Installing a Windows 2019 Server, Windows 10 Client, and Active Directory on VMs In Virtual Box:

By Brennen Tse

Purpose:

Configure a Windows 2019 Server as a domain controller to create an Active Directory domain which would manage users and computers including a Windows 10 Client VM.

Background:

Windows 2019 Servers are the usual hardware for domain controllers and Active Directories Domain Services. Because I do not have physical equipment, I will have to settle with virtualization with virtual machines through VirtualBox. Although VMWare is more well known, that costs money, while VirtualBox does not.

Active Directory (AD) is a Microsoft proprietary database and directory service. In the AD, there exists information about the environment like what users or computers there are, their access levels, permissions, etc. Active Directories can contain hundreds of organizational units, which are basically smaller containers which can be managed by a single policy. An example might include an econ company has a tax division, so that tax division may be an organizational unit inside the larger company domain.

Active Directory Domain Services (AD DS) and its servers are the backbone of the AD framework. These servers with AD DS on them are called domain controllers. There are multiple of these DCs for redundancy.

There are three tiers in Active Directory, domains, trees and forests:

-Domains are management boundaries. Objects in a single domain can be stored in one database and managed as a whole with Group Policy Objects.

-Trees are collections of multiple domains. A group of trees is a forest.

-Forests are security boundaries, with objects in different forests prevented from interacting with each other unless there’s trust between them. An example of

this could be different departments in a company. HR and finance may have trust between their forests, but finance and maintenance probably won't.

AD Database:
In the database, there is information about AD objects. Objects include users, computers, printers, folders, and applications. These objects can be organized through organizational units and users can be grouped. Every object has attributes. For example a user object can have a name, username and password, department, email address, ID, logon time, etc.

Table of Contents:

Prerequisites:

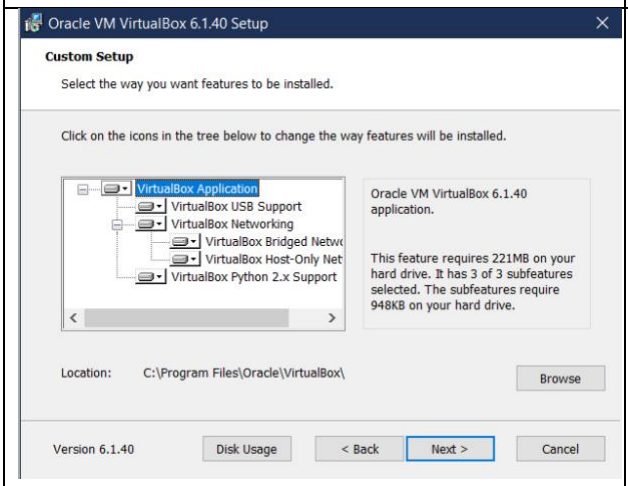
Download Windows Server 2019:

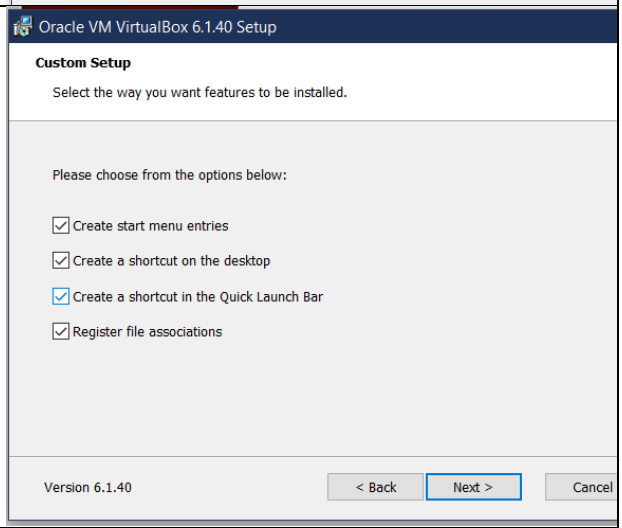
Download VirtualBox:

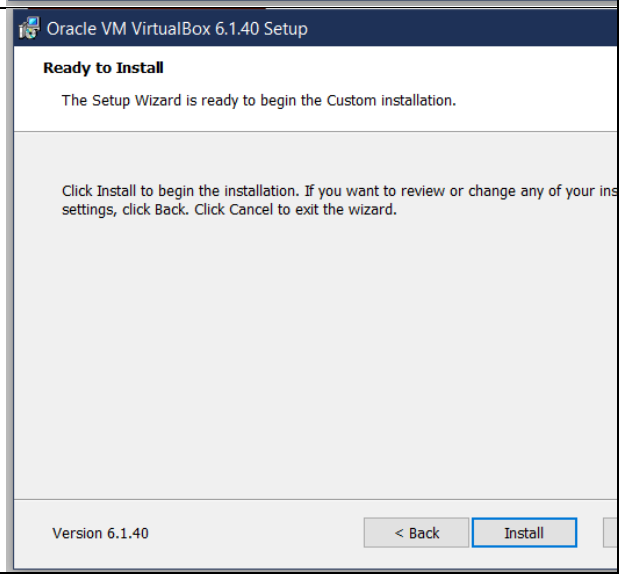
Download Windows 10 ISO Image

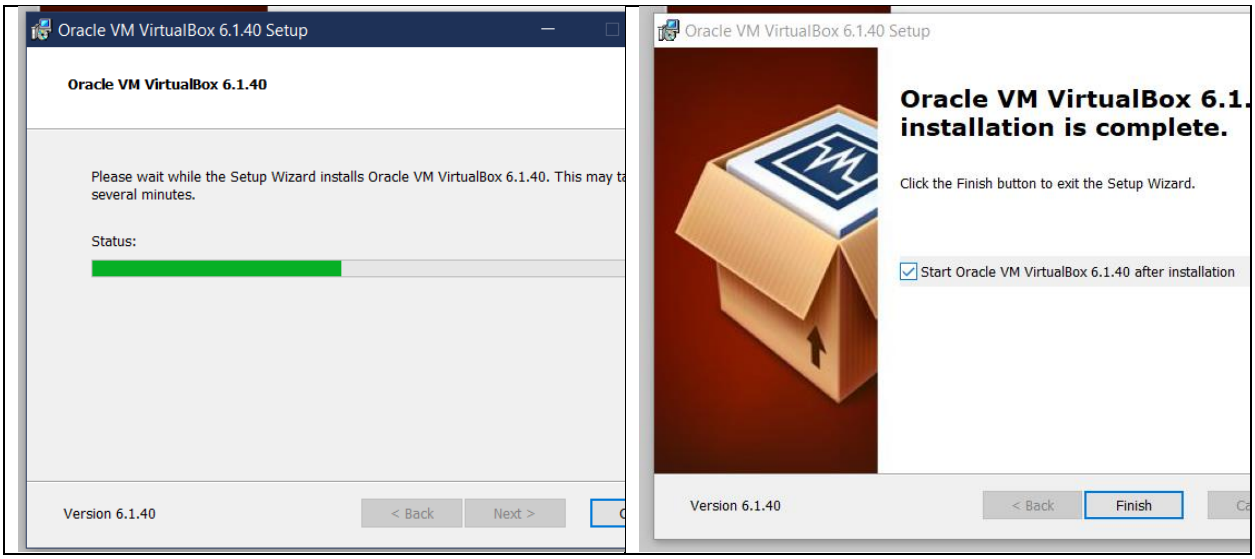
Virtual Box Installation:

Download version 6.1.40
Just keep the defaults and click next until VirtualBox is installed.



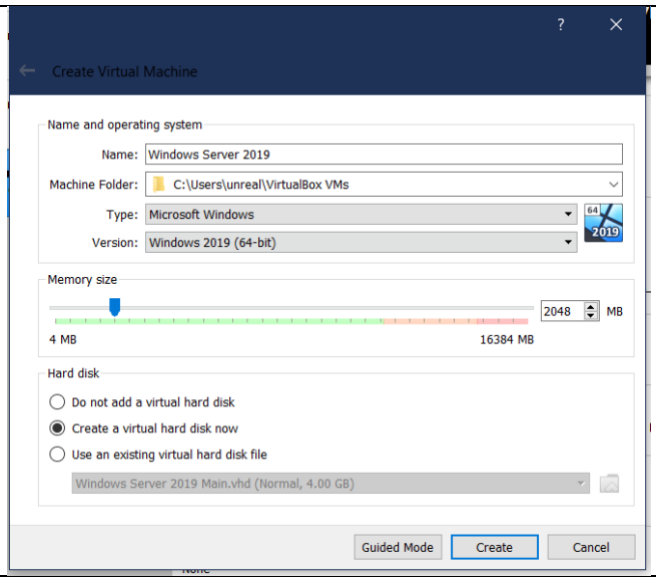


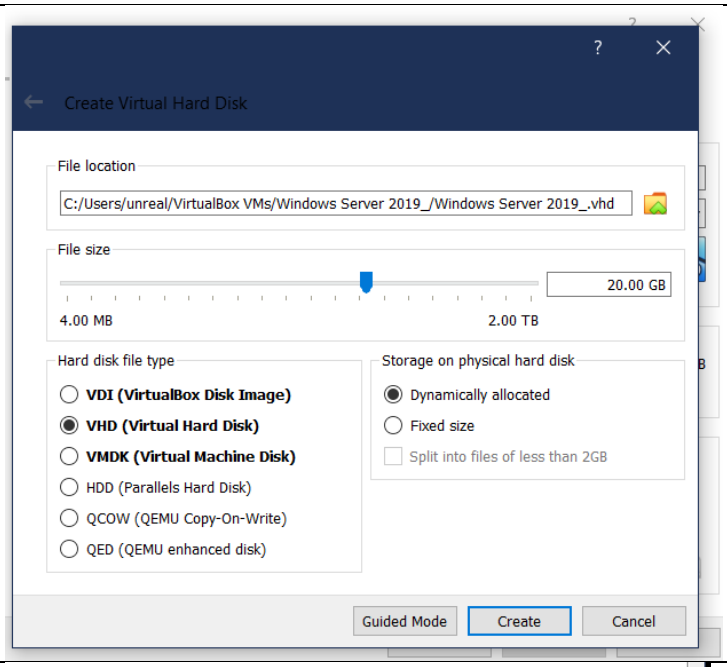
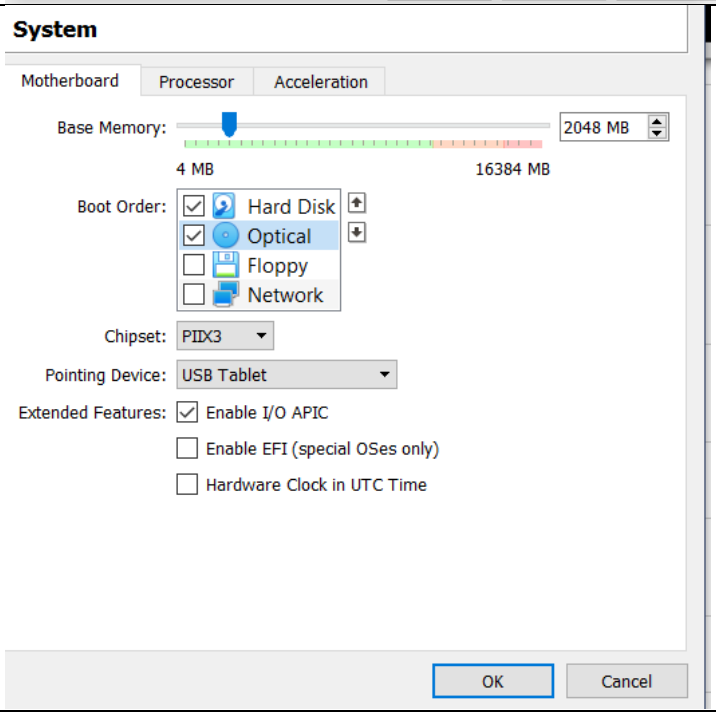


