**Home Lab Active Directory**

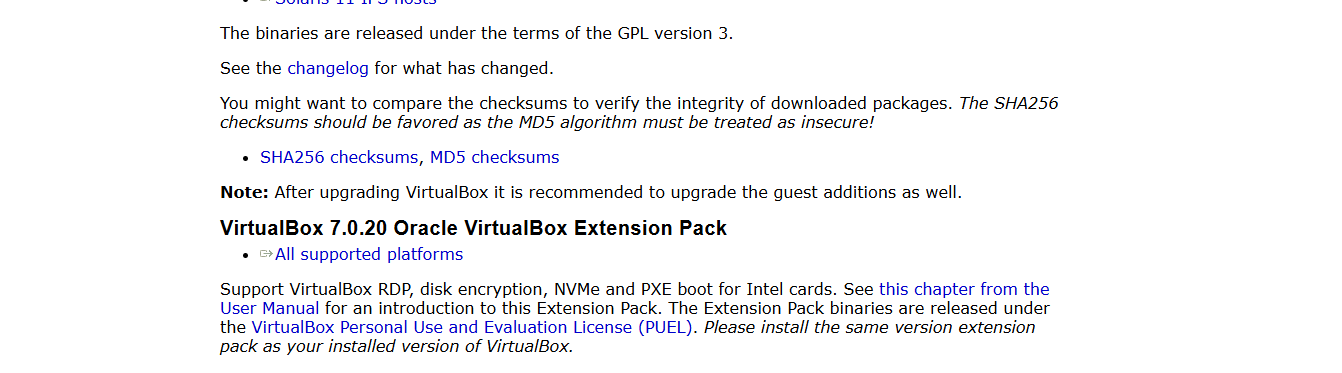
Virtural Box/ Powershell

Before we start this lab, we’ll have to understand what each part means.

**Active directory** is a directory service developed by Microsoft that stores information about objects like users, computers, and groups, and allows administrators to manage permissions and access to network resources. It acts as a centralized authentication and authorization system within a Windows domain network.

**Active Directory Domain Services** is a core component of Microsoft's Active Directory that provides a centralized system for managing users, computers, and other resources in a network. It enables secure authentication and authorization, allowing administrators to implement and enforce security policies, and manage access to resources across a Windows domain.

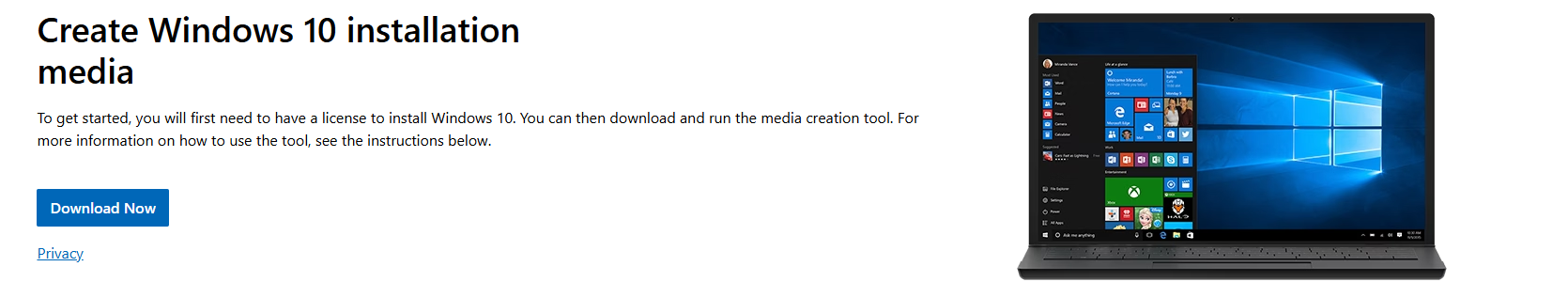
**Domain Controllers** are servers in a Windows domain network that run Active Directory Domain Services (AD DS), providing authentication, authorization, and directory services for users and devices. They manage security, store directory data, and respond to authentication requests such as user logins, ensuring centralized control over network resources.

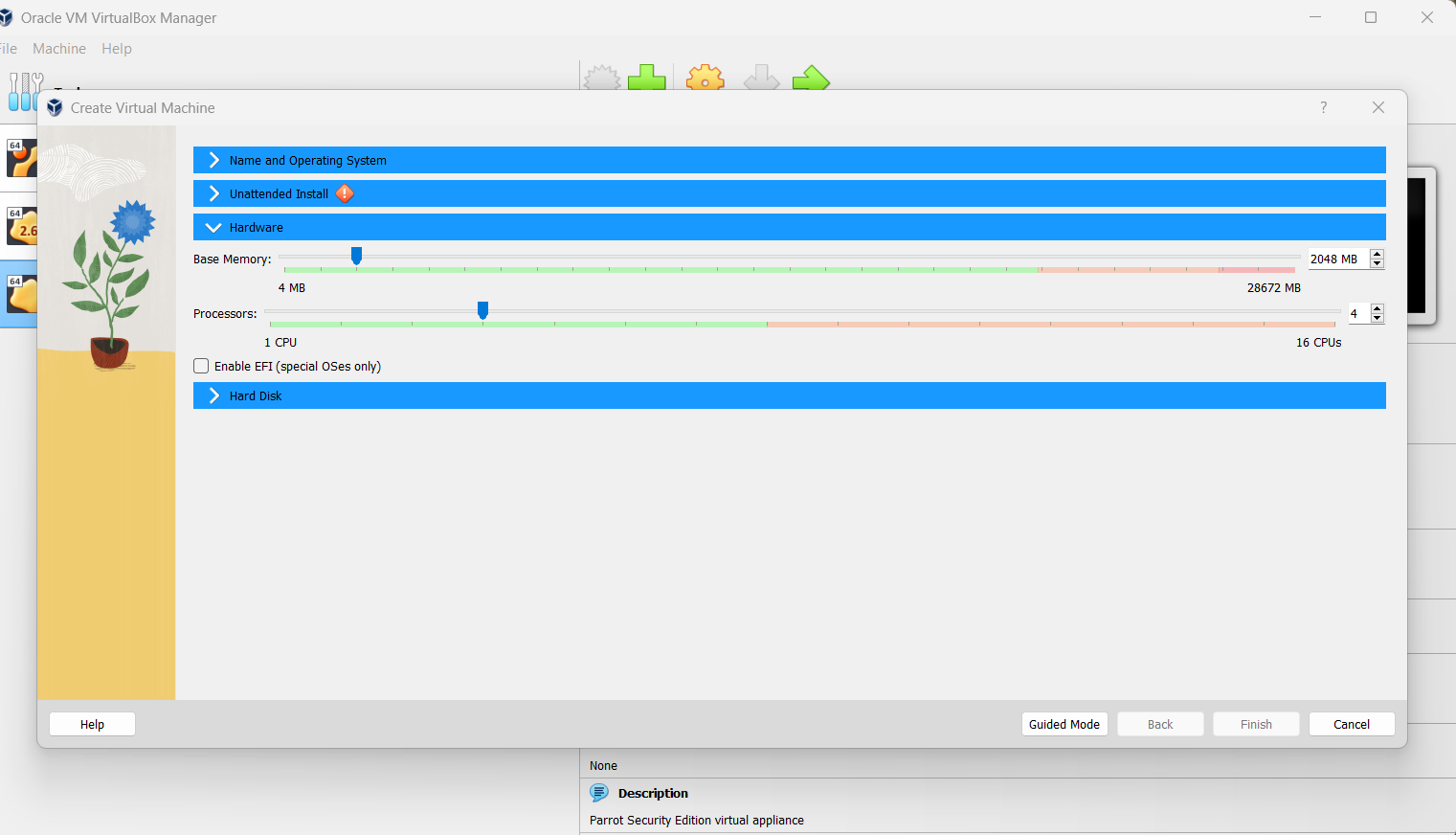
To start this lab we’ll need Virtual Box Downloaded. You can find it here [Downloads – Oracle VM VirtualBox](https://www.virtualbox.org/wiki/Downloads). After downloading be sure to also install the **Virtual Box Extension Pack. **

This will allow for more complex virtualization that will be useful when setting up virtual machines.

