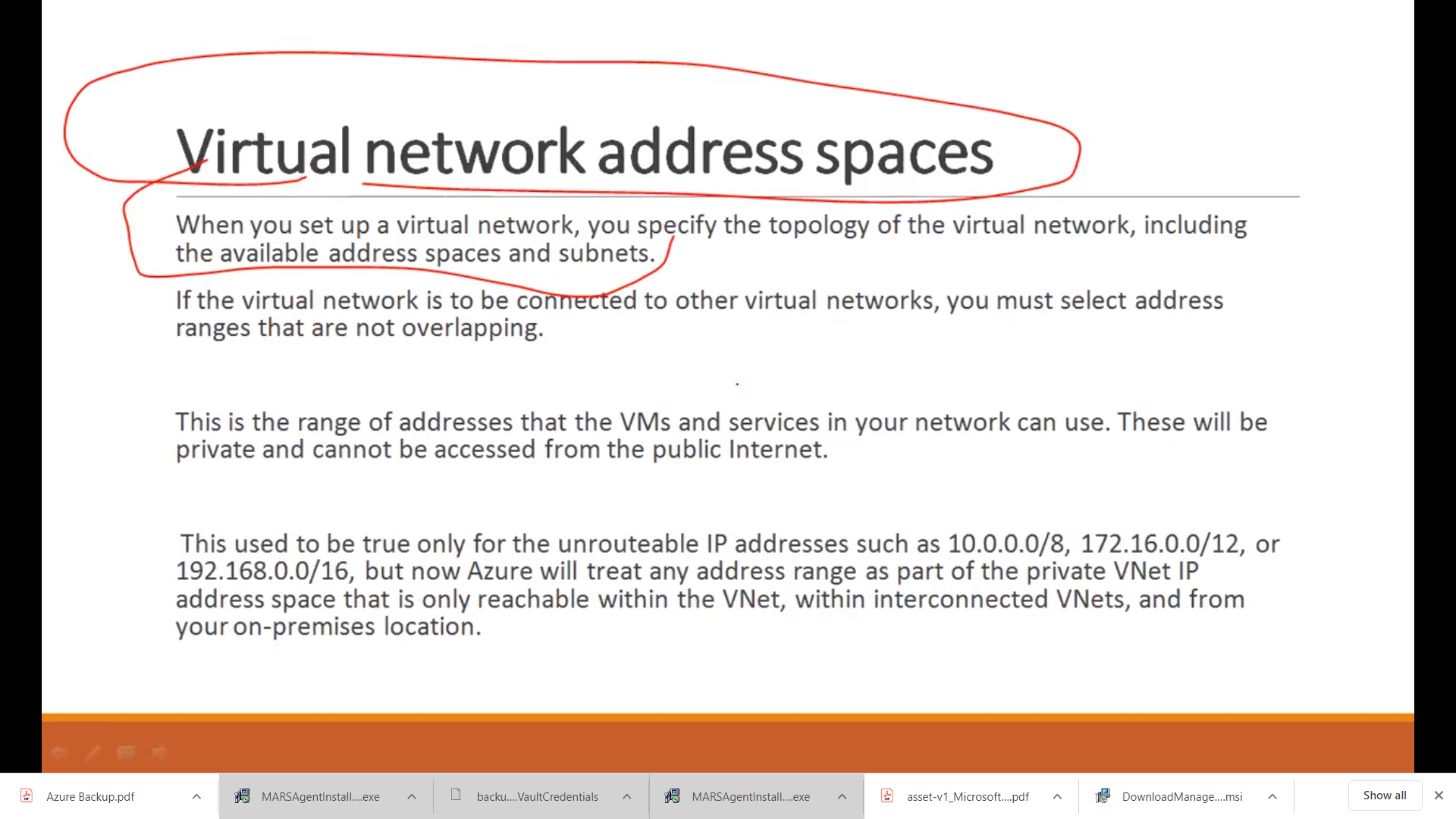
**Azure VNet limits**

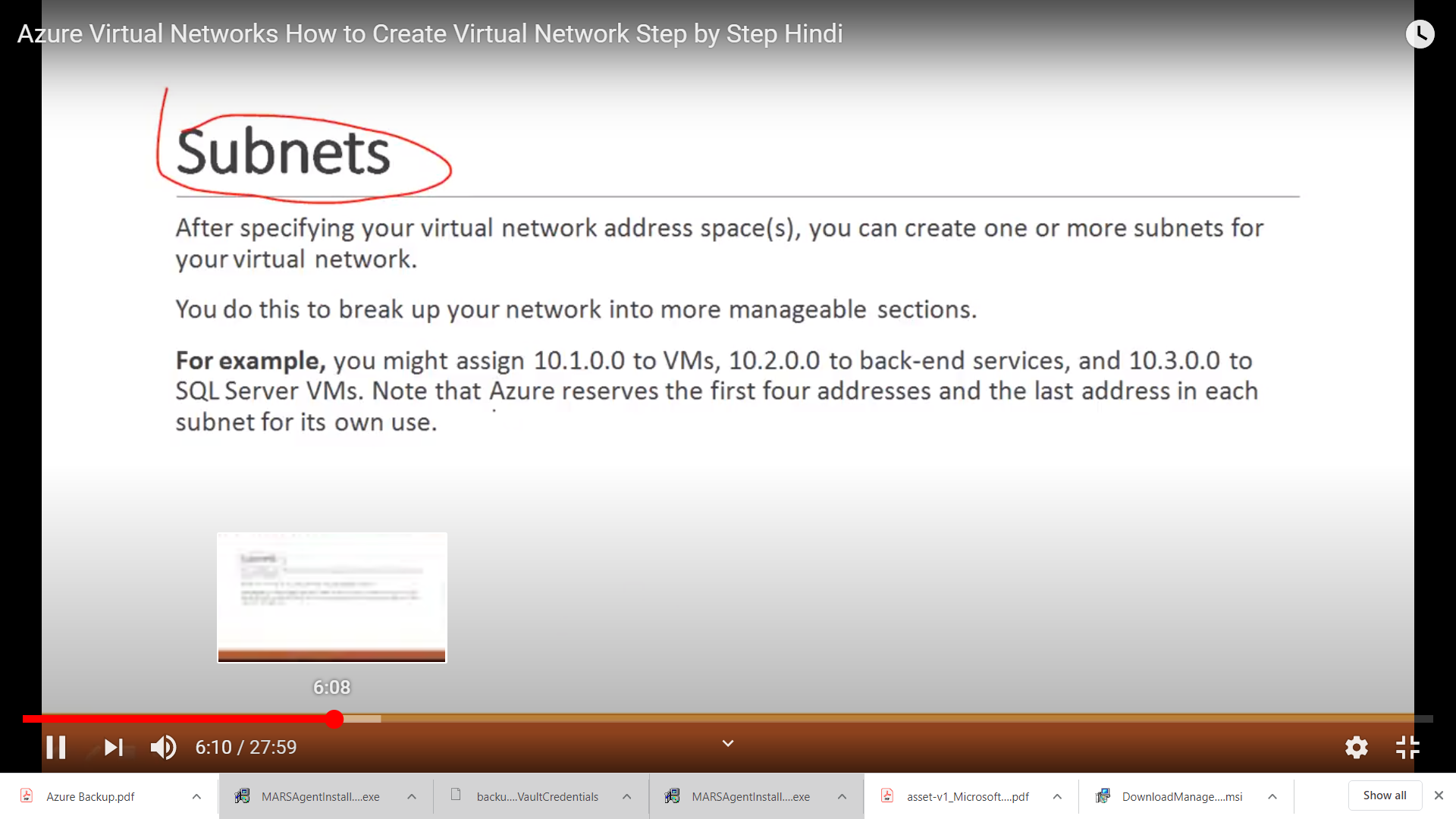
There are certain limits around the number of Azure resources you can deploy. Most Azure networking limits are at the maximum values. However, you can [increase certain networking limits](https://docs.microsoft.com/en-us/azure/azure-portal/supportability/networking-quota-requests) as specified on the [VNet limits page](https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/azure-subscription-service-limits#networking-limits).

**Pricing**

There is no charge for using Azure VNet, it is free of cost. Standard charges are applicable for resources, such as Virtual Machines (VMs) and other products. To learn more, see [VNet pricing](https://azure.microsoft.com/pricing/details/virtual-network/) and the Azure [pricing calculator](https://azure.microsoft.com/pricing/calculator/).