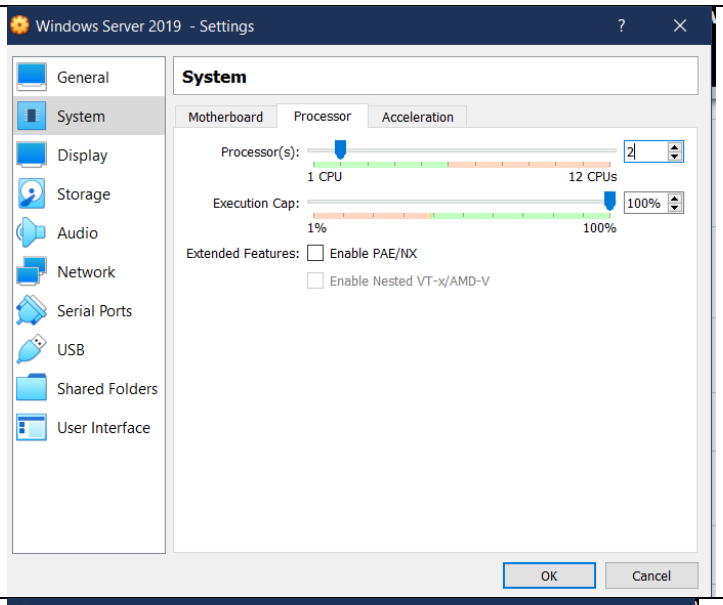
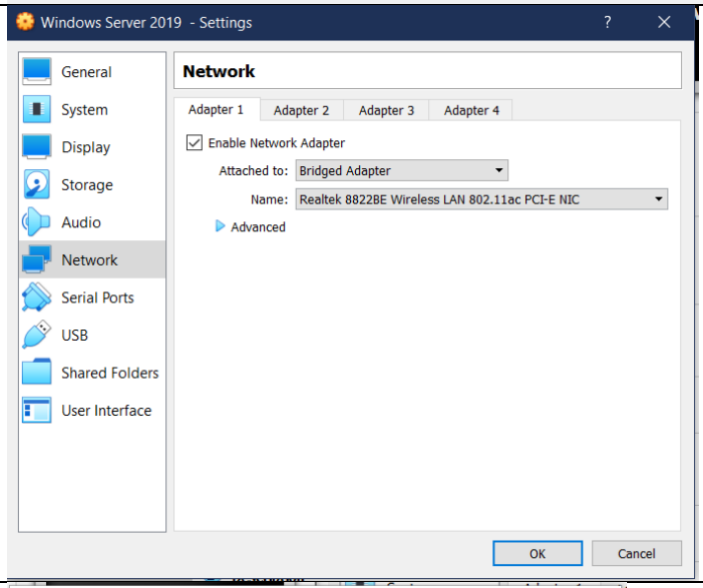
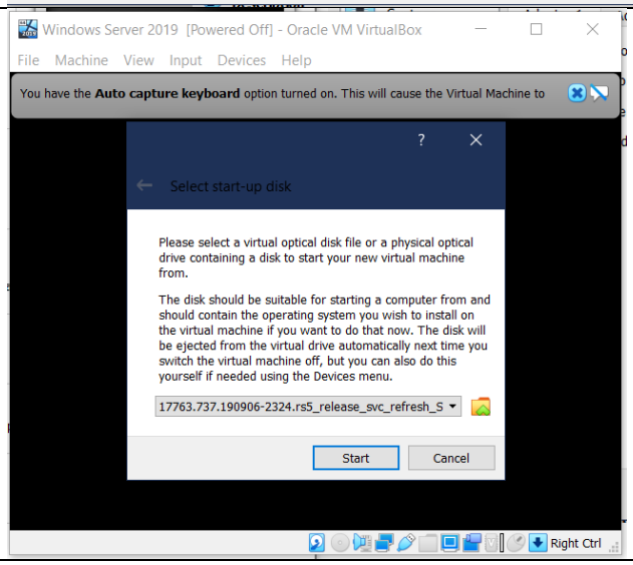


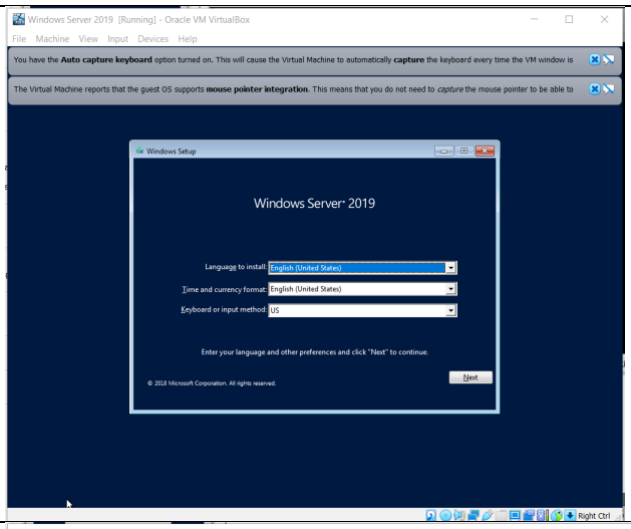
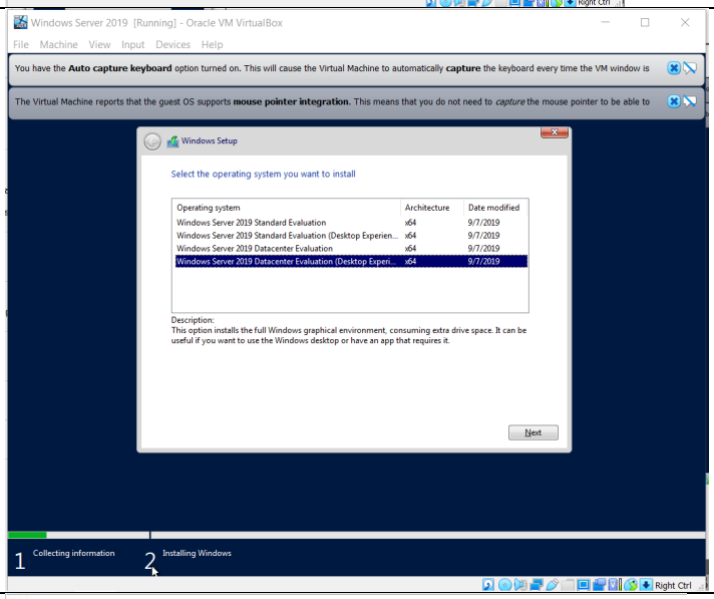
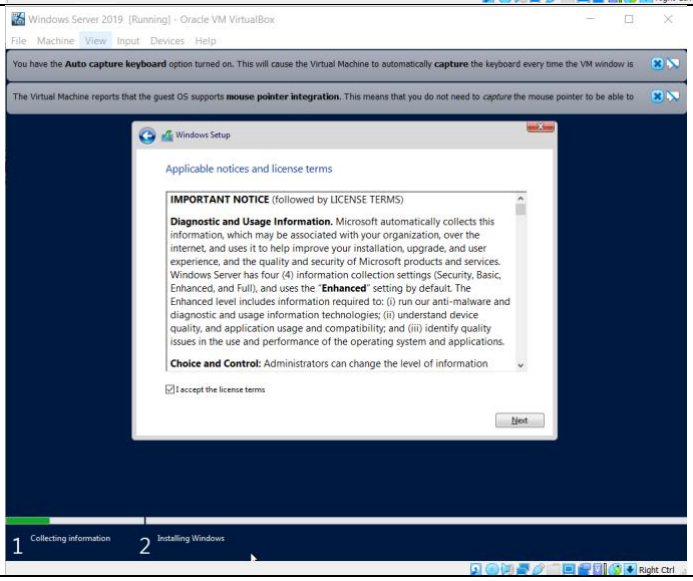
Creating Windows Server Domain Controller:

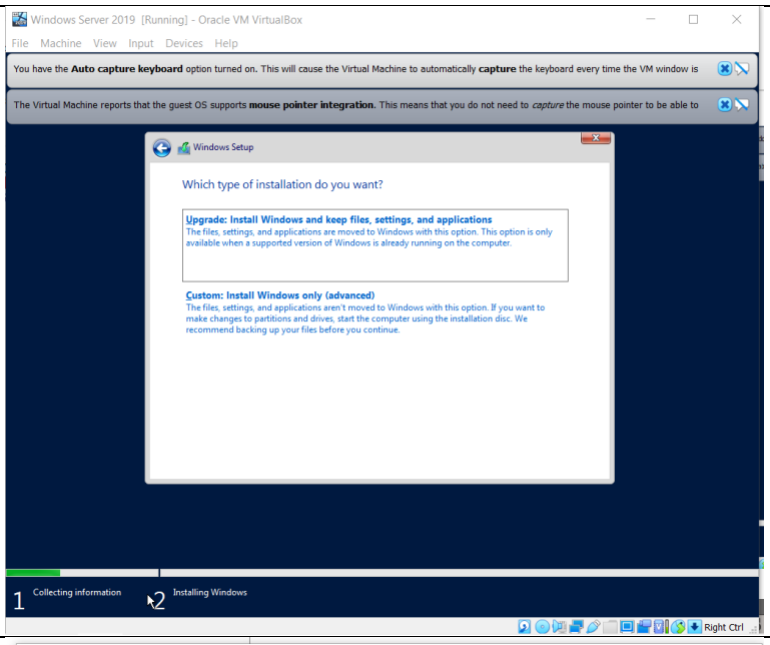
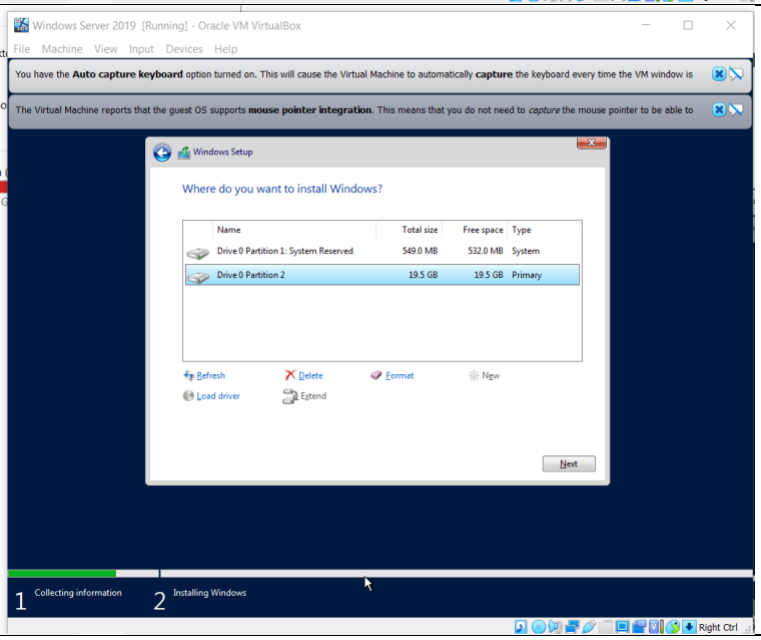
Click the blue button labeled New to create a new virtual machine. I will be creating a Windows 2019 server to use as a domain controller, so select accordingly. I allocated 2048 MB to memory and used a virtual hard disk.