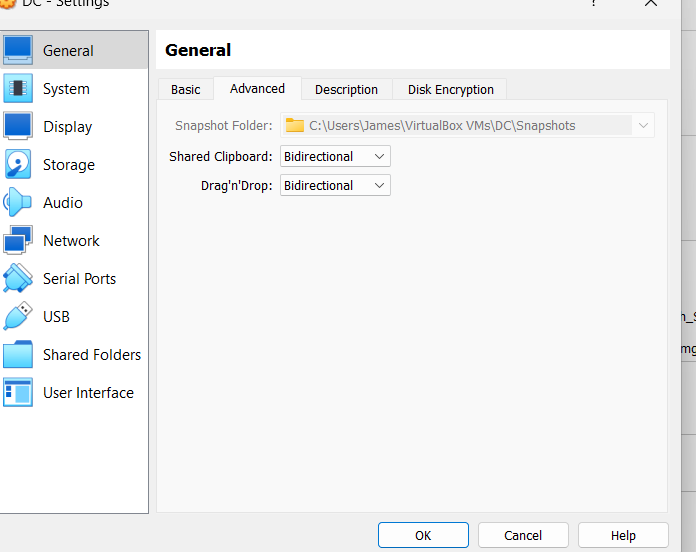
Next navigate to [Windows Server 2019 | Microsoft Evaluation Center](https://www.microsoft.com/en-us/evalcenter/download-windows-server-2019) and download the 64 bit edition. 

For the other windows computer download the Windows 10 ISO. [Download Windows 10 (microsoft.com)](https://www.microsoft.com/en-us/software-download/windows10)



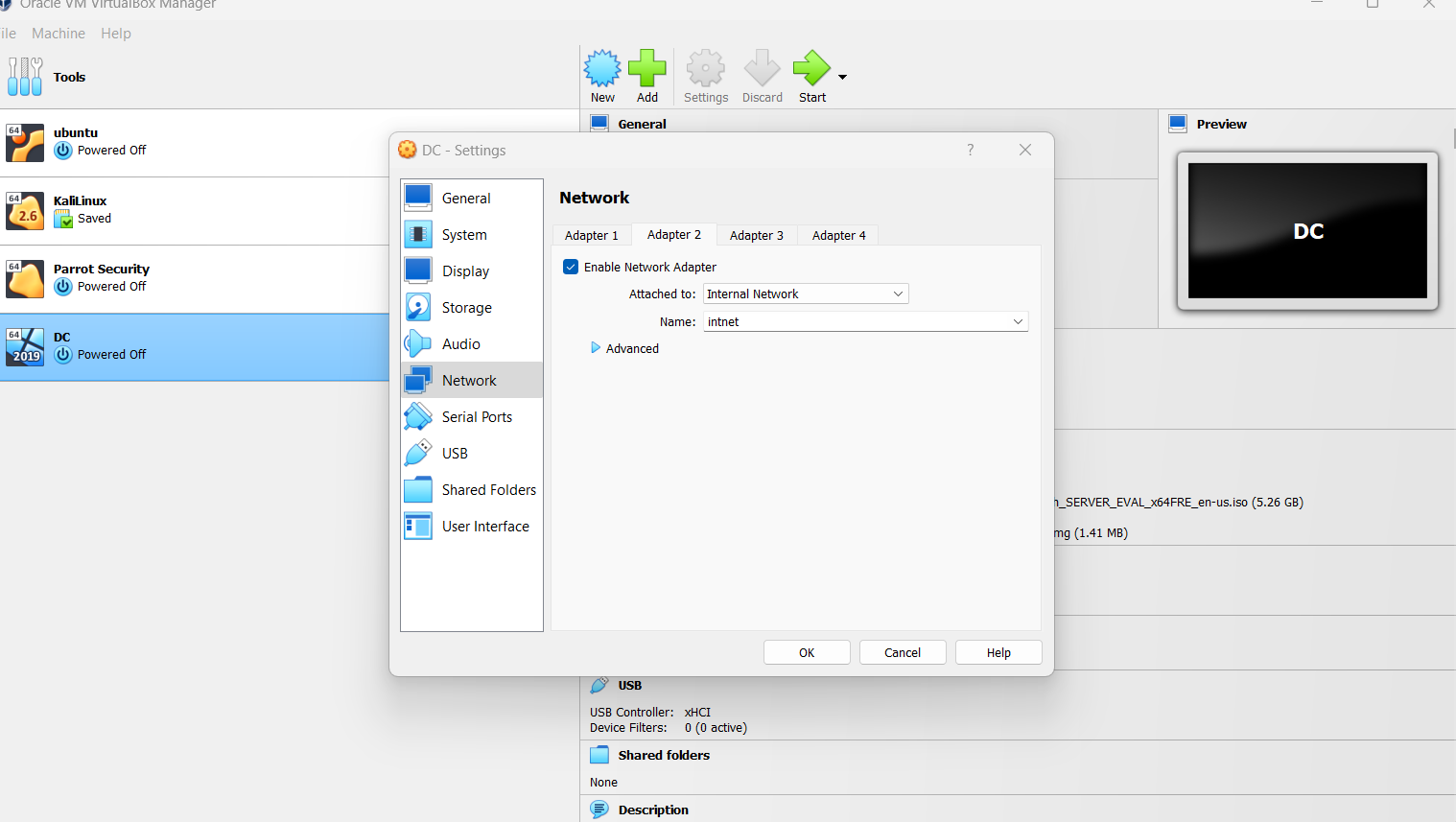
When Setting up your Virtual Machine, name the domain controller DC using 2048 mb of Ram and 4 CPU’s.

