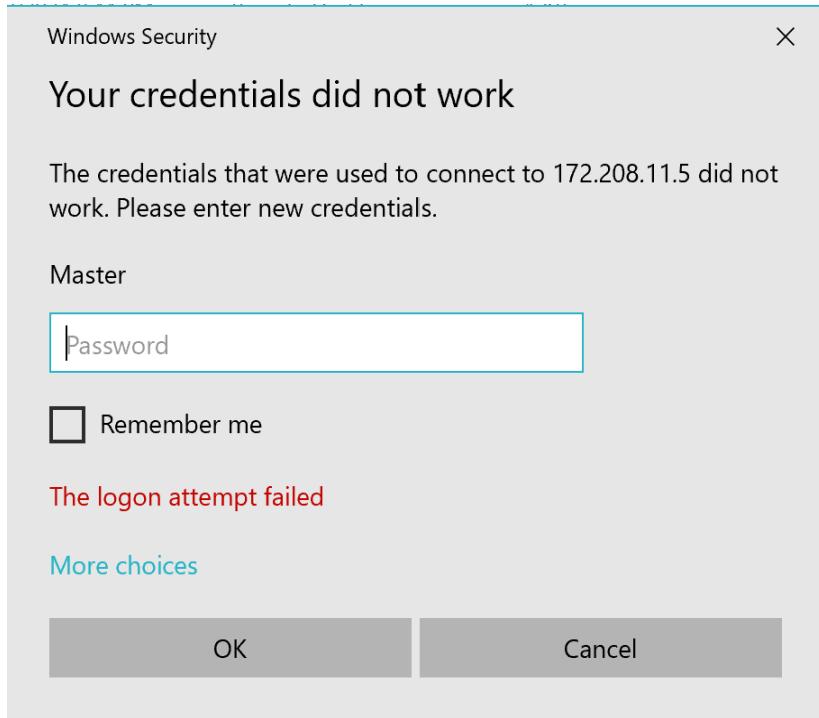


Today I'm creating 2 virtual machines on Azure - a Windows 10 PC and a Windows server 2019. After creating them I'll login to the server and install the Active Directory Domain Services role. Then I'll put both machines on the same vnet so they can communicate with each other & join them on the same Domain in Active Directory. I could've configured them to be on the same vnet from the start but I wanted to demonstrate the process of moving a VM to another vnet.

Azure allows for a 30-day free trial and a \$200 credit for signing up. After signing up it brought me to the main dashboard where I clicked on virtual machines and then clicked create Azure virtual machine. There are many options to choose from but I stuck with the simple & free options for both of my machines. During the process, it prompted me to create an Administrator account and password to be able to log in.