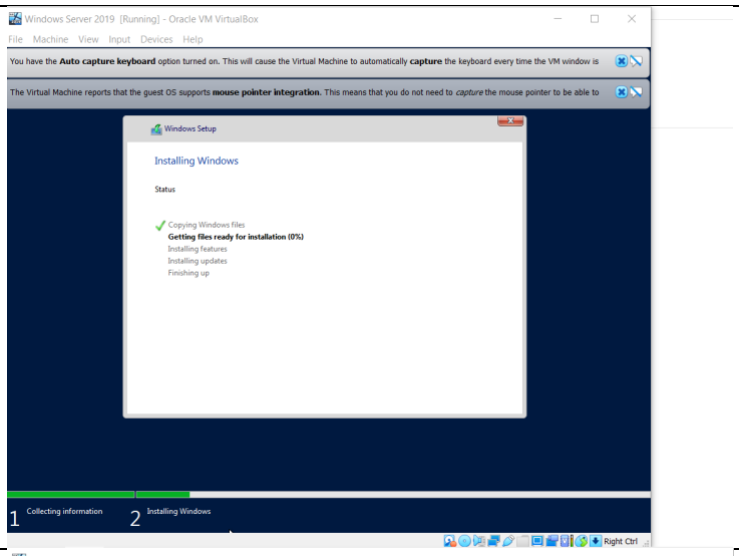
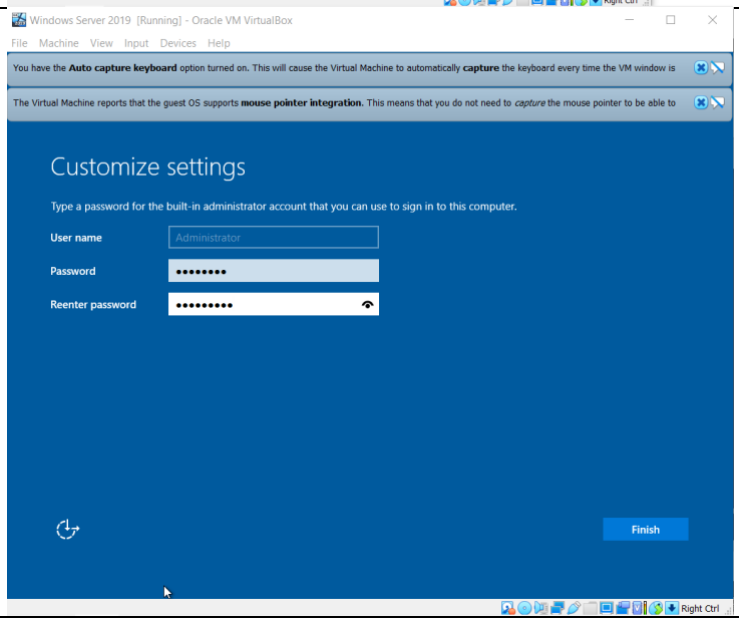


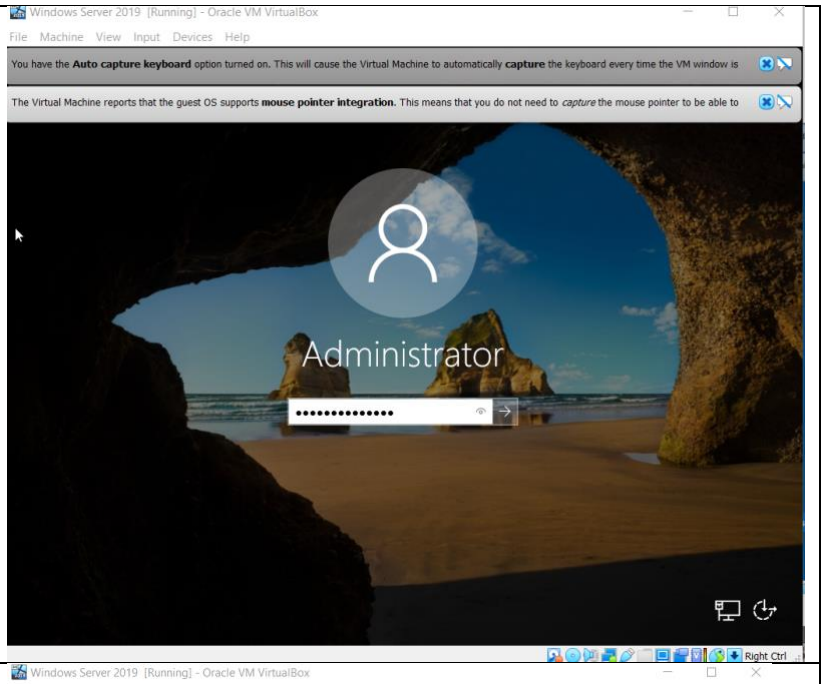
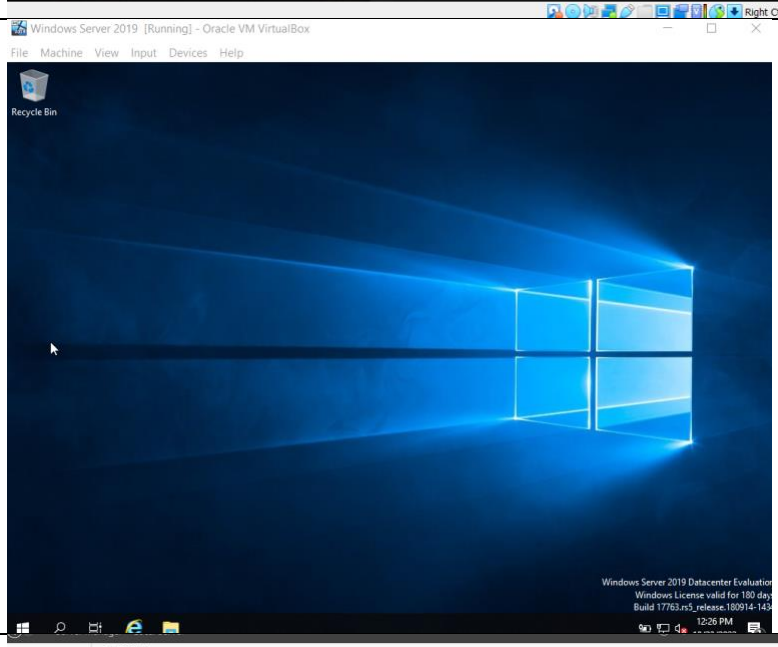
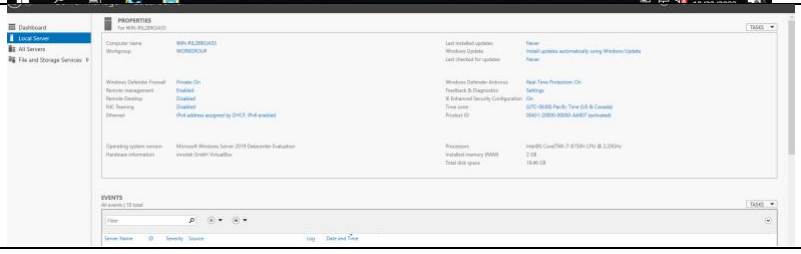
I allocated 20 GB of file size, you need at least 12 GB for a windows server to function effectively. I selected VHD although you could choose either VDI or VMDK.	
Deselect the floppy disk from the boot order and reorder so that the hard disk is on top with optical after.	

I gave my server 2 of my CPUs, if you only have 1 you can select that.	
Select the bridge adapter instead of NAT for Internet connection.	
Locate the Windows Server 2019 ISO image you downloaded earlier and select it here. Press start.	

Choose the language and select next.	
Click install now, then select Windows Server Datacenter (Desktop Experience).	
Read and accept the license terms.	

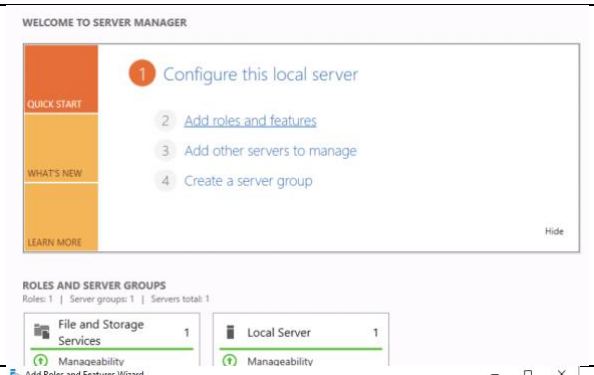
I used the custom install since I will only be using this machine to run the domain controller for active directory.	
Create a new partition by selecting new, this will serve as the primary drive.	

Click next and windows will install.	
Type in the username and password for the administrator account. Remember these for later, as they will be important for joining computers to the domain.	

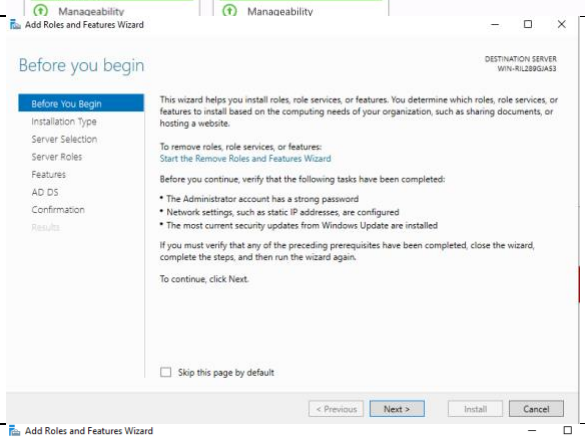
Login.	
You've successfully installed a Windows 2019 Server on your VirtualBox VM.	
You can check the specifics in the local server properties.	

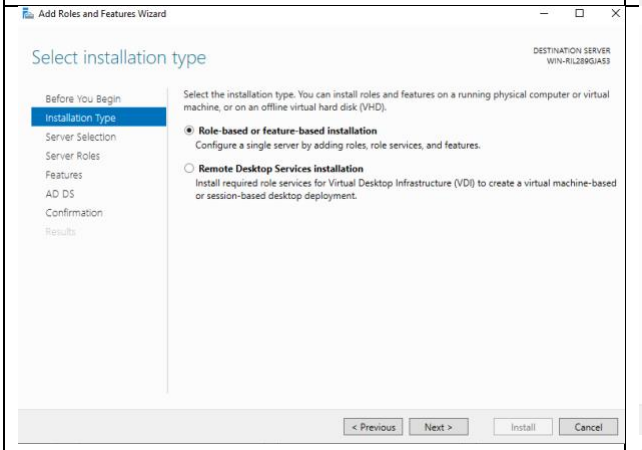
Installing Active Directory:

From the server manager dashboard, you should see an icon where you should click “Add roles and features”

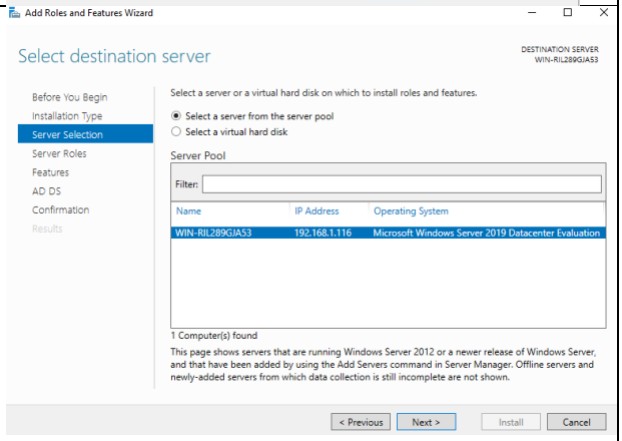


Read through and click next.