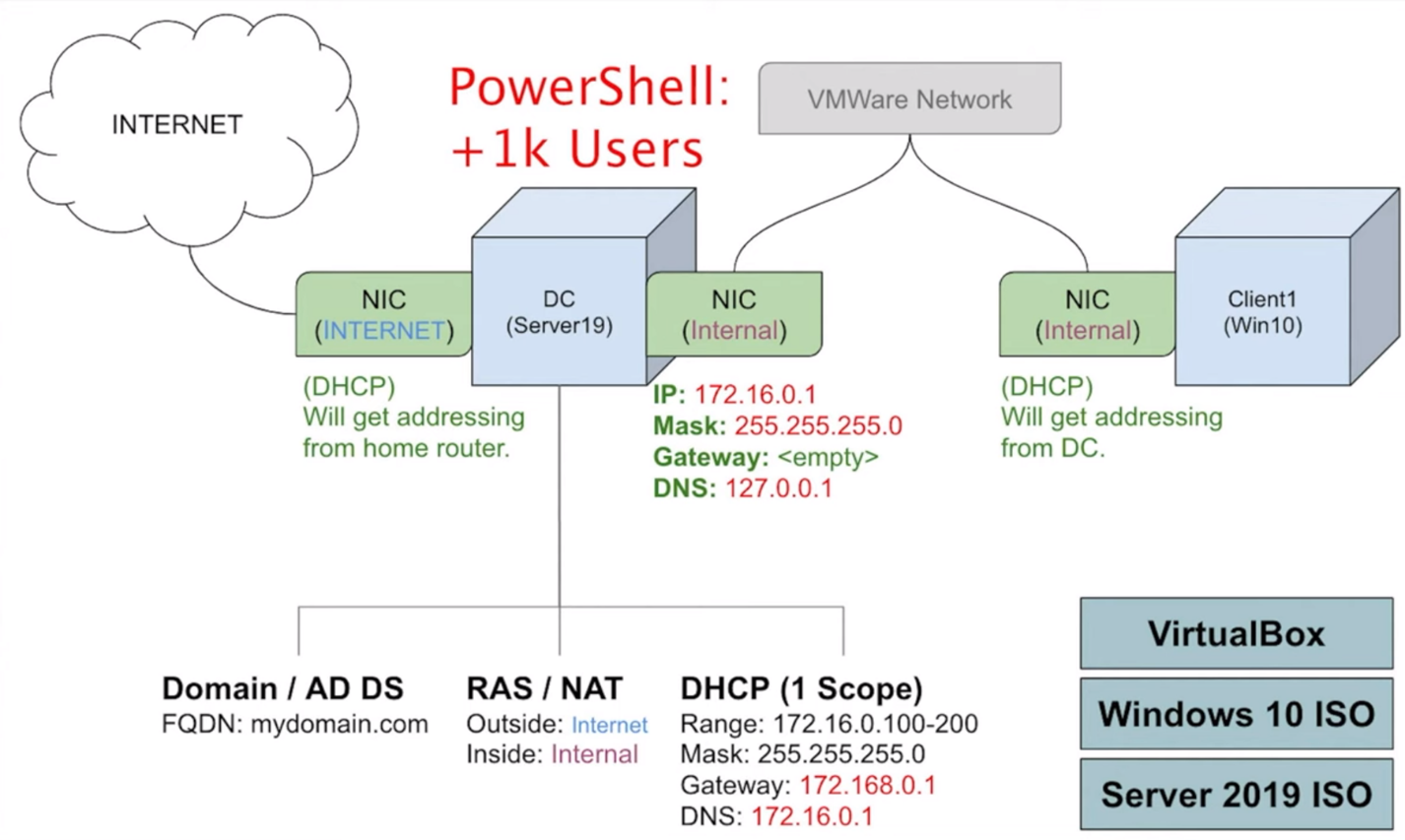
In the settings under the general tab under “advanced” select “bidirectional” for clipboard and drag’n’drop.



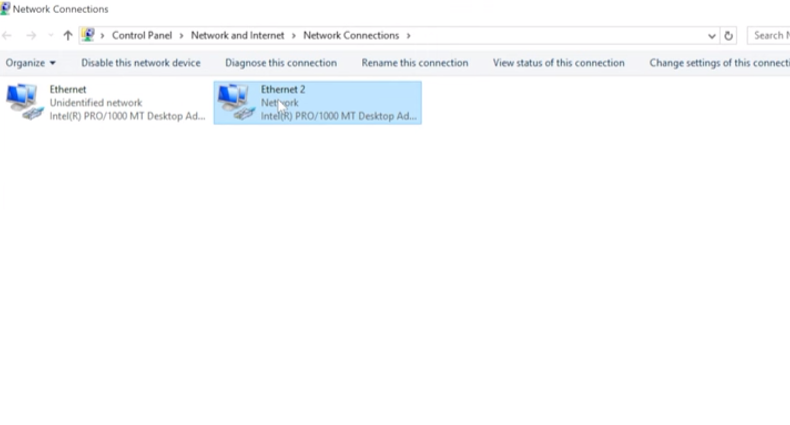
Under the Network tab of the DC VM, Select adapter one as NAT. This allows virtual machines to access external networks (like the internet) using the host machine's IP address. The VM is effectively "hidden" behind the host machine, and all outbound traffic from the VM appears to come from the host. This setup provides internet access to the VM without exposing it directly to the external network, offering basic security and convenience for general usage.

On adapter 2 select enable and select “internal network” This creates a private network that is isolated from the external network and the host machine. Virtual machines connected to this adapter can communicate with each other but do not have access to the outside world or the host machine. This setup is useful for creating isolated testing environments or private networks within a virtualized environment where you want VMs to interact only with each other. After configuration well be able to use the DC to forward traffic to the internet.



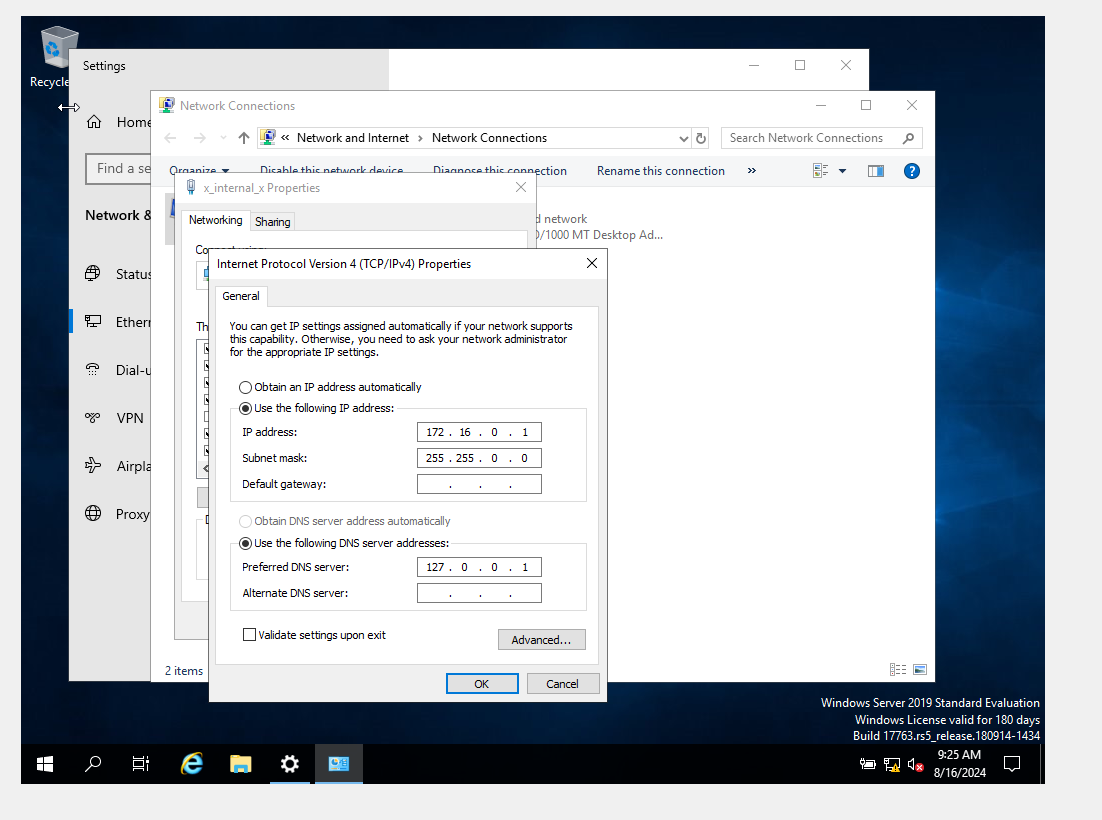
We can refer to Steven Sampson’s high level overview of the process. [www.linkedin.com/in/steven-sampson-70a05776](http://www.linkedin.com/in/steven-sampson-70a05776)

As mentioned earlier we’ll be forwarding all traffic on the private network through the Domain Controller with DHCP set up to assign Ips to our private network.