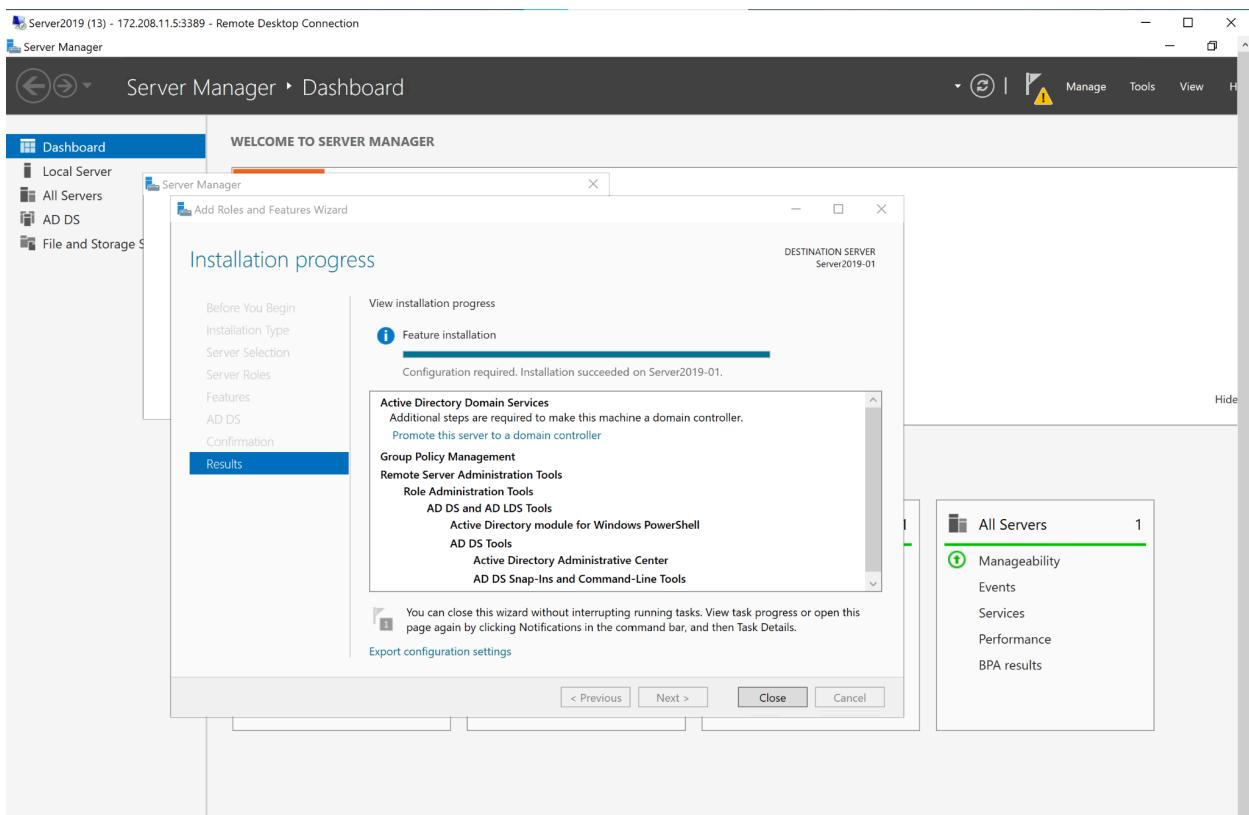
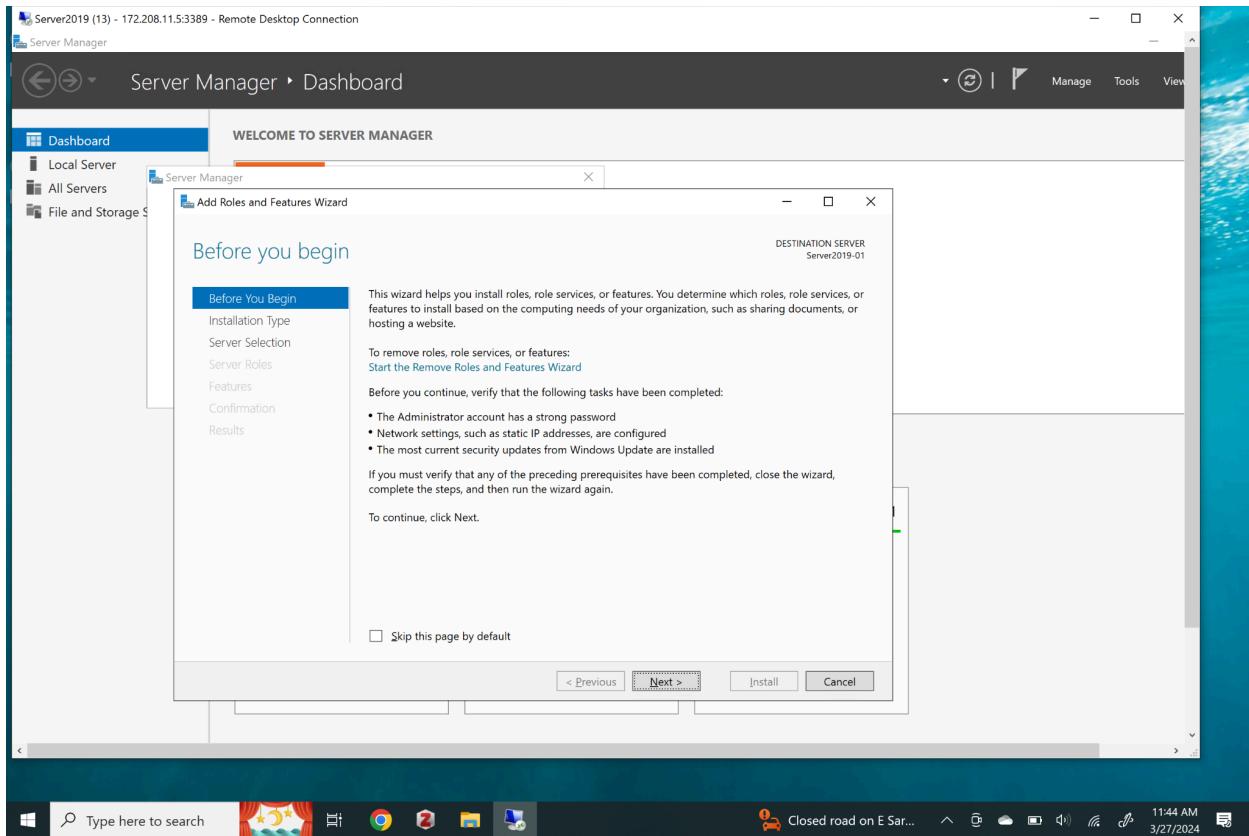
An important aspect is to have a password that satisfies the full requirements. Passwords must have 3 of the following: 1 lower case character, 1 upper case character, 1 number, and 1 special character, AND the value must be between 12 and 123 characters long. For some reason, 12 characters were fine and I was able to login to my Windows 10 VM with no problem, however the same wasn't true for the Server. Despite using the same password for both it would not let me log in. I ended up troubleshooting for hours and double-checking all network and security configurations. I even changed the password multiple times and dug deep

into Microsoft documentation on Azure.



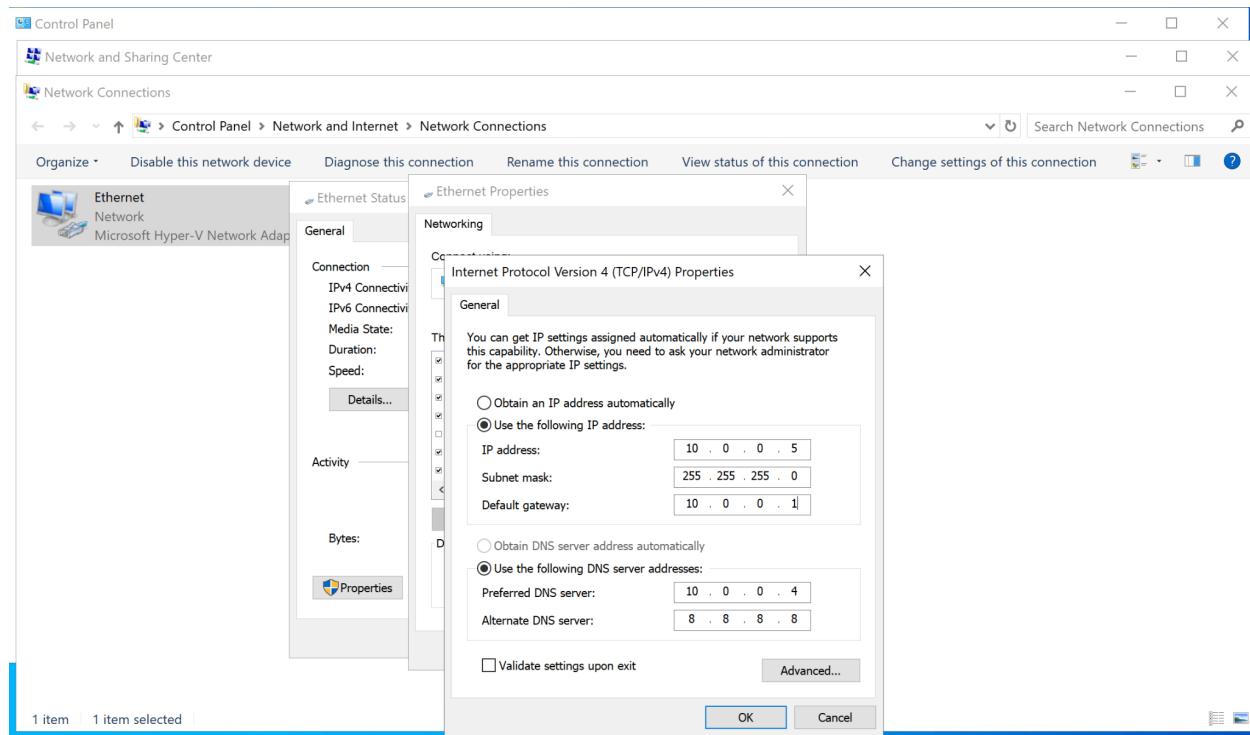
I was about to delete the VM and create a new one until I finally figured out that I had to use 15 characters to be able to log in to the Server. I thought I was going crazy because I could not find any other issues, so that is a mistake I'll be sure never to make again. [This video here](#) is what helped me finally figure it out.