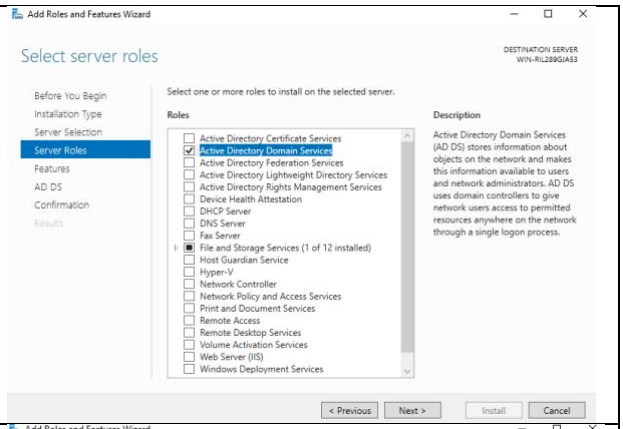


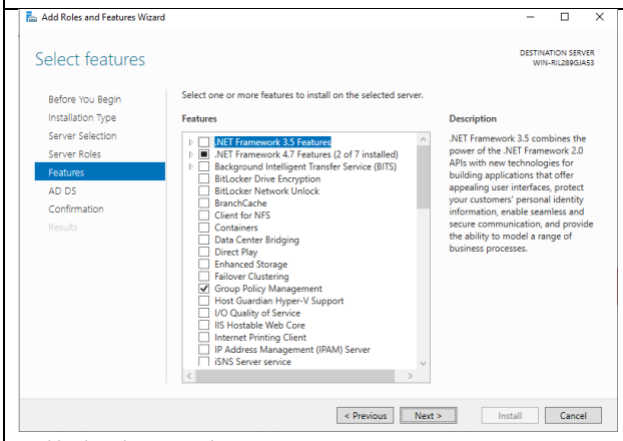
Select role based installation.



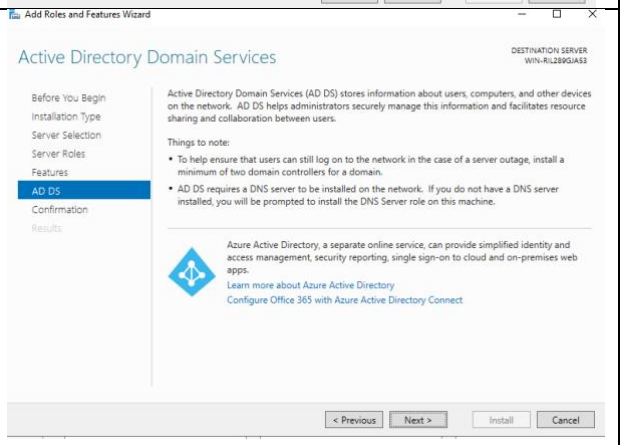
Select your windows server.

Select the Active Directory Domain Services roles as this is the main service for Active Directory and Group Object Policies.



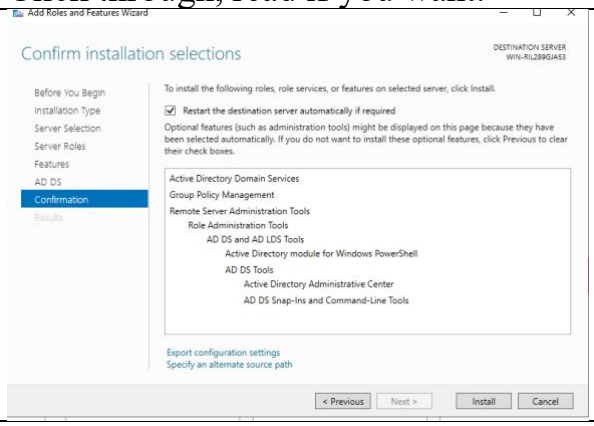


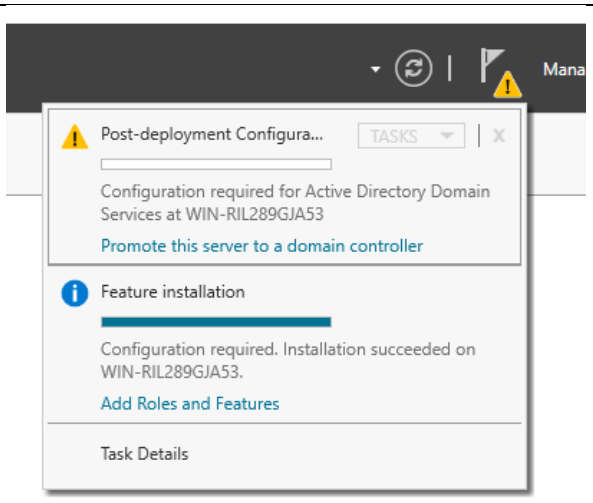
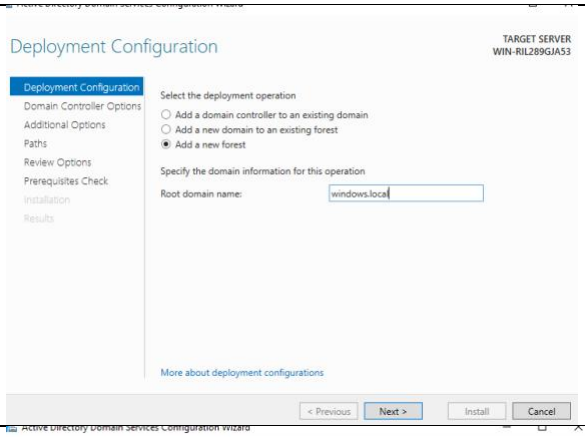
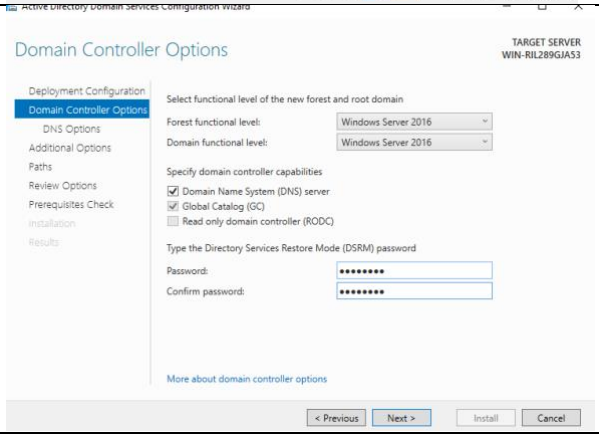
Click through.



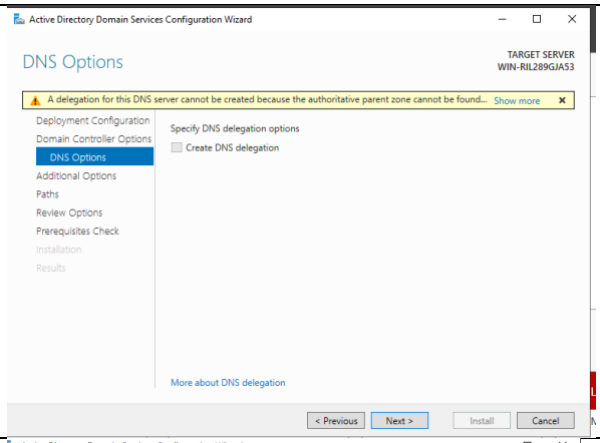
Click through, read if you want.

Select restart automatically, and confirm that all the necessary services are included. You can see GPM and ADDS there as well.

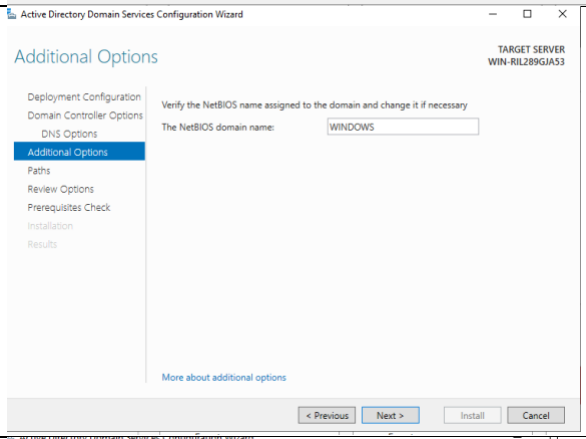


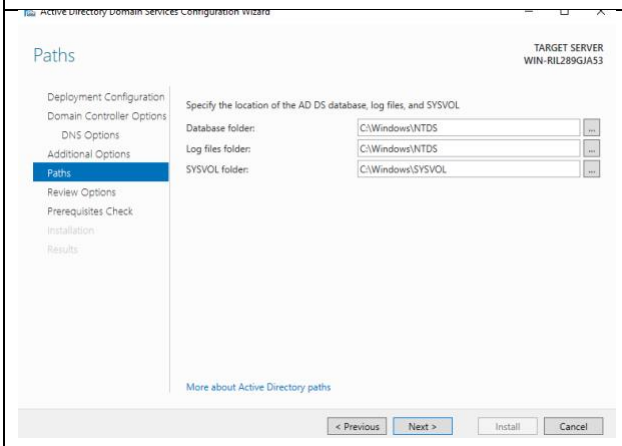
When the Domain Services finishes installing, you will see this warning pop-up in the upper right. Click on it to finish the domain controller configuration.	
Add a new forest and select a domain name, I chose windows.local because it's local.	
Leaving the defaults alone should be fine for this, but remember the DSRM password.	

Leave DNS delegation blank, we will configure DNS manually later.

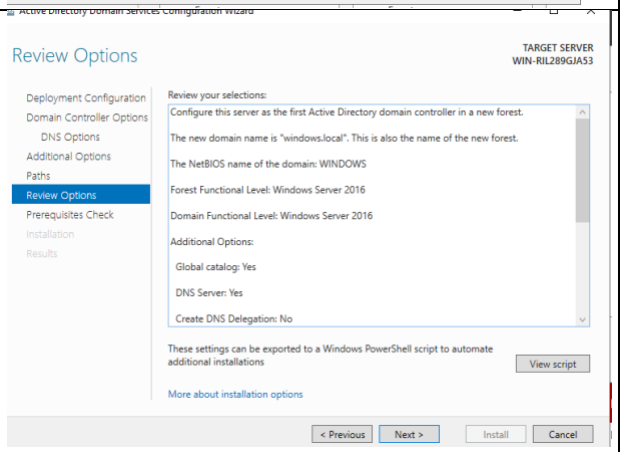


Set the NetBIOS domain name, I set mine to WINDOWS, and you can always change it later.

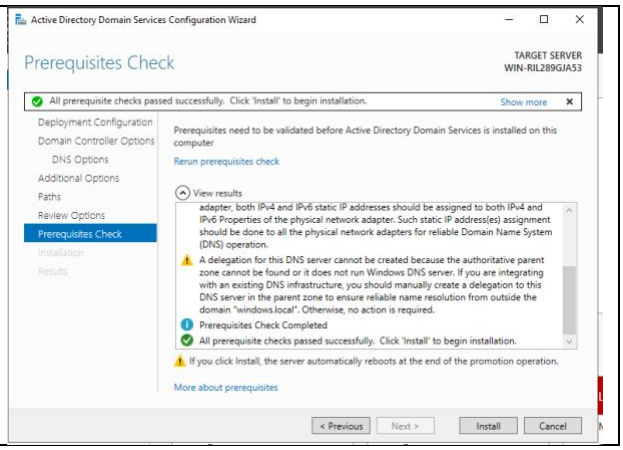




Keep the defaults.