Once in the DC VM navigate to the adapter options. There you will see 2 ethernet options

Right click into properties click status on the right ethernet and select details. Usually on the “sent and received” numbers on each side indicates packets successfully sent and received meaning an active home connection. Further investigation into details. A home IP address will appear. Once identified rename “\_Internet\_.”

On the other internet following the same process, we can see a DHCP IP address. This meaning we could not successfully get an IP address. Right click on the adapter and select properties. Navigate to “Internet protocol Version 4 (TCP/IPv4)” and double click it. Here we will be assigning the IP for the Internal NIC.



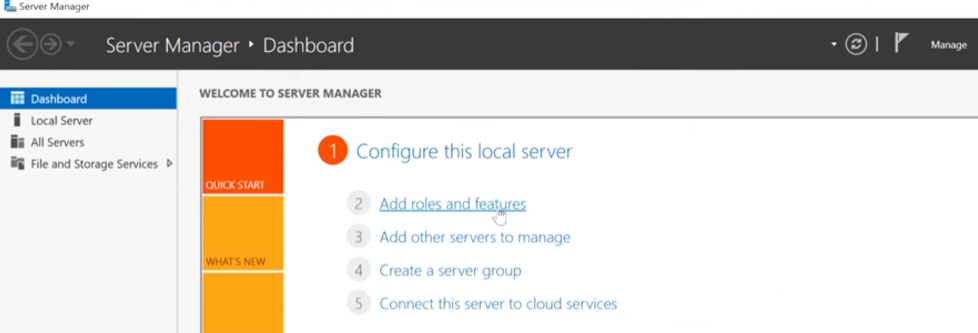
Input 172.16.0.1

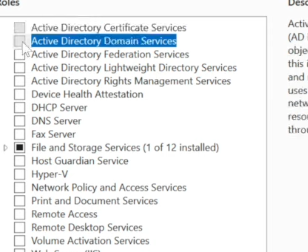
Set the subnet mask “255.255.255.0”

For the DNS server use the loop back address of 127.0.0.1 Since once AD is installs DNS DC will be the DNS server.

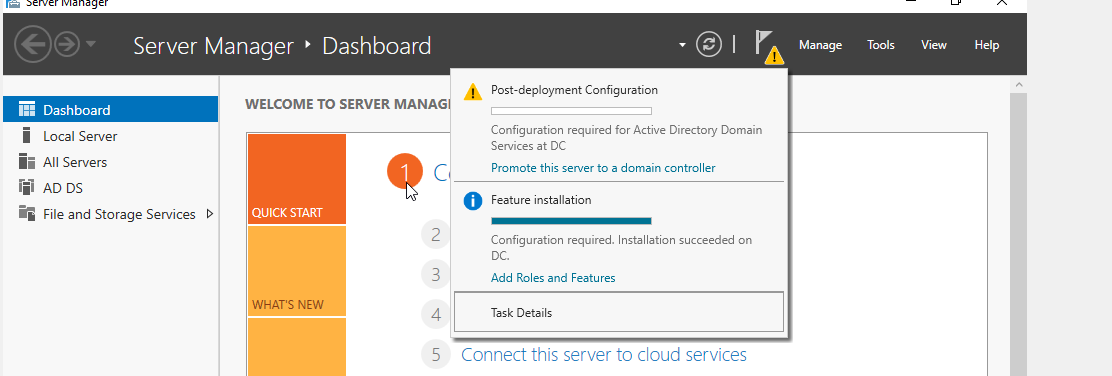
Next Right click the start menu and then system. Click on “rename this pc” and enter DC.

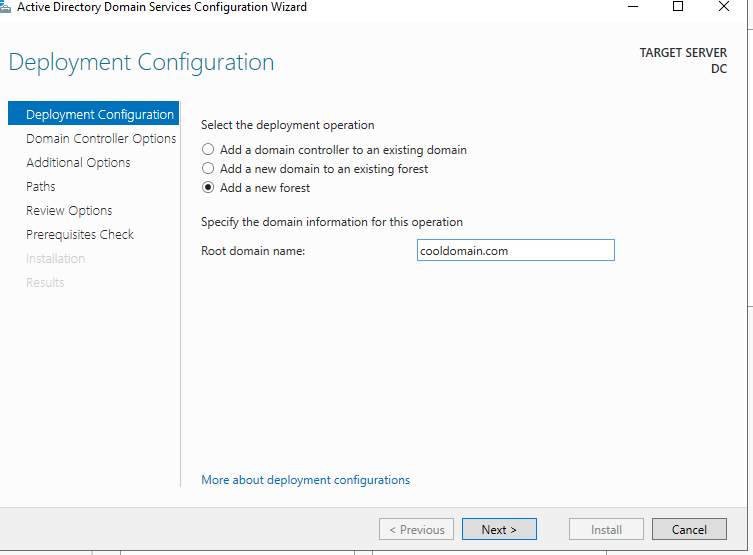
After restarting we will install active directory.

On server manager dashboard click on “add roles and features” 

Click next until you reach server roles. There select “Active Directory Domain Services.”

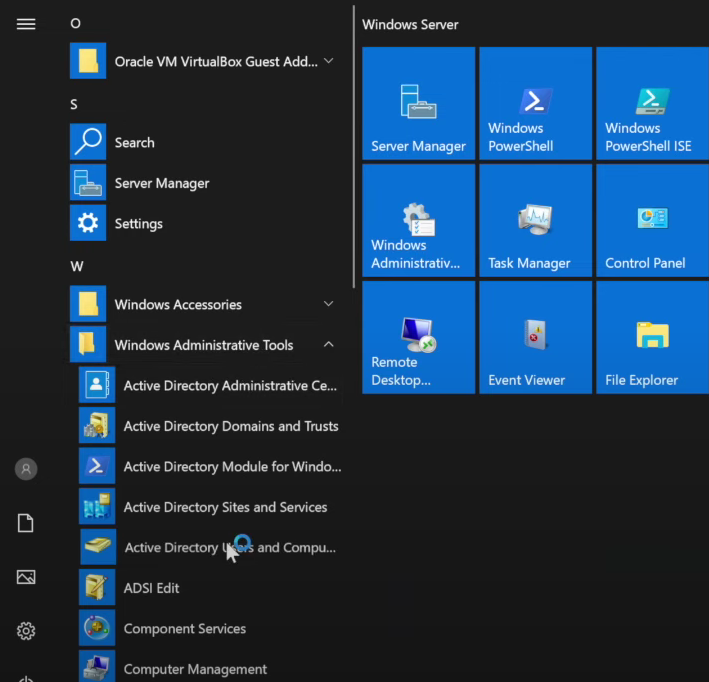
Click “add features” and then next until install.