OK, now that I was passed that issue, I can log in to the server and configure it for active directory domain services. I did this simply by going into the server manager, clicking on the manage tab on the top right, and choosing “Add roles and features”. I chose the installation type “role-based or feature-based installation”. From here it brought me to server roles, there was a big list of roles to choose from. I chose “Active Directory Domain Services” and gave the name domain.homelab, then installed it. After installation, it logged me out and restarted the computer. The pictures below show the process:



In this next part, I attempted to join the Windows 10 VM to the domain of the Windows Server 2019. I took the following steps before I realized that the VMs are on 2 different vnets and cannot communicate. The first thing I did was open the command prompt and attempt to ping the server to see if communication could be established. They would not ping each other.

I then went to my control panel -> view network status and tasks -> change adaptive settings -> double click on ethernet -> click properties ->click internet protocol version 4 (TCP/IPv4) -> properties -> chose “use the following DNS server address”, then I entered the private IPV4 address for the server(10.0.0.4). I also selected “use the following IP address” and entered IP address 10.0.0.5, subnet mask 255.255.255.0, and default gateway 10.0.0.1 for my Windows 10 machine.

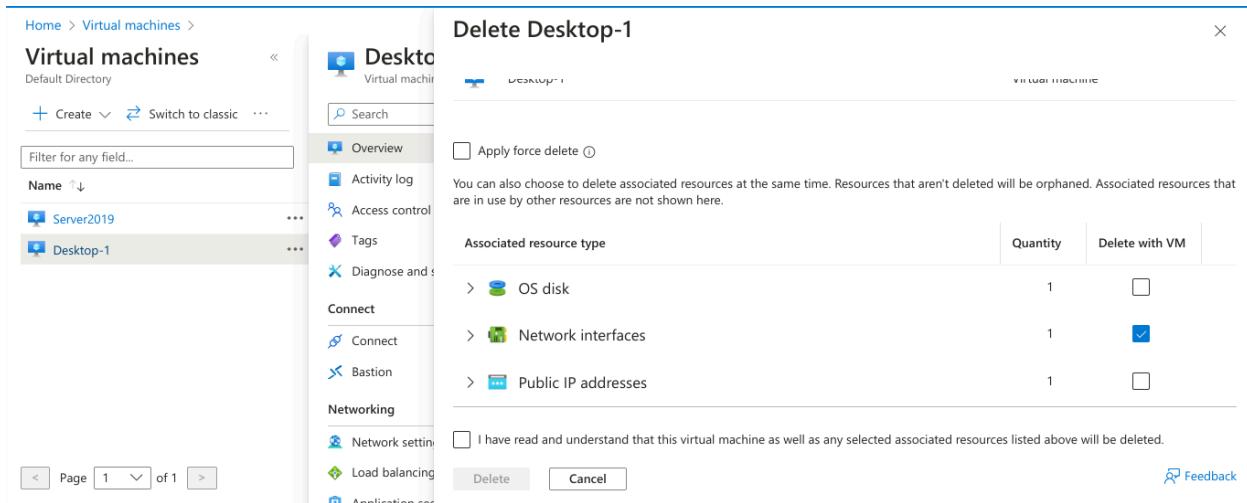


After pressing enter it kicked me out of the Windows 10 VM. This is because I changed the private IP address from 10.0.0.4 to 10.0.0.5. In the Azure dashboard, I then navigated to network settings -> windows10220_z2 | IP configurations -> IP configurations. There I clicked on ipconfig and an option popped up to edit the private IP address.