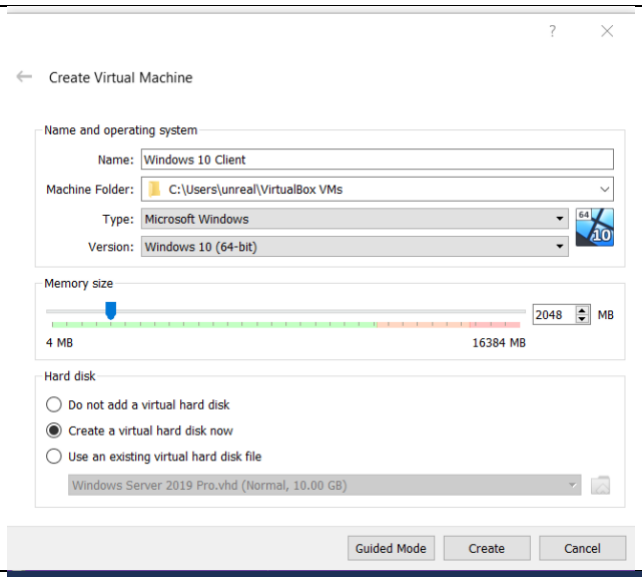
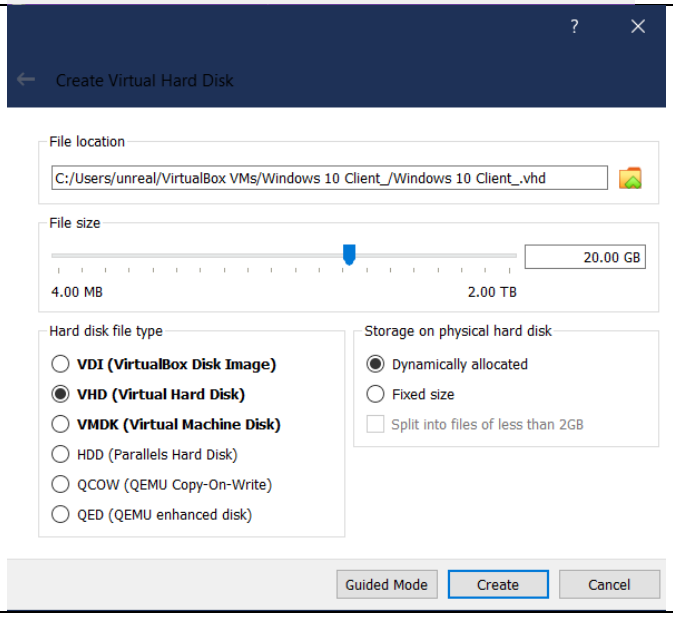
Keep the defaults.

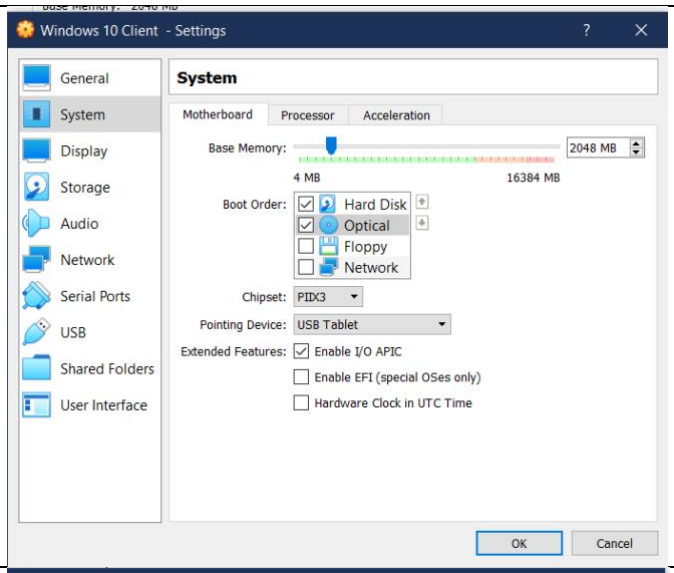
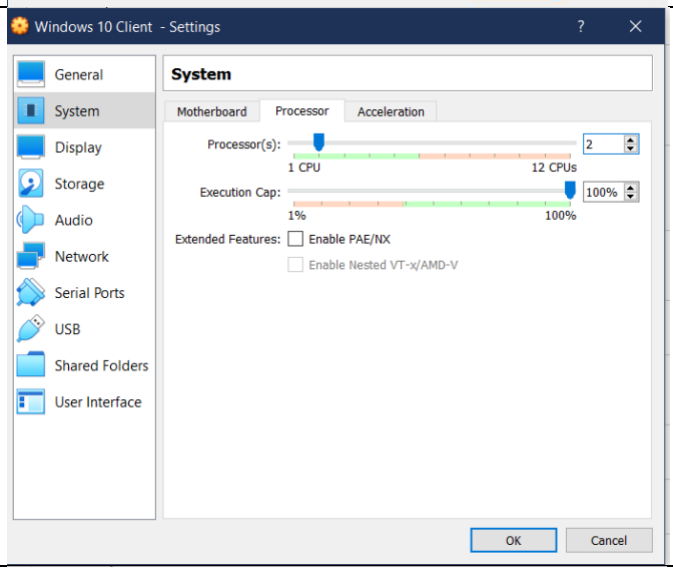
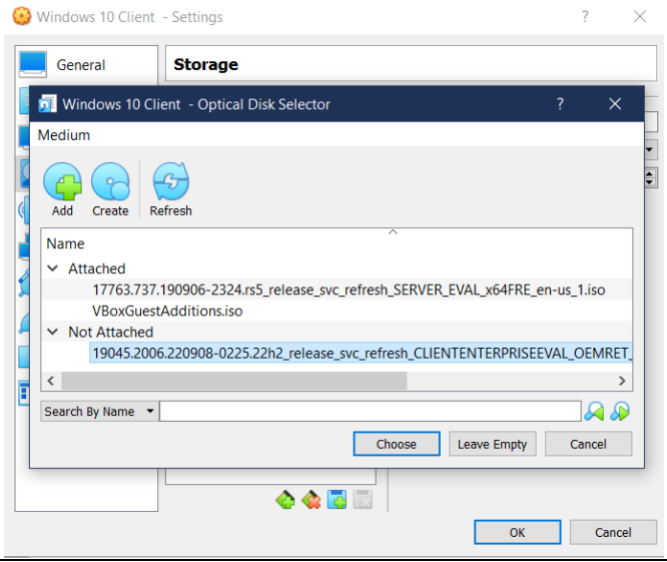
Make sure that you have all prerequisite checks passed, and click Install.	

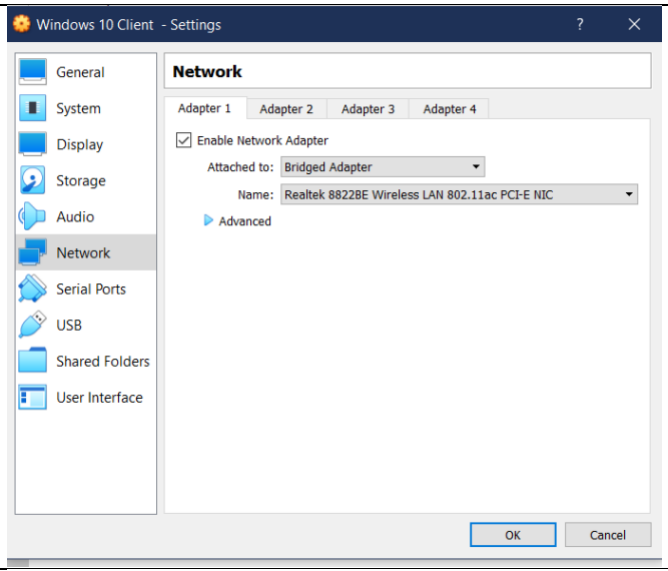
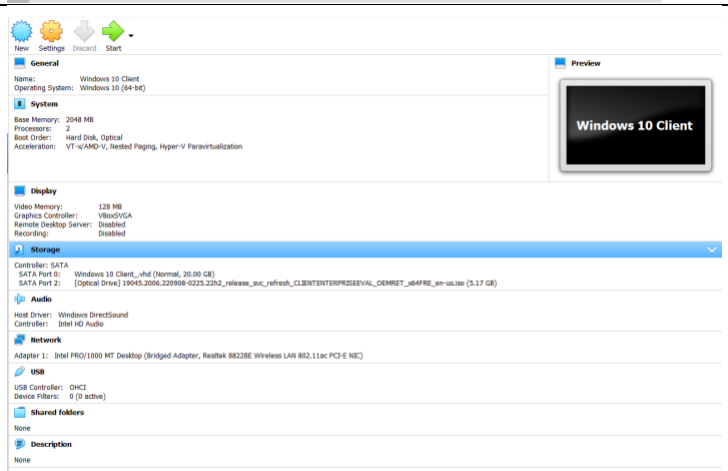
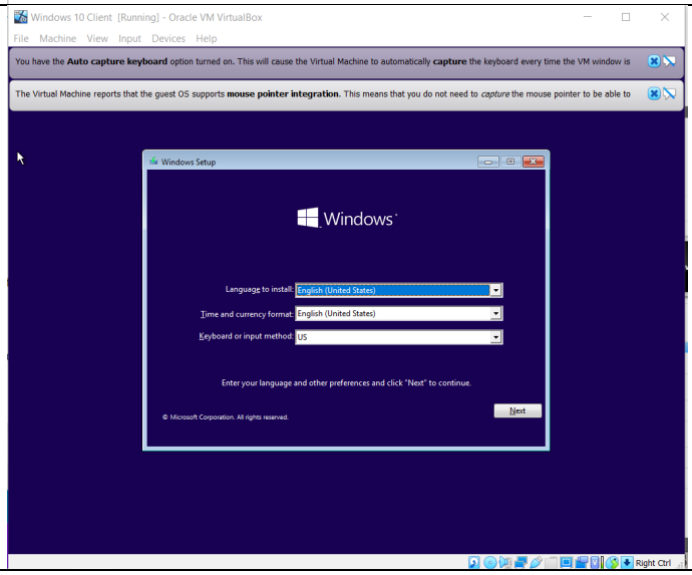
Here's the PowerShell script:

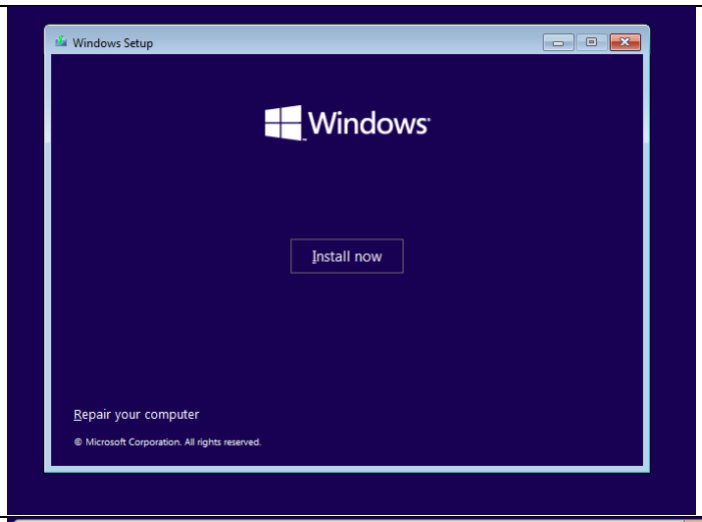
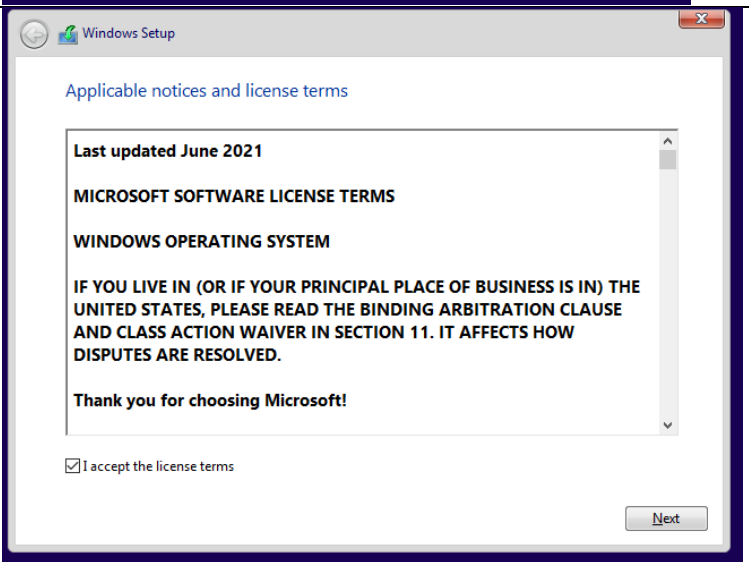
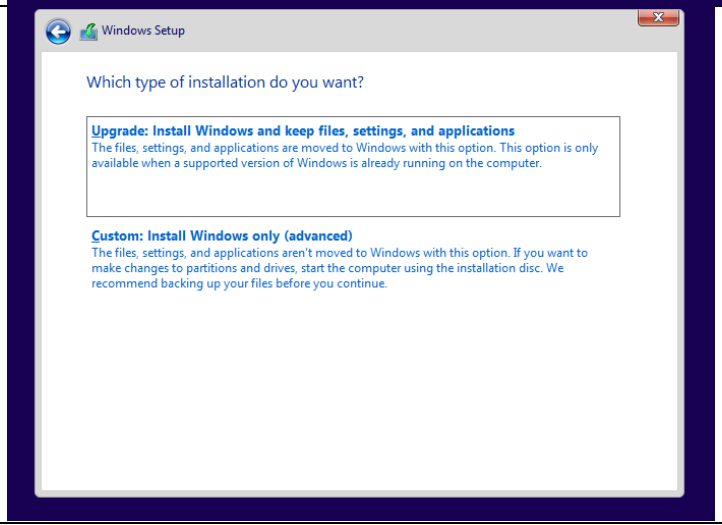
```
# Windows PowerShell script for AD DS Deployment

Import-Module ADDSDeployment

Install-ADDSForest `

-CreateDnsDelegation:$false `

-DatabasePath "C:\Windows\NTDS" `

-DomainMode "WinThreshold" `

-DomainName "windows.local" `

-DomainNetbiosName "WINDOWS" `

-ForestMode "WinThreshold" `

-InstallDns:$true `

-LogPath "C:\Windows\NTDS" `

-NoRebootOnCompletion:$false `

-SysvolPath "C:\Windows\SYSVOL" `

-Force:$true
```

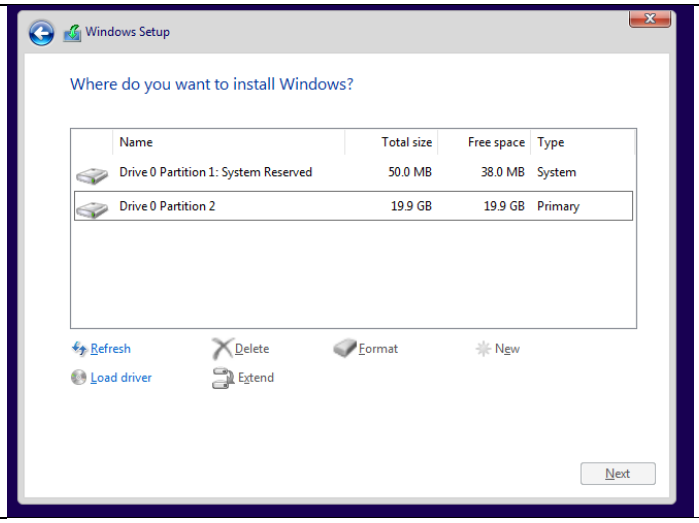
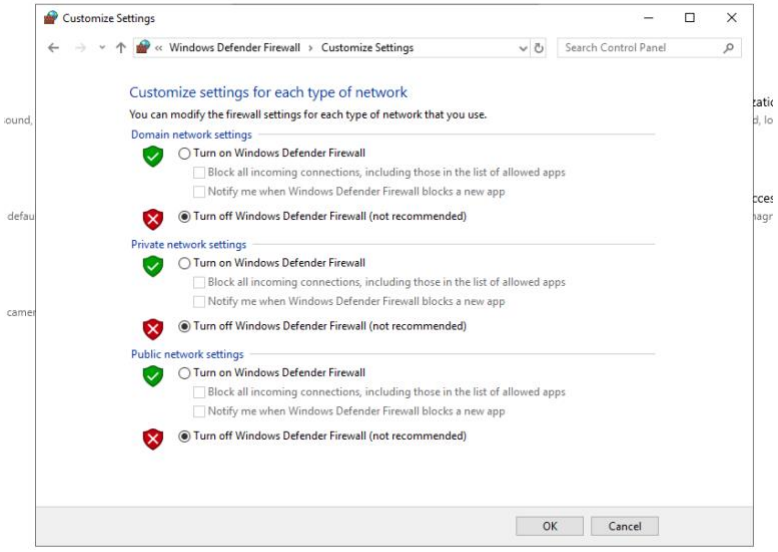
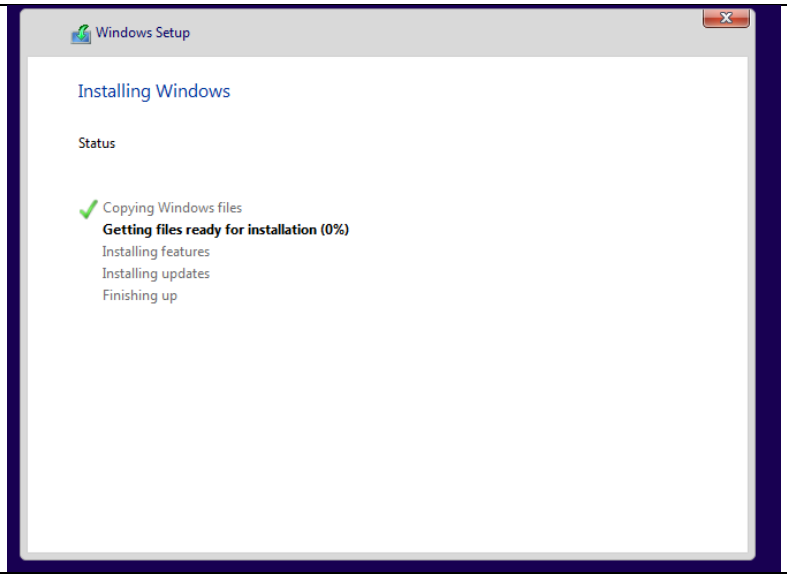
Creating Windows 10 VM:

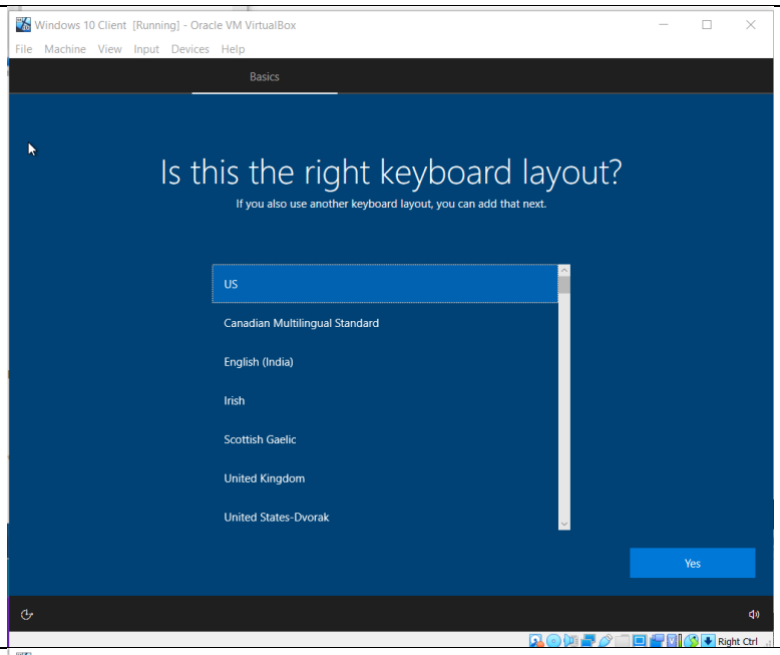
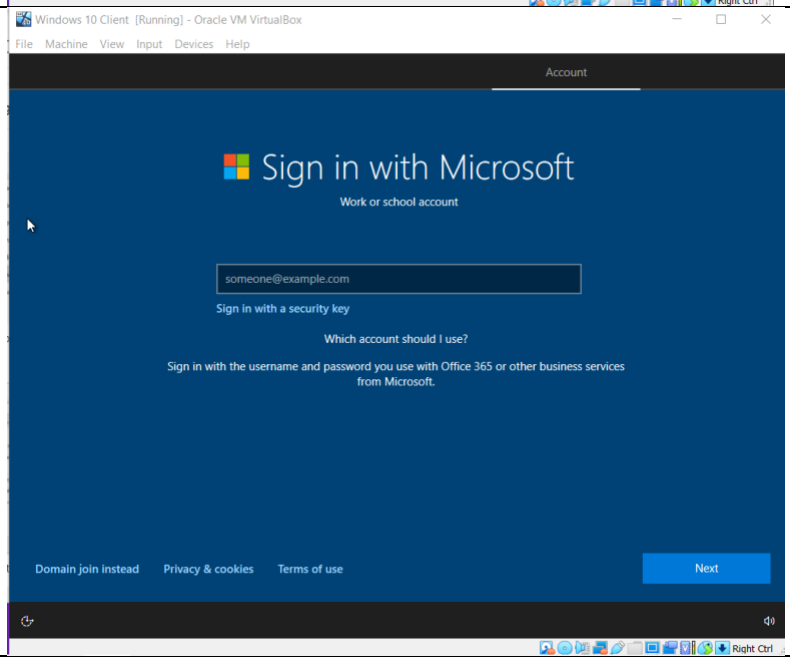
Create a new VM by clicking the blue icon. I named my VM Windows 10 Client to denote it's client status. Choose Windows 10, 2048MB of memory, and a virtual hard disk.	
I also gave 20 GBs to this VM, although it likely doesn't need that much as it's not a server domain controller. Also you can choose what hard disk you need based on your needs.	