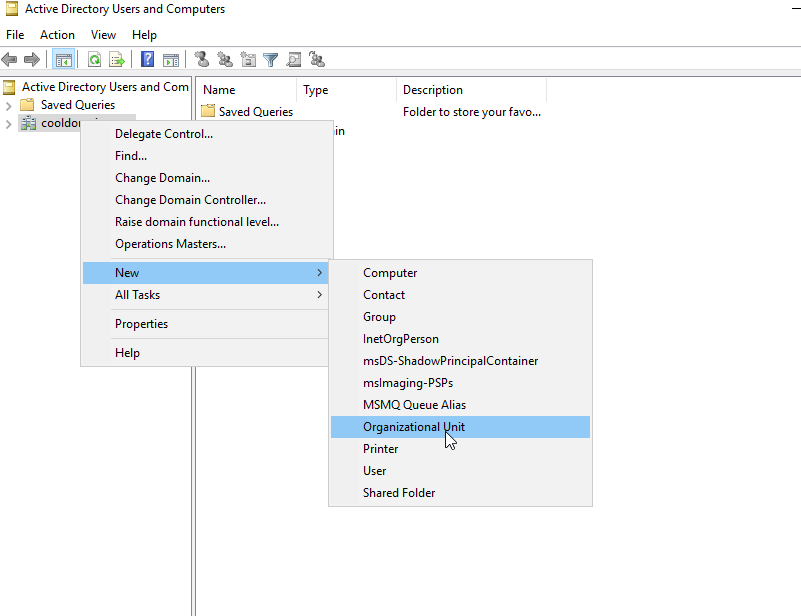
Once that’s completed on the top right a yellow notification should appear. Here you click and then click “promote this server to a domain controller.” 

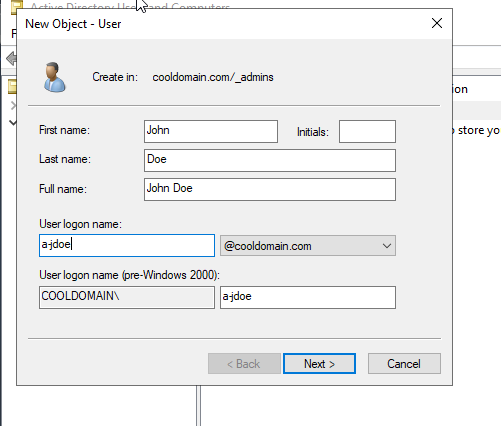
This will lead you to a deployment configuration, there select “add a new forest.” Here you can name your domain. I named mine “cooldomain.com” 

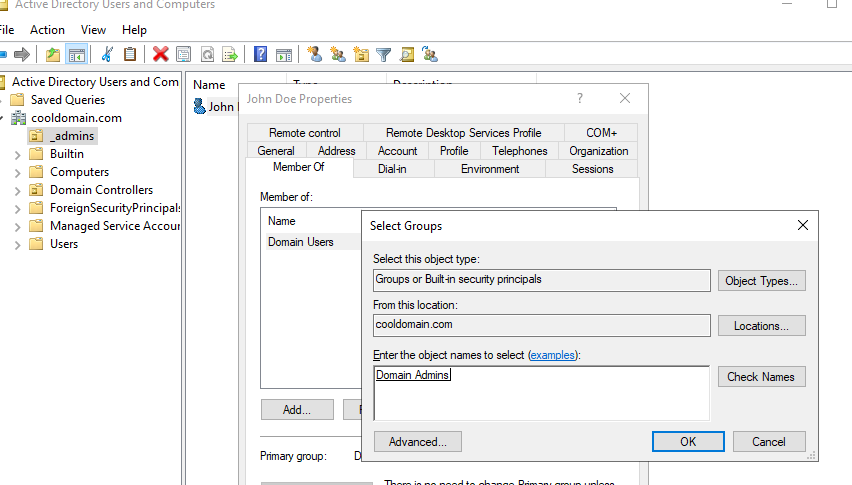
Click next until you reach install. Then you will sign out

After the restart you should notice the domain name in the login screen. 

Once in the machine navigate to the active directory users and computers. 

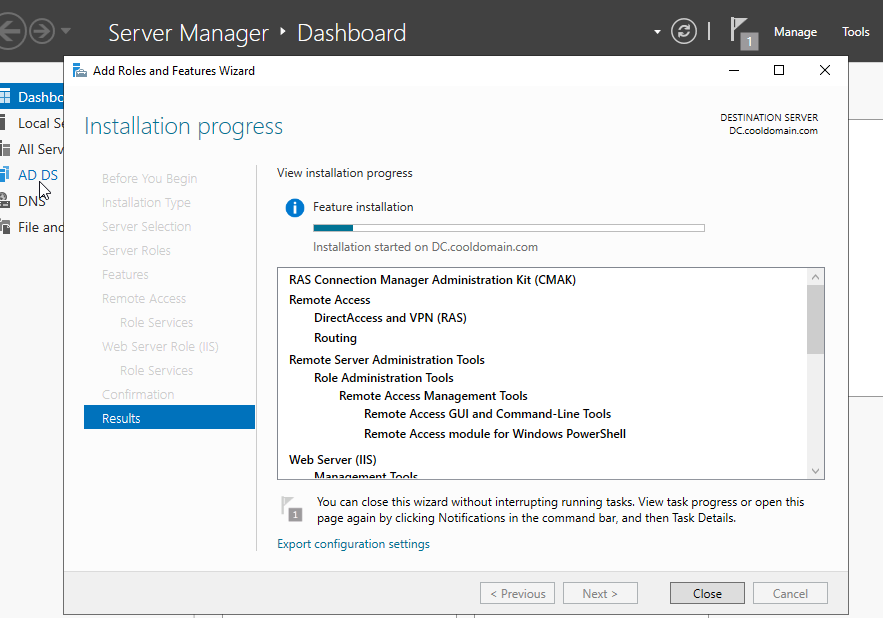
Right click the domain and add a new organization unit. We can put admin roles in this account. 

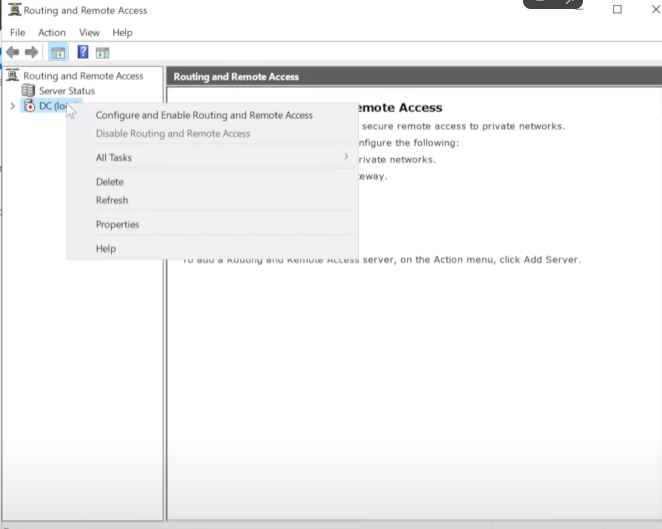
Add the name “\_ADMINS” and then right click the folder and select New User I used John Doe.

After this is created you’ll be prompted to enter a password and then the account is created. Next right click the account and select properties. Select “member of” and then in the object domain enter, domain admins and click check names. This will assign the role of administrator to the account. 

Now the account is an admin!