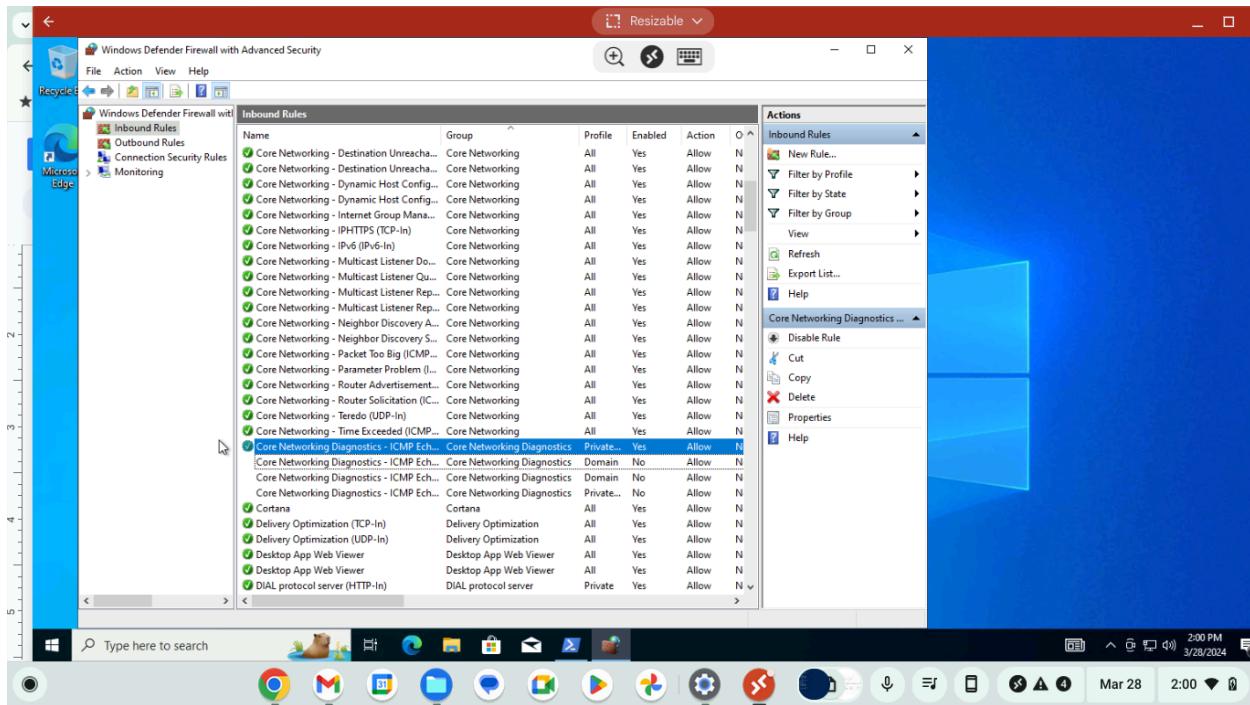
I changed the private IP address to static and entered 10.0.0.5. This allowed me to log in again to the PC, however the 2 VMs were still not communicating with each other. The reason for this was because they are on 2 different isolated virtual networks. If I wanted them to communicate I was considering 2 options, 1: Switch one VM over to the other vnet by partially deleting it (probably the Windows 10). 2: peer them so they can communicate, a feature in Azure that allows for vnets to establish a network connection between them.

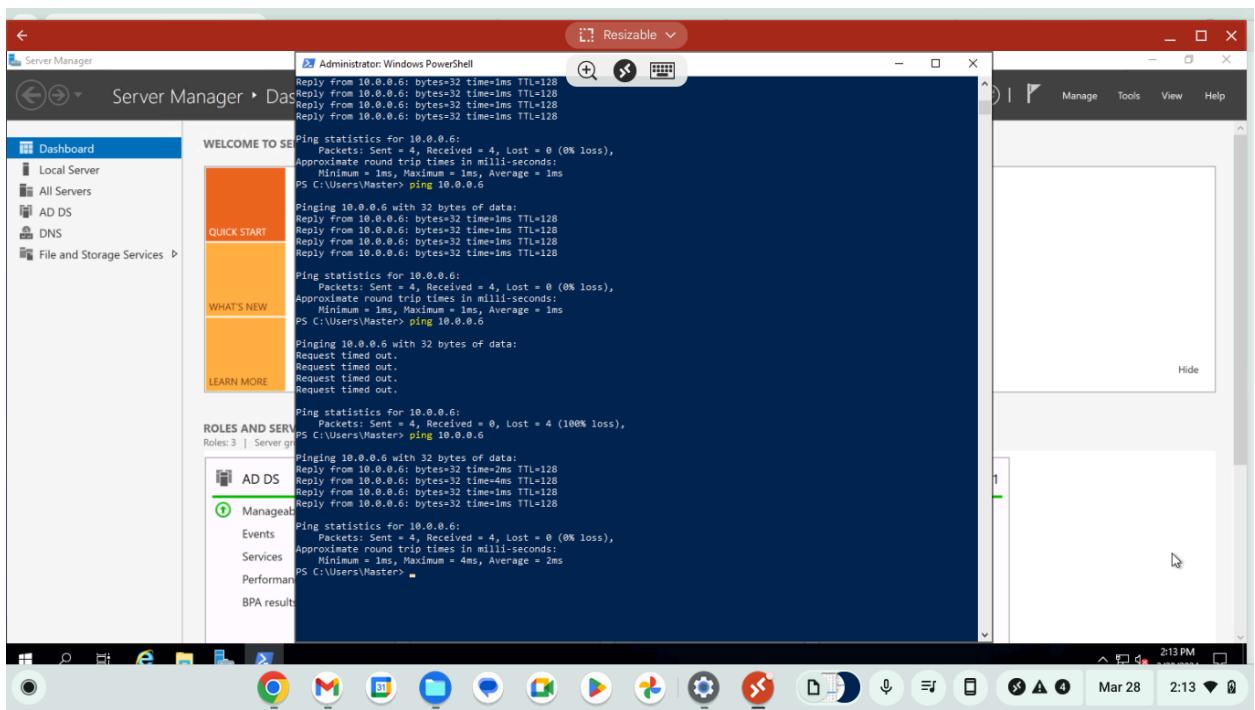
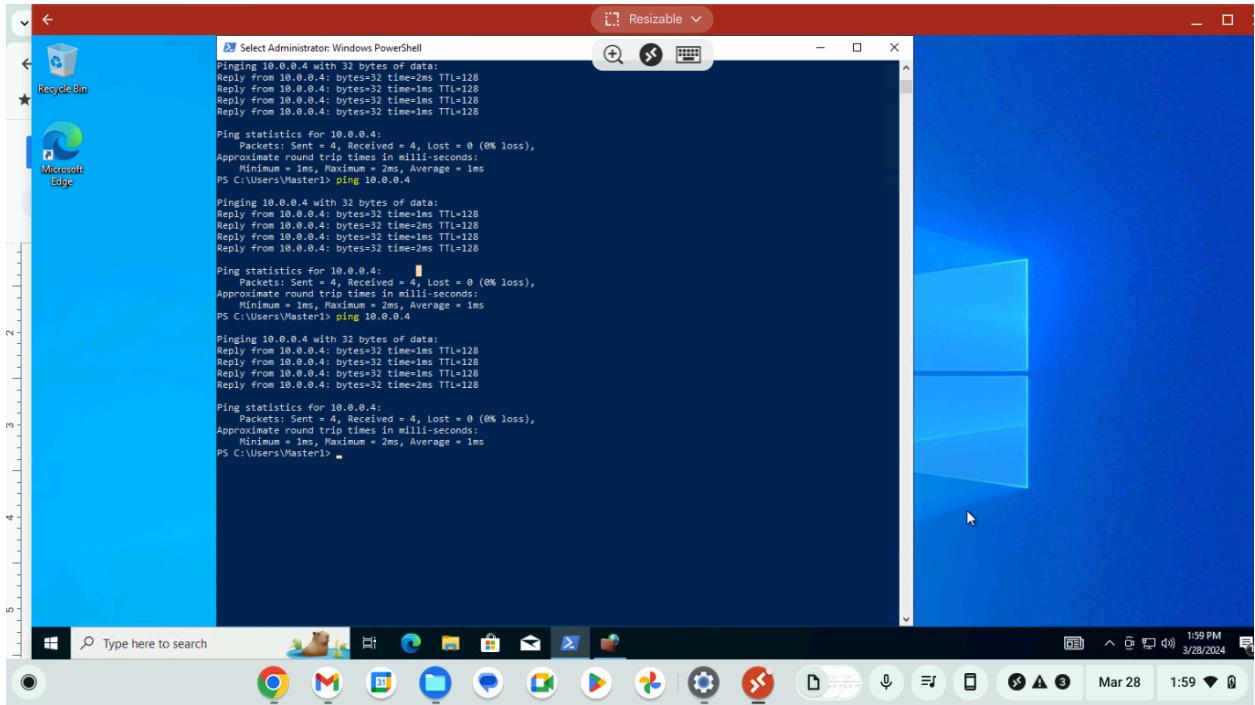
I chose option one and placed them on the same vnet. This solution included partially deleting the Windows 10 VM. I did this by [following the instructions in this video](#). I went to the VM I wanted to move, clicked delete, then selected delete the network interfaces and unselected the disk. Then I hit delete. This deleted all the resources associated with the VM