Connectivity in VM’s? in same VNet

Create a Virtual Network

Create two Virtual Machines

Take same Virtual Network in both these VMachines

Take RDP of both machines

Off Windows Firewalls in both the VM’s

Go to cmd, then check ping their private address not public address

We do not need Internet ? because of Microsoft Backbone

There are lines which are used for connectivity

Virtual Networks always work on private IP

Data is encrypted ? Not by default, but it is secured

DEMO

Create VNet

Create VMachine

Take same region, same RG of VNet

Windows RDP

Port take RDP , HTTP only

Next > Networking > Select ur VNet

Take subnet

Create

Create one more VMachine

Take same RG, region

RDP both these VM’s

To check private IP from VM

Select Internet Sign > Network & Sharing > Ethernet > Details > IP 4

Select cd

Ping ip address of other VM

Because ping work on ICMP protocol , whereas windows blocks on firewall

Firewall is enabled

Again on Network & Sharing

* Windows Firewall >Turn Windows Firewall > Turn on Firewall

Do same on other VM too

Now ping

It will work

Establish communication between two Virtual Nets in same region,

Azure supports two type of VNet peering

VNet Peering > connecting VNets within same region

Global VNet Peering > Connecting VNets in different regions

**VNEt to VNet Peering in same region**

Communication thru private network

Useast

VNET2

VNET1

Communication between VNets

Benefits

Network Traffic between peered V Networks is private

Low latency, High Bandwidth connection (Low Response time) Bandwidth depends upon machine configuration

Ability to communicate, transfer data across Azure regions and subscriptions

The traffic between Virtual machines in peered virtual network is routed directly through the Microsoft Backbone infrastructure not thru Gateway or thru public IP

(Data is not coming or going thru Internet)

Peerings are not transitive

VNet1 > VNet2

VNet2> VNet3

VNet1 VNet3

VNet2

In subnets of two VNtes , the IP addresses should not overlap

When we start peering from VNet1 to VNet 2, we see Initiated in VNet1

When we start peering from VNet2 to VNet 1, we see connected in VNet2

After that we will see Connected in VNet1 too

Firewalls should be off in VM’s

**DEMO**

VNET to VNET Peering in same region

1. Create first V Networks > 10.0.0.0/16

Subnet > VNetsubnet1 > 10.0.0.0/24

Keep others same

1. Create second V Networks > 192.168.0.0/16

Subnet > VNetsubnet2 > 192.168.1.0/24

Keep others same

KEEP THEM ON SAME REGION , SAME RESOURCE GROUP

1. Now add V Machine1 in First VNet
2. Now add V Machine2 in Second VNet
3. NOW WE HAVE TO ADD VNet PEERING
4. For that got to Virtual Networks > Go to VNet1 > Peerings> Add > Give peering a name , select second VNet Name > OK
5. Do same on other VNet2
6. For that got to Virtual Networks > Go to VNet2 > Peerings> Add > Give peering a name , select first VNet Name > OK
7. Open both the VM’s
8. Ping them after disabling Firewall

**VNet to VNet Connection in different regions**

Us-east

Us-west

Virtual Network Gateway

Virtual Network Gateway

TUNNEL

TUNNEL

Microsoft backbone infrasyructure



NO

NO NEED OF INTERNET OR PUBLIC ADDRESS TO COMMUNICATE

DATA IS NOT ENCRYPTED BY DEFUALT, BUT DATA IS SECURED

PEERING WILL NOT WORK HERE

Now we need to create Virtual Network Gateway on both the VNets

Steps :

Create VNet1 in one region east-use > 10.0.0.0/16 , resource group , new , subnet > subnet1 > 10.0.0.0/24

Create VNet2 in other region east-use > 192.168.0.0/16 , resource group , same , subnet > subnet2 > 192.168.0.0/24

Create VNetGateway for first VNet1, Gateway1

Gateway Type > VPN

VPN Type > Root based

Choose Virtual Network > VNet1, in same region

Public Ip > Create New

Add one more gateway for VNet2 . Do Same on for VNet2

Create VM1 on VNet1

Create VM2 on VNet2

Once Gateways are created , now go to Gateway, there I an option Connection >Add > Select type VNet to VNet , First VNetGateway select your VNet1, in other Select VNet2

Do same for other Gateway

Shared Key has to be same on both