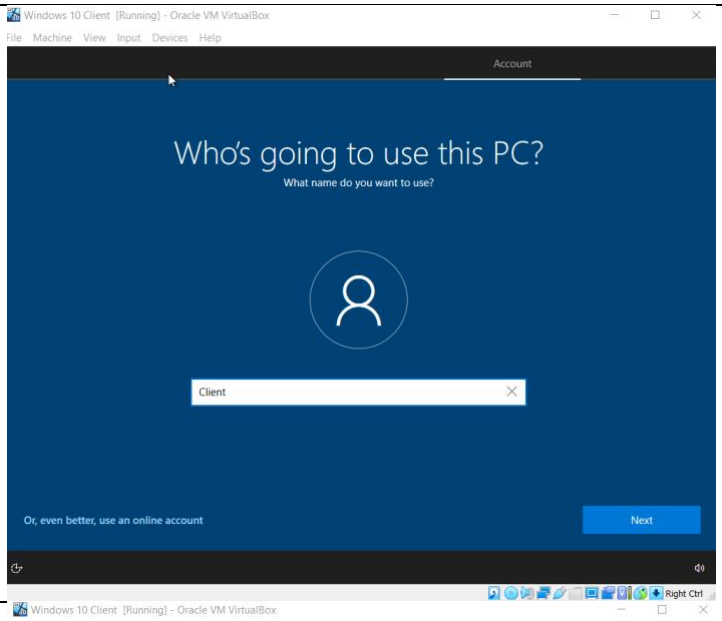
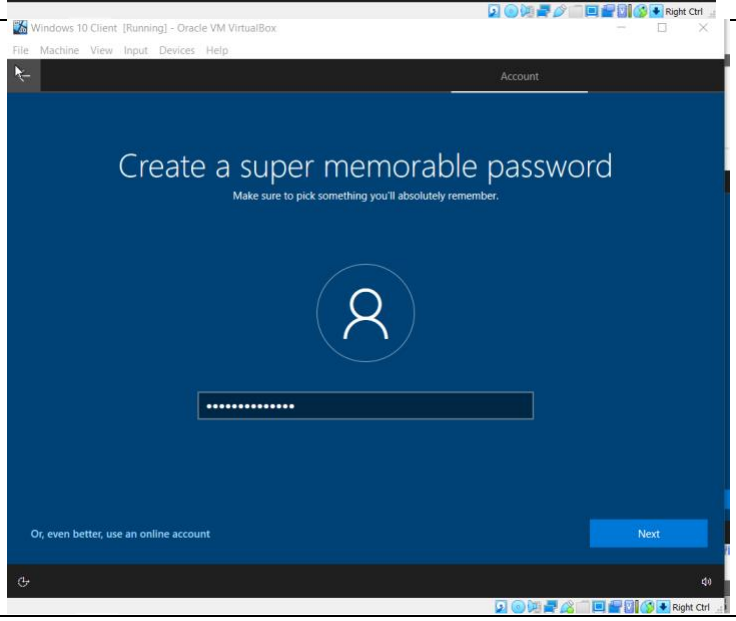
Again make sure that the floppy disk is deselected in the boot order and hard disk is on top.	
Select 2 CPUs if possible, if not, 1 is fine.	
In this instance, I went to the settings and storage for the Windows 10 VM before I started it and added the Windows 10 ISO image medium by clicking the hard drive icon. Click choose and it should be attached.	

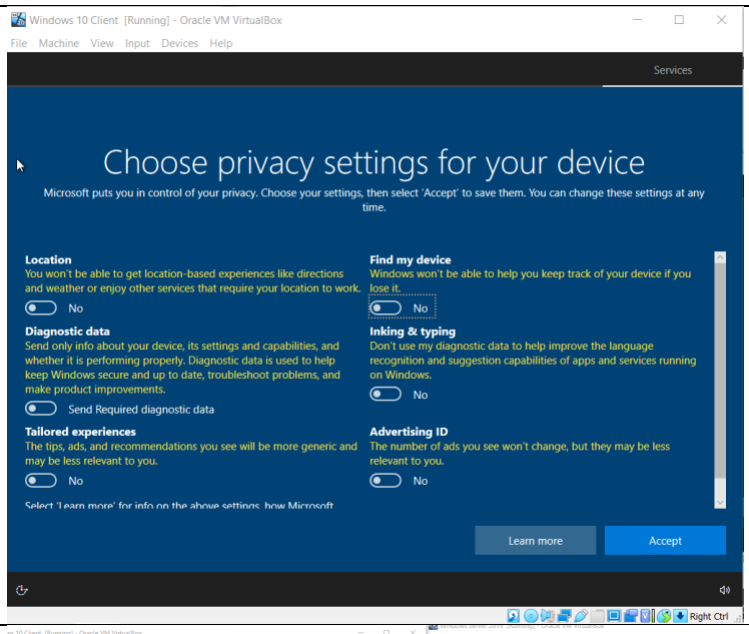
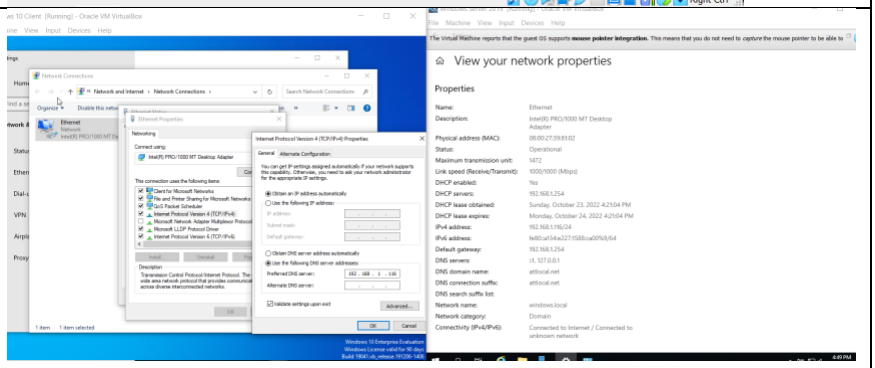
Choose the bridged adapter, and select wireless for connectivity.	
Confirm the details on the right. Then click the green start arrow.	
Select the language and next.	

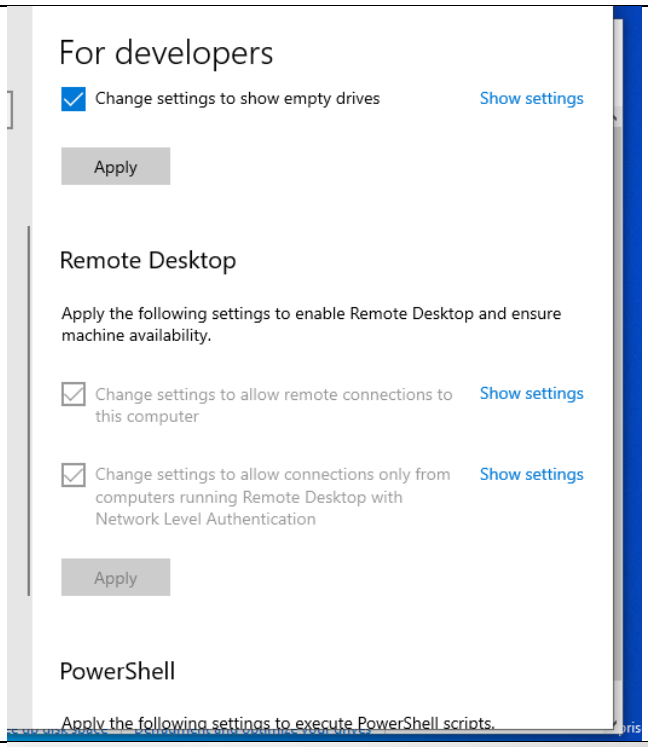
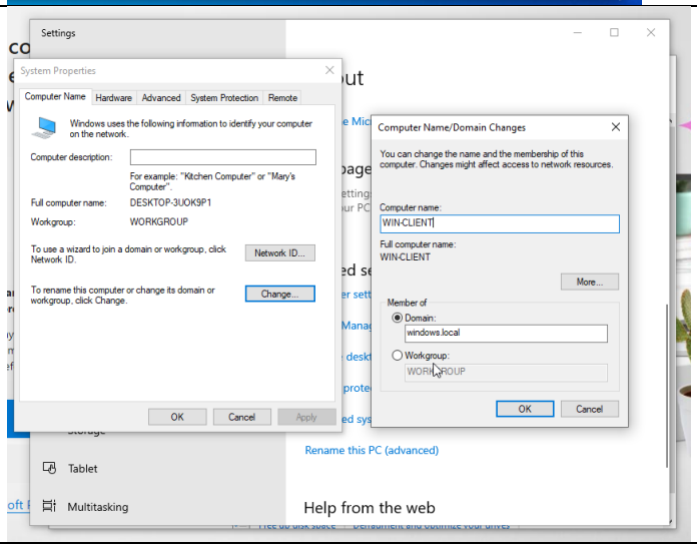
Click Install Now.	
Accept the license terms and continue.	
I'll still choose custom install because this PC will just be used to test group policy and security for the local domain.	

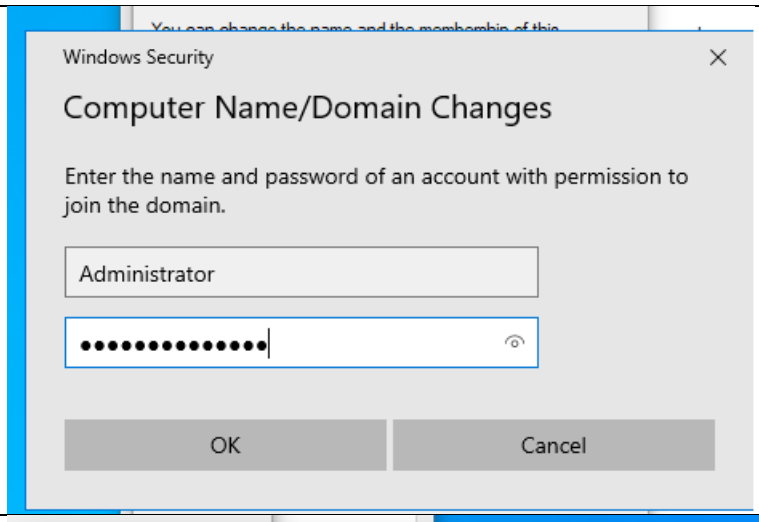
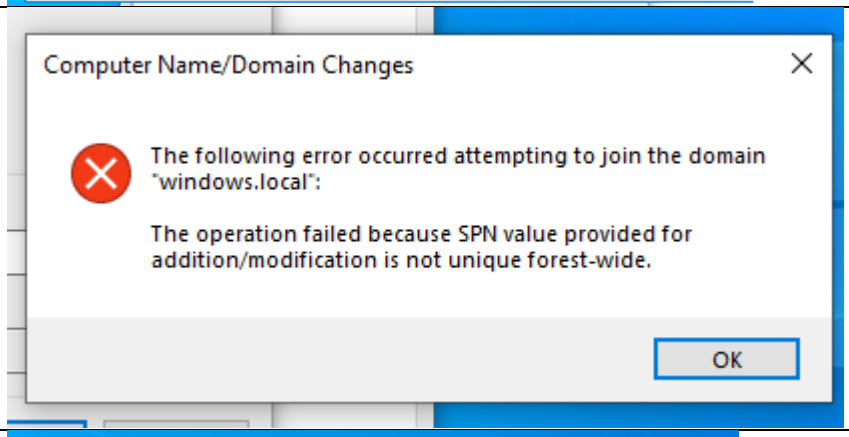
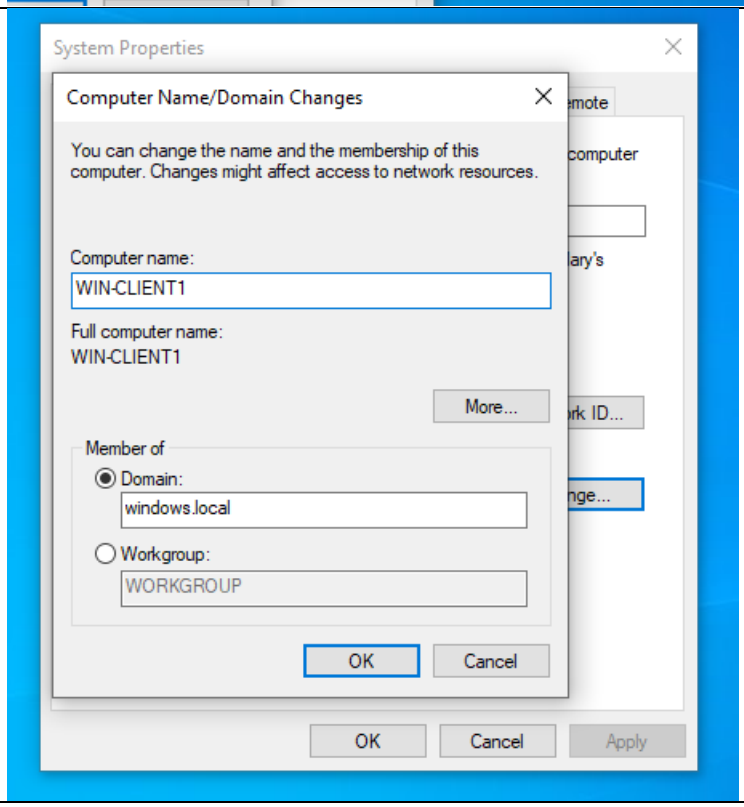
Again, create another partition for Windows by clicking new that will serve as the primary drive.	
(WARNING: NEVER DO THIS OUTSIDE PRODUCTION NETWORKS) I turned off the firewall for the Windows Server so that the new Windows PC can more easily connect to the domain. Once it's connected, I'll reenable it.	
Wait for the installation to finish.	