(NIC & vnet) except for the disk image itself.

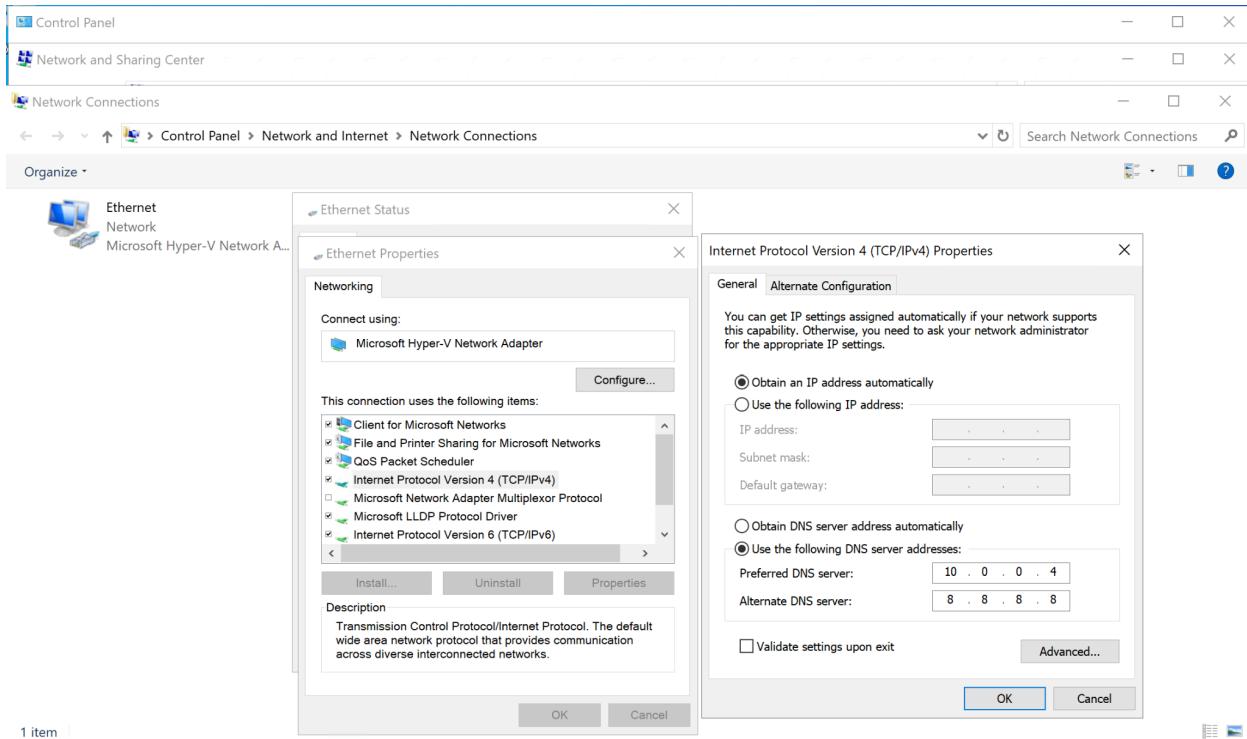


Afterwards, I went to the resource group where the VM is located and selected osdisk -> then selected create VM. From there I filled out everything as before EXCEPT - I made sure to select the vnet I wanted to move the Windows 10 vm to(server2019-vnet/default). I then clicked create and sure enough they are now on the same vnet. I activated both the server and Windows 10 VM and pinged both just to be sure. I also made sure to adjust the firewall settings on the Windows 10 VM to allow inbound and outbound ICMP echo requests over IPV4 private. I did this by going into the advanced settings in Windows Defender. As you can see in the images below I can now ping both the Windows 10 VM (10.0.0.6) & the Windows 2019 server (10.0.0.4).