You can now sign in with the credentials you created. And we will install RAS and NAT

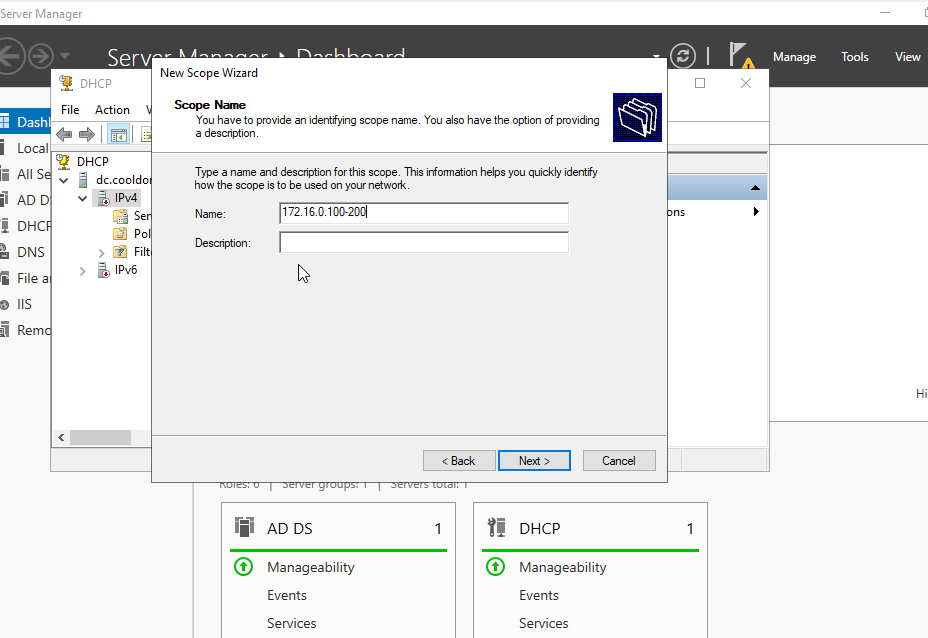
Once again in the server menu select add roles and features, click next, next, next, and in server roles select “remote access” and click into roles services and add “routing”

Once that’s installed click on tools in the server manager dashboard and select Routing and Remote access. From here click on DC local and select configure and enable routing and remote access. 

Select the option NAT and select the interface the the (DHCP) option and finish the process. If the DC local icon switched to green, this step is complete.

Now to set up DHCP with IP scope information.

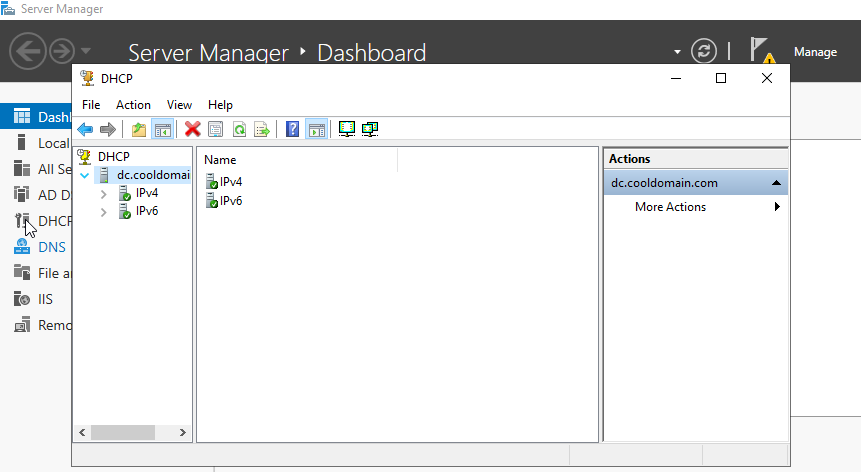
On the dashboard click “add roles and features” and next until you see DHCP server and “add features. Then next until you install. After go to tools and select DHCP and add the scope. On the ipv4 tab select add scope.

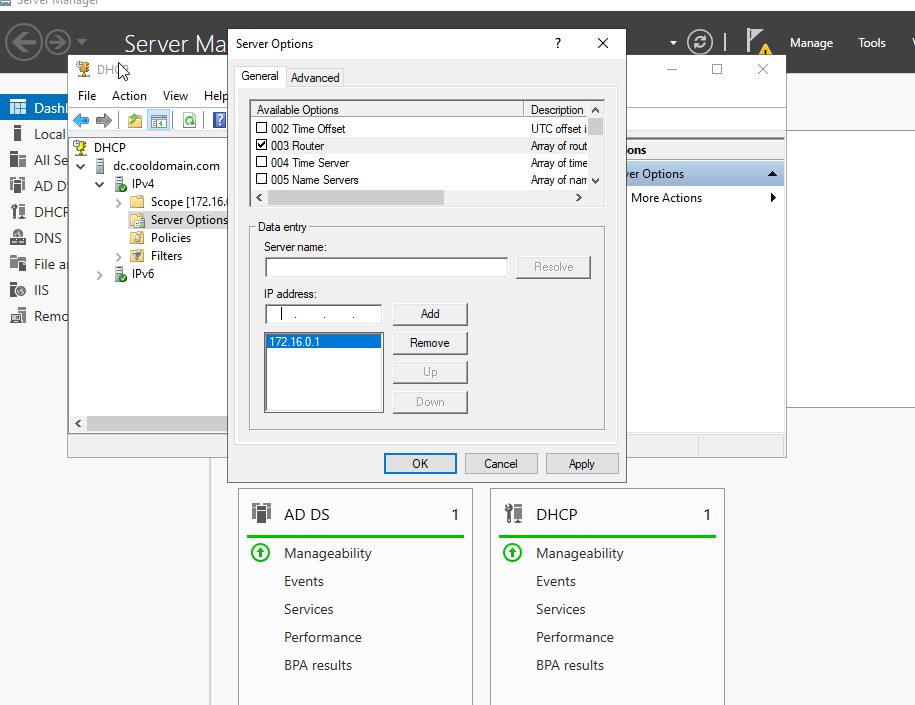


The subnet mask is 255.255.255.0

Click next until you can configure your dhcp setting.

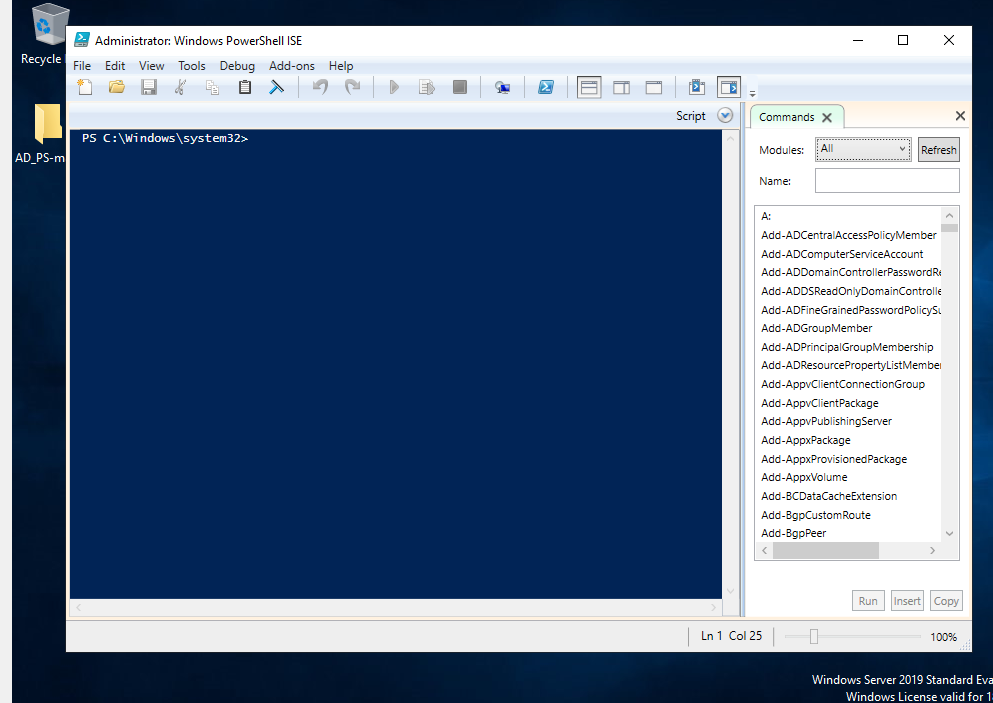
For the router, were using the DC as the router so the ip should be 172.16.0.1 Click next until you can st the DNS and doman ip as the same. Click next until finish