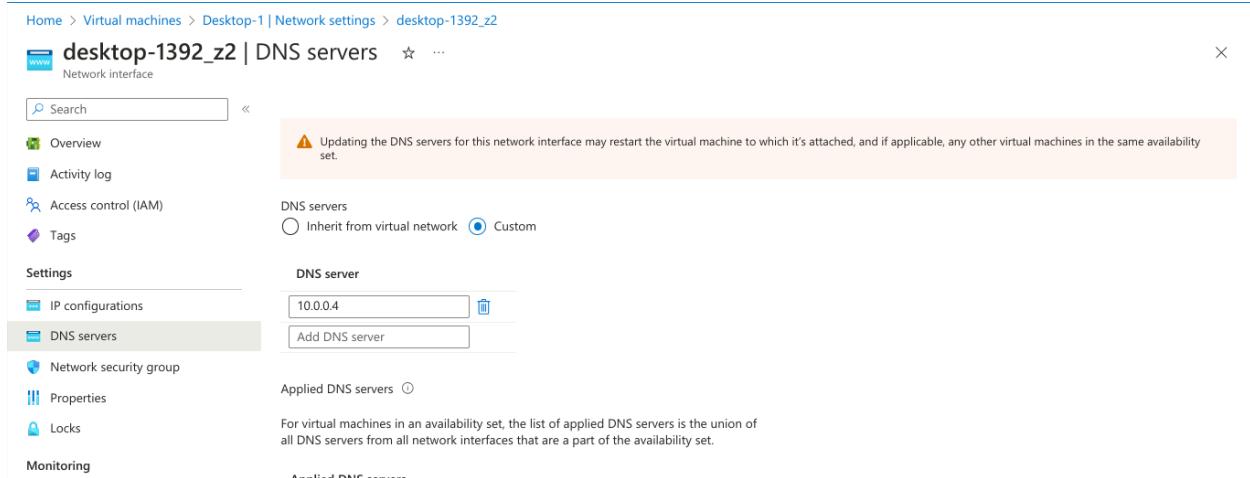


Finally, I was able to join the Windows 10 VM to the domain of the 2019 server. After going into the pc I followed the same prior steps: control panel -> view network status and tasks -> change adaptive settings -> double click on ethernet -> click properties ->click internet protocol version 4 (TCP/IPv4) -> properties -> choose “use the following DNS server address”, then entered the private IPV4 address for the server (10.0.0.4).

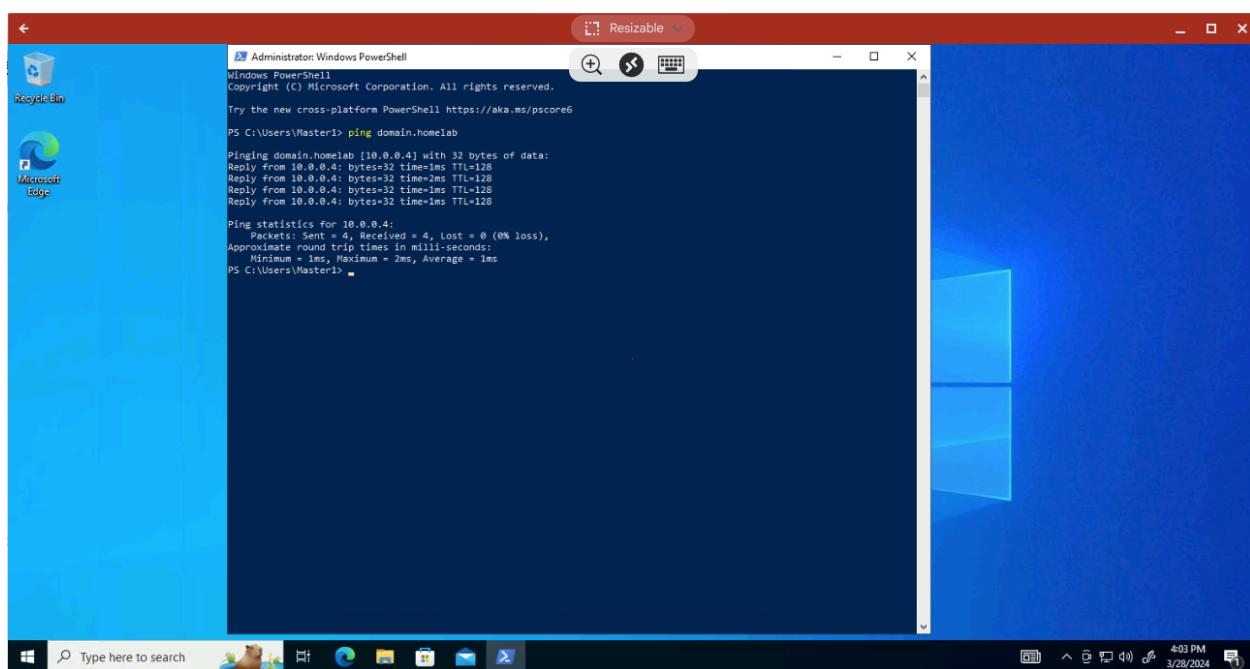


I also had to use the dashboard in Azure to go into the NIC for the Windows VM and go to overview -> networking -> click on network interface desktop -> settings on the left pane -> DNS servers -> select custom -> and add the private IP of the server 10.0.0.4. I then restarted the

VM and it took a while but I was able to log back in and ping domain.homelab.



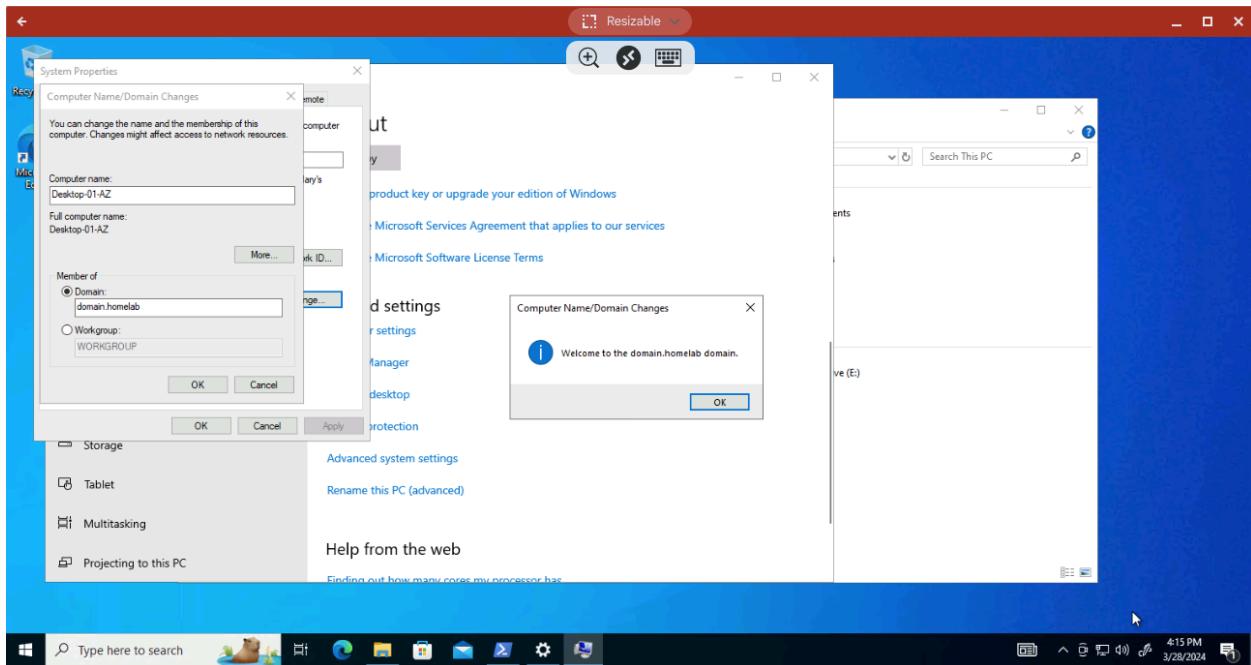
The screenshot shows the Azure portal interface for managing a virtual machine named "desktop-1392_z2". The "Network interface" tab is selected. In the "DNS servers" section, there is a dropdown menu with two options: "Inherit from virtual network" (unchecked) and "Custom" (checked). Below the dropdown is a text input field containing "10.0.0.4". A button labeled "Add DNS server" is visible. A warning message at the top right states: "Updating the DNS servers for this network interface may restart the virtual machine to which it's attached, and if applicable, any other virtual machines in the same availability set." On the left sidebar, under the "Settings" section, the "DNS servers" option is highlighted. Other settings like IP configurations and access control are also listed.



The screenshot shows a Windows 10 desktop environment. A PowerShell window titled "Administrator: Windows PowerShell" is open, showing the command "ping domain.homelab" and its output. The output indicates four successful ping requests to the IP address 10.0.0.4, with round-trip times ranging from 1ms to 2ms. The desktop background is blue, and the taskbar at the bottom shows icons for File Explorer, Edge, and other system tools. The status bar at the bottom right shows the date and time as 3/28/2024 4:03 PM.

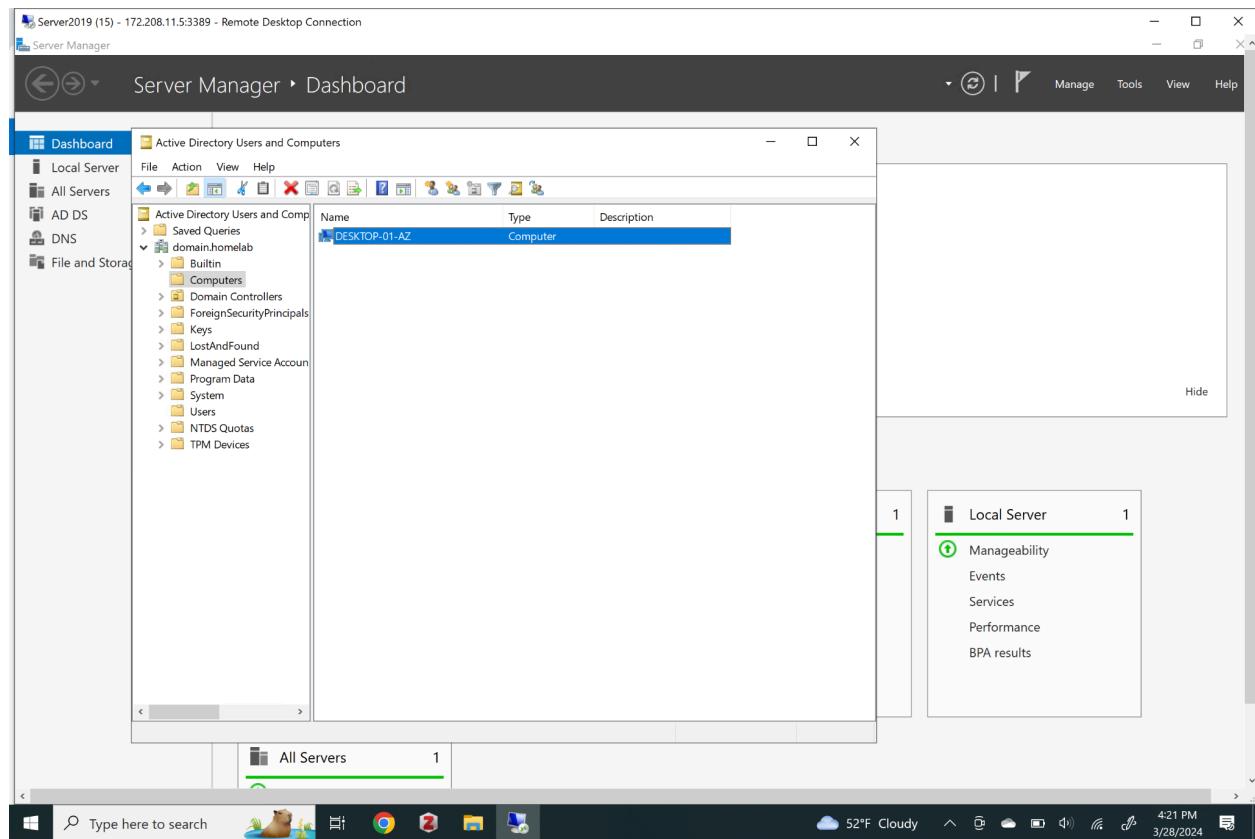
The next step is to join the Windows machine to the domain. I went to File Explorer -> right click this PC -> properties -> rename this PC (advanced) -> change -> select member of domain -> type domain.homelab -> click ok - >log in using server credentials -> computer will