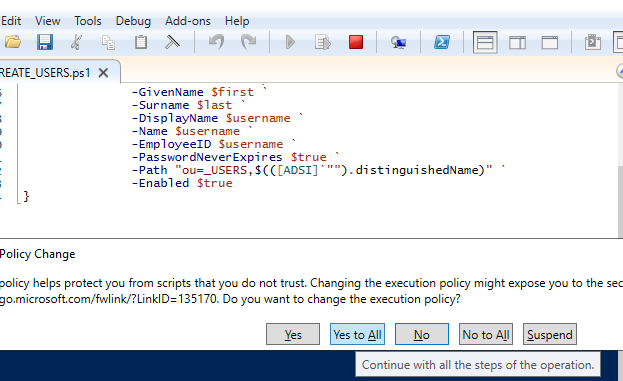
Once you refresh the ipv4 authorize the domain. Right click to find the option and then both ipv4 and 6 should be green.

To avoid a common error well make the server a router. This can be done by clicking server options under the ipv4 options. Right click and select configure options. Here we can select router and input the controllers IP address. 

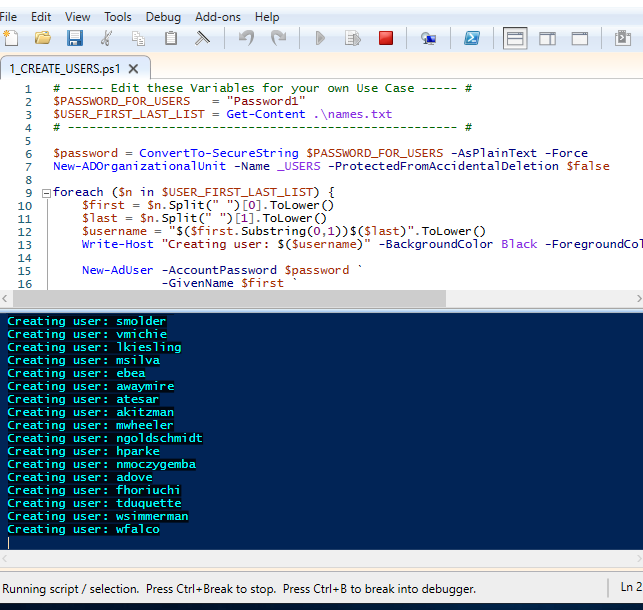
Once this is added right click on the domain, select all tasks and restart. Before we can run the powershell script we have to disable some security features. On the server manager dashboard select configure this local server and find “disable IE enhanced security configuration options.” Select off for both.

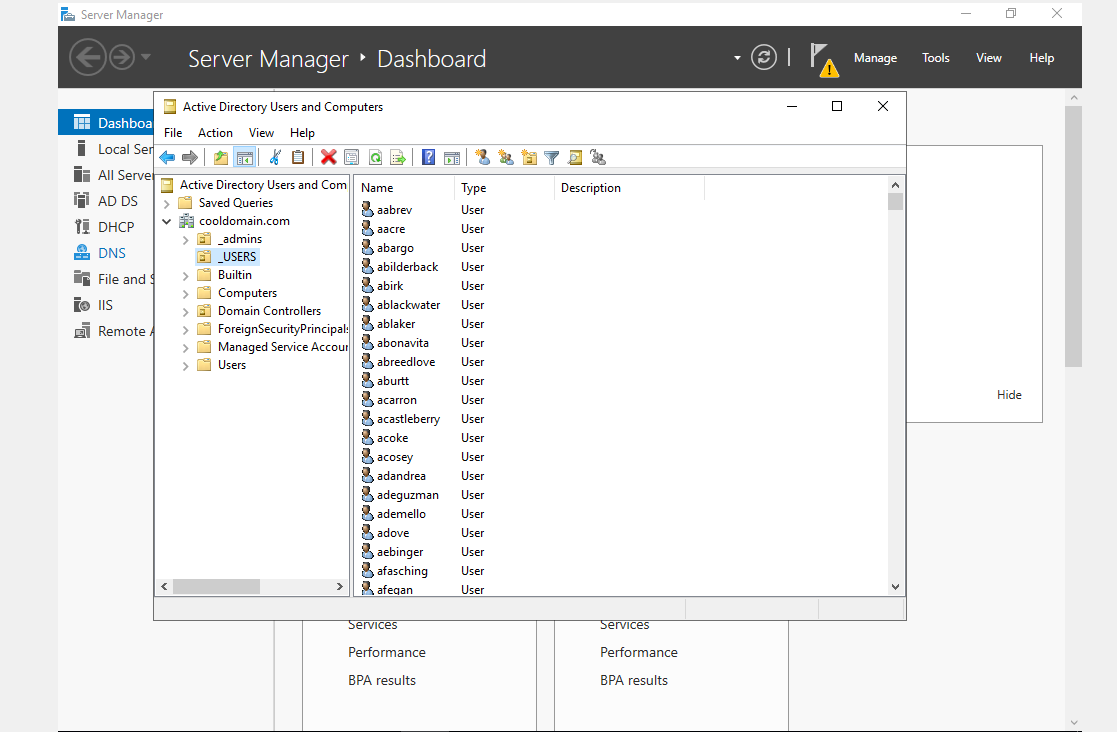
Now we can run our script. I will be using Josh Madakor’s powershell script. [joshmadakor1/AD\_PS (github.com)](https://github.com/joshmadakor1/AD_PS) Edit the text and add your name to the top of the “names” list. I added “John Doe”

Navigate to “Powershell ISE” 

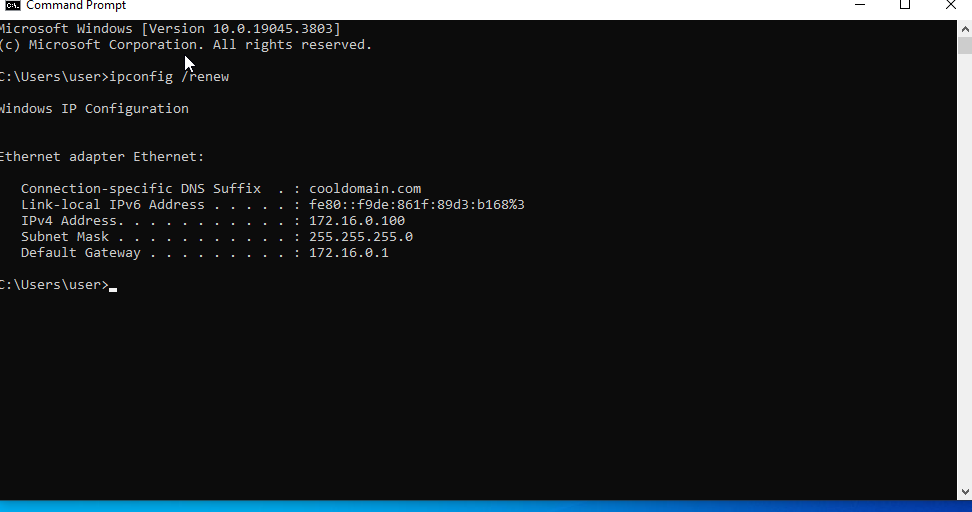
Click file and open here and load the “1\_CREATE\_USERS” script. Before running the script we have to give the system permission to execute the script. Enter Set-ExecutionPolicy unrestricted. When Prompted select “yes to all” 

Now CD into the folder downloaded by using “C:\Users\YOURUSERNAME\Desktop\AD\_PS-master” And then press the play button to run the file. The following should appear if the script is ran successfully.