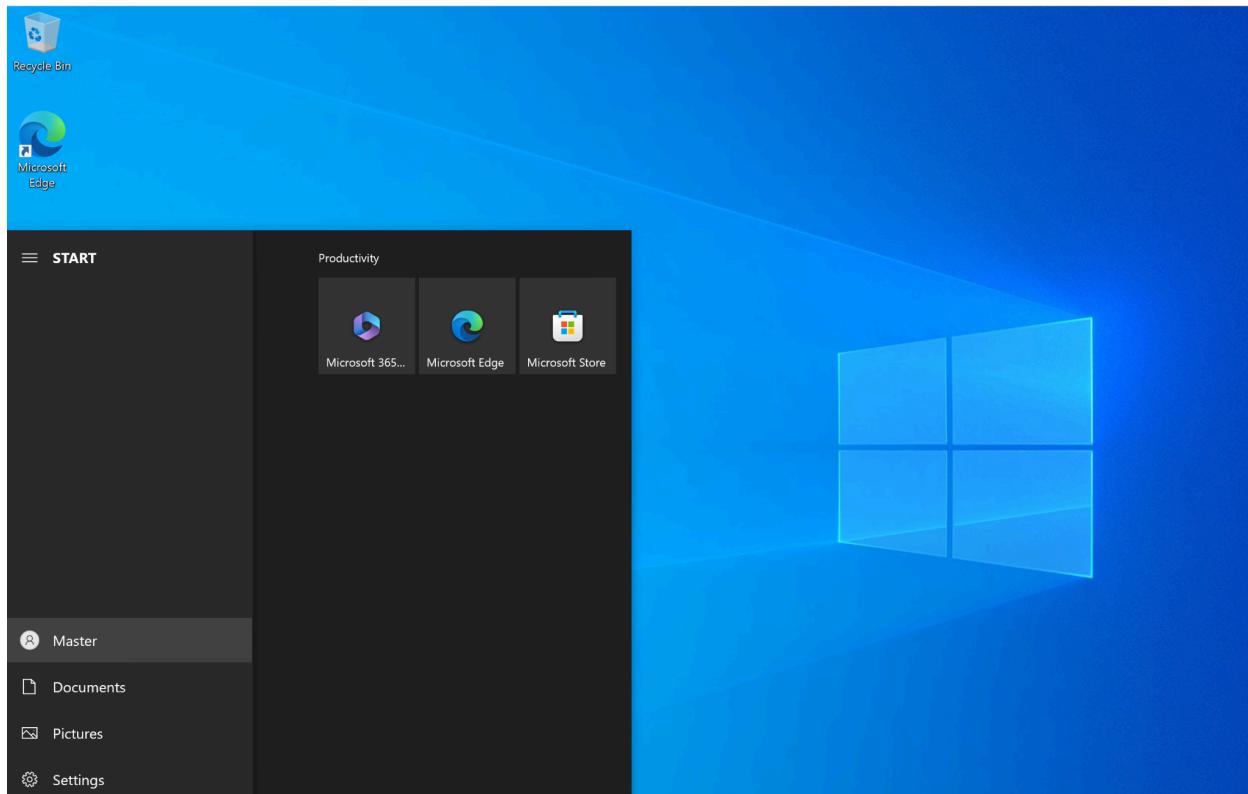
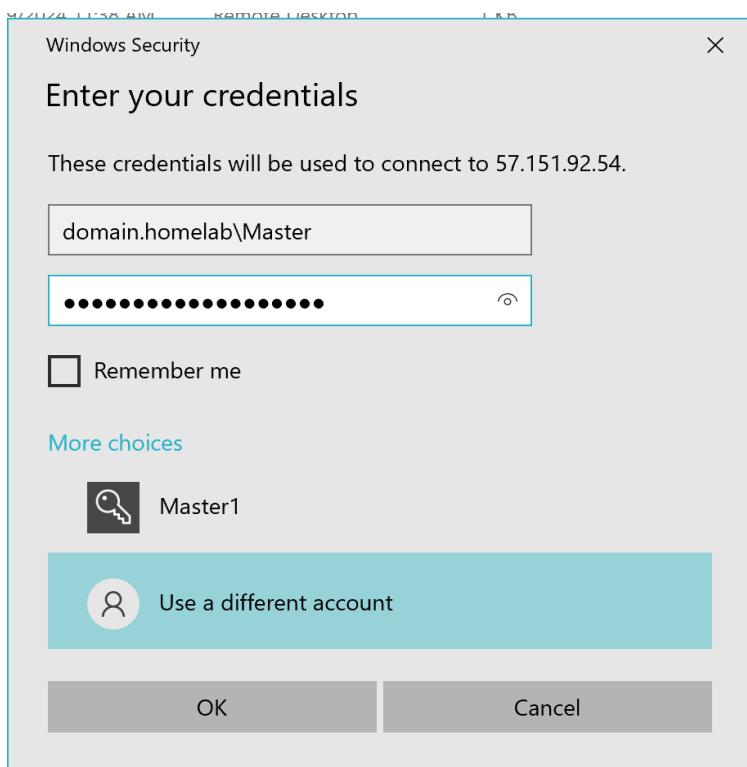
restart after login.



After restarting I checked the server, under active directory users and computers -> computers -> and there it is as shown below. I can now log in to the PC using the credentials from my server with the domain name Master using “domain.homelab\Master”, this is the power user

who has access to the active directory from the Windows 10 machine.





This concludes part 1 of the Azure Active Directory project. In part 2 I'll be creating a new user with Powershell, Giving that user permission to use RDP, then creating and mapping a folder.