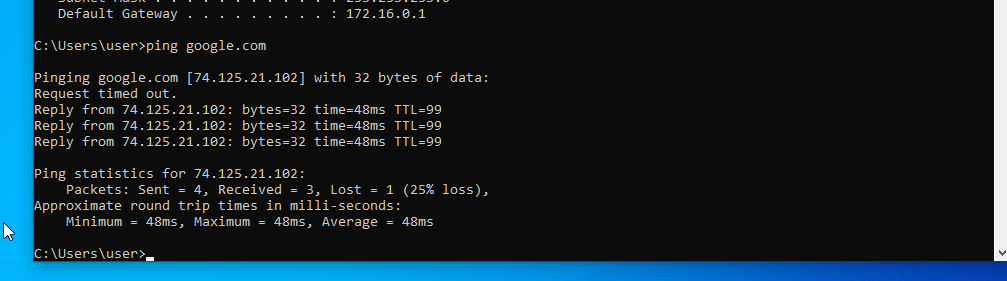


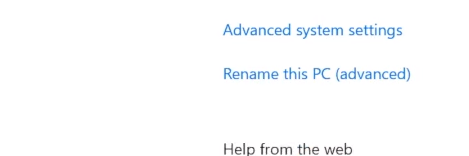
After we can navigate to “active directory users and computers” to confirm the accounts have been created. 

Now we can navigate to the other windows 10 machine and confirm the connections.

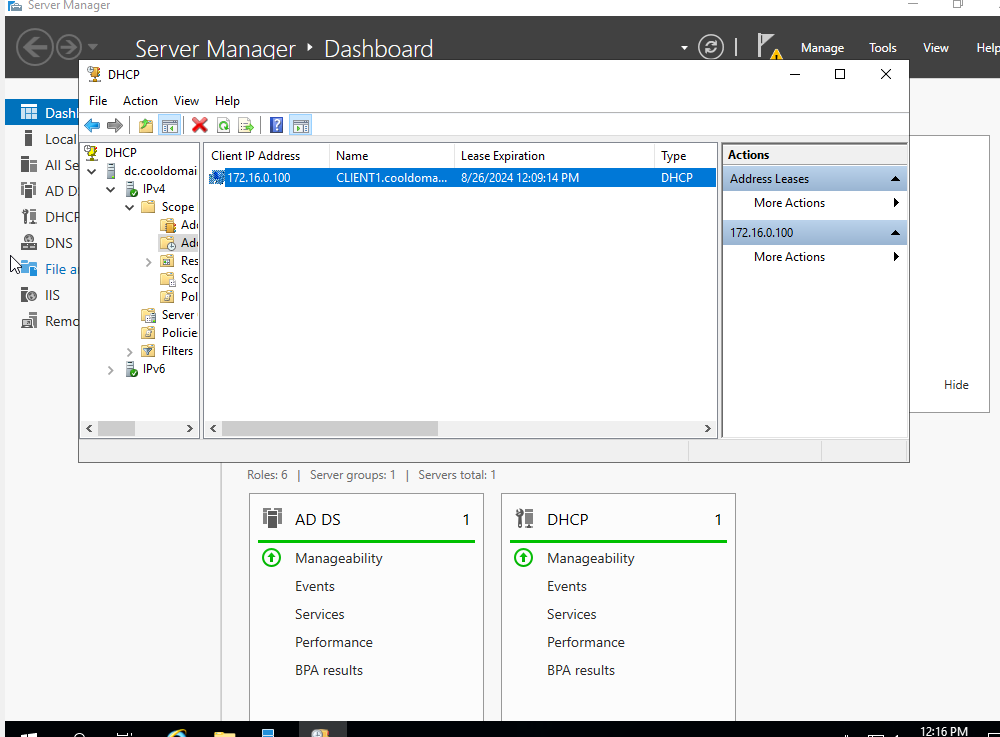
Once in the windows 10 machine go to the command line and enter “ipconfig /renew” and press enter. 

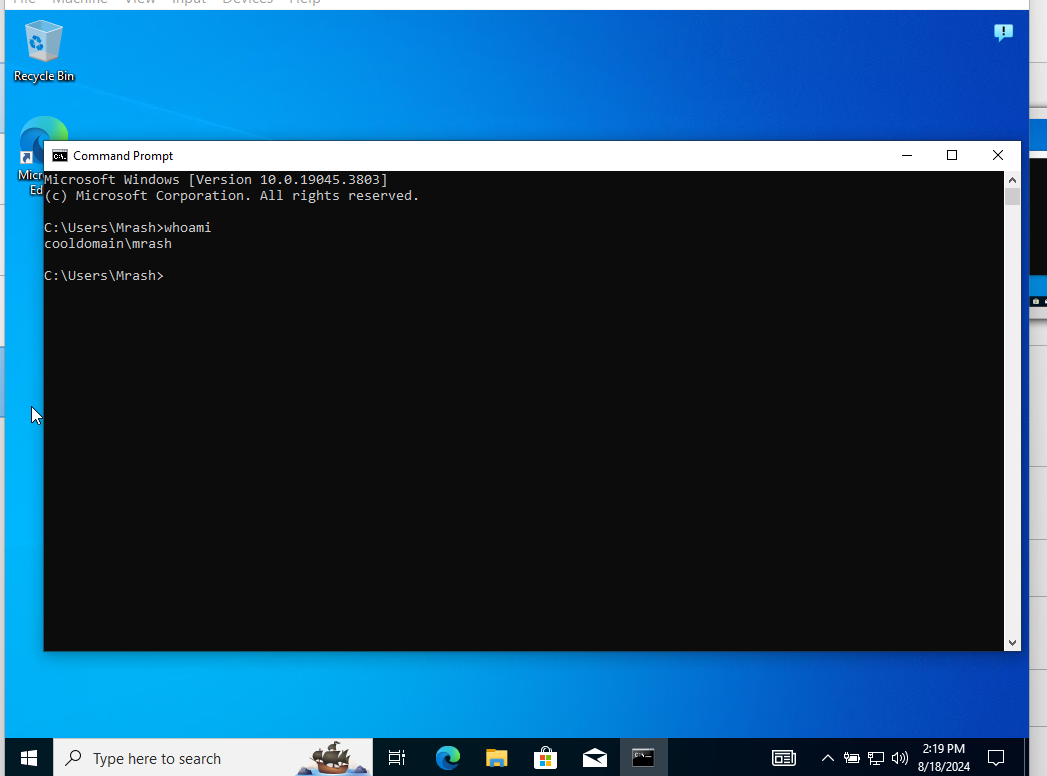
Here we can see our windows 10 machine connected to the default gateway and the internet!

To check connectivity we can ping google

Right click start and select system then “rename this pc advanced.”.

Enter the name “CLIENT1” and under the domain add your domain name. Mine is “Cooldomain1.com” then when prompted enter the DOMAIN CONTORLLER admin credentials and restart.

Let confirm the client was successfully added navigating to our domain controllers DCHP setting (found in tools) and then under ipv4, under address leases we should see CLIENT1 listed. 

Heading back to our windows 10 machine we can now log in with any user credentials we’ve created. For example, lets log in with “Mrash” Enter the created credentials and open command prompt and enter whoami. Here we can see our infrastructure is a success.