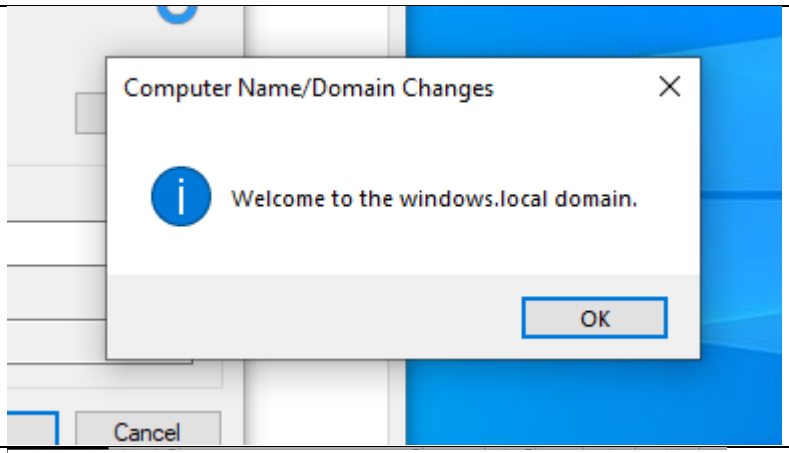
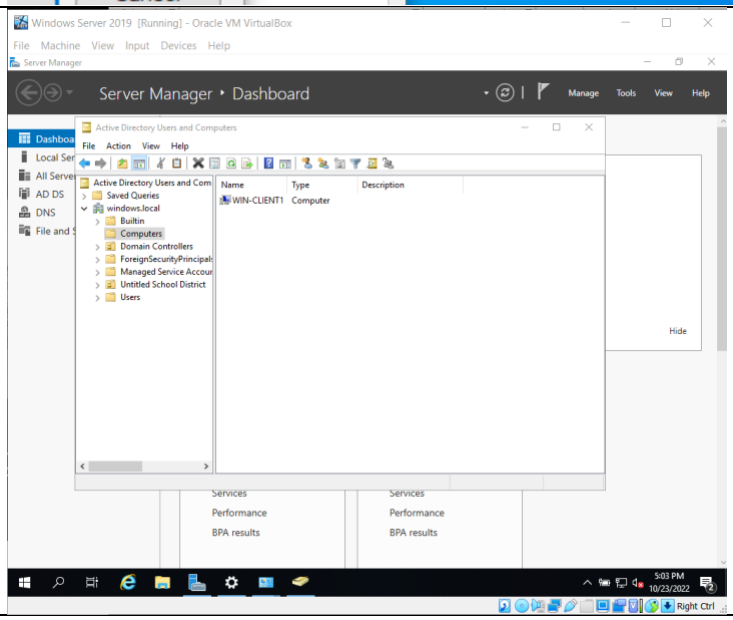
Choose the keyboard.	
I skipped signing in, instead using domain join instead.	

Name the PC.	
Give a password to the Client account.	

Deselect all these privacy settings.	
DNS: Go into your ethernet properties for your Windows 10 client. Select obtain IP address automatically, but make sure that the preferred DNS server is the SAME IP Address as the Domain Controller.	 In this case it was 192.168.1.116.

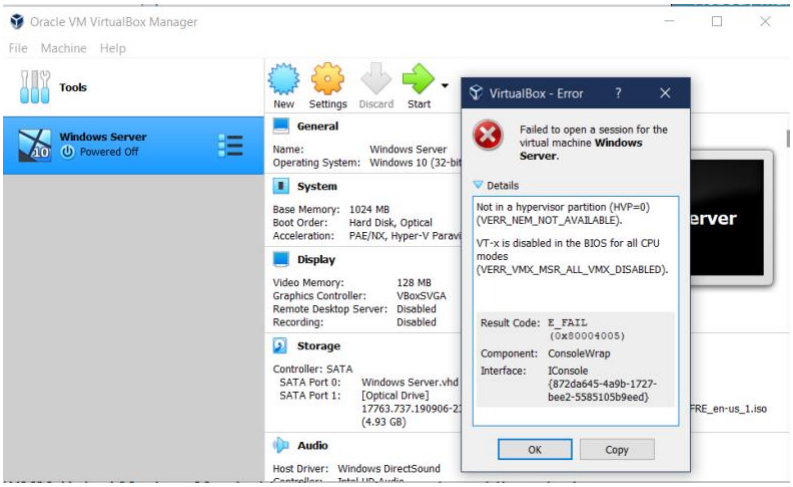
Before the computer can join the domain, remote connection must be allowed. Click apply once it's clicked.	
From the system properties in the control panel, make sure you select member of _ domain to join the domain. In my case, it was windows.local. I also renamed my computer for easier identification.	

Remember from earlier your login information for the main windows server? Enter that now.	
Possible problem: Make sure client name and domain controller name are different. Also refresh because if you don't, it won't register the new name change.	
Closer view.	

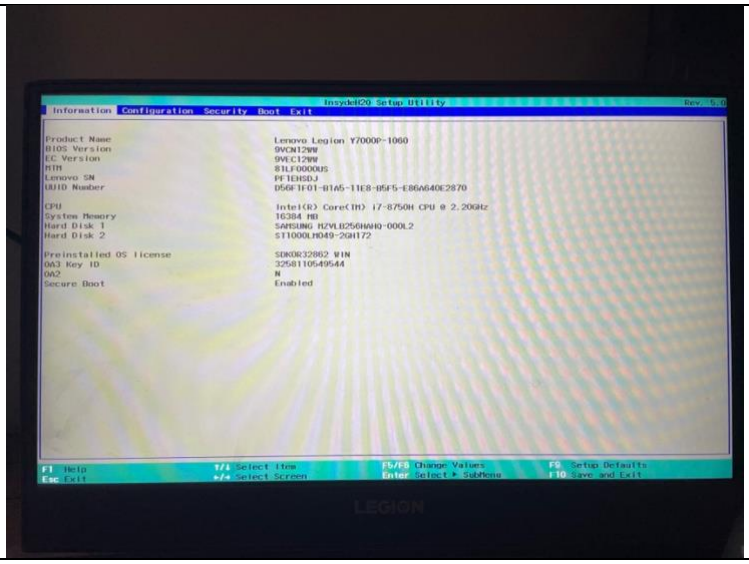
If it's successful, you should be greeted with this message.	
To confirm, open the active directory and navigate to computers, and the computer should be there. If not simply search for it by name.	

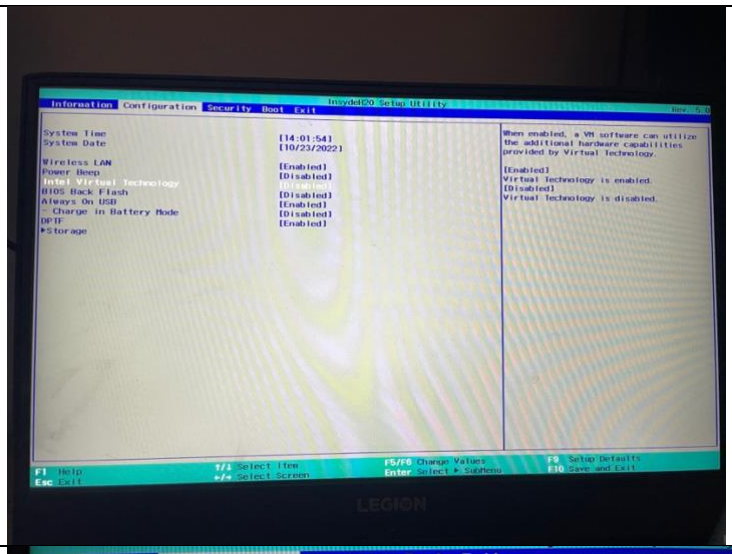
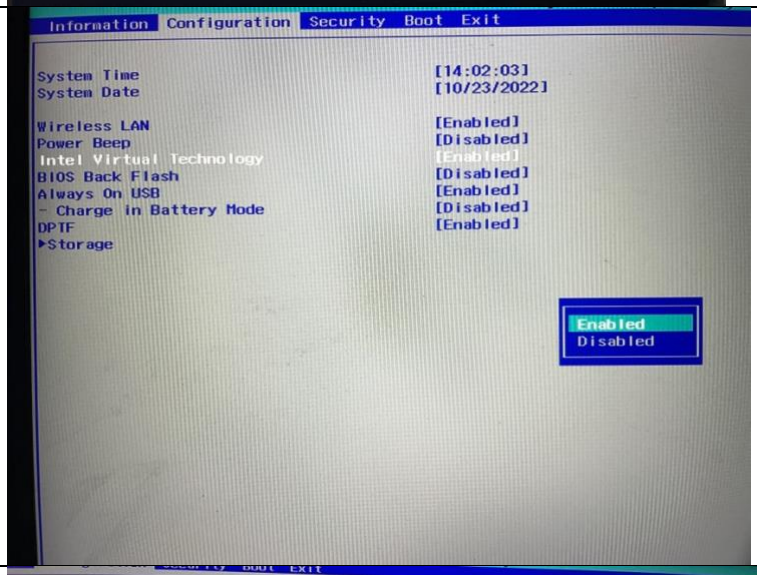
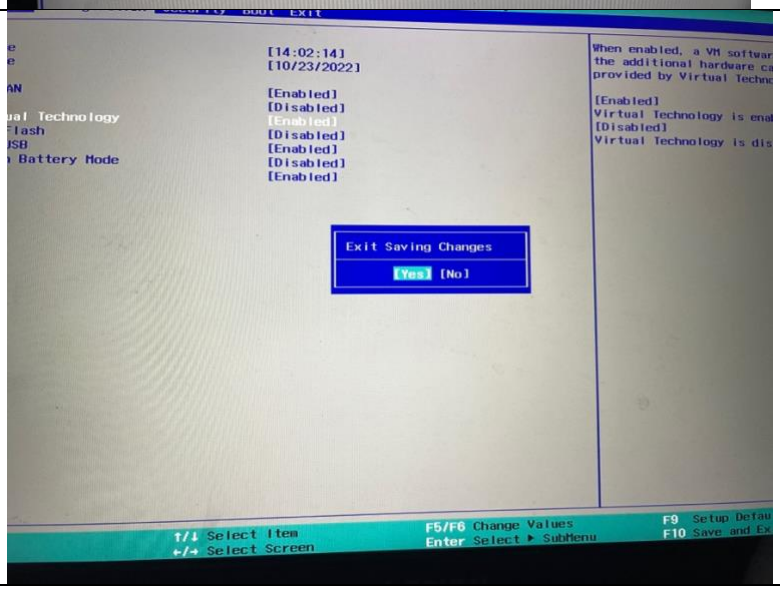
Problems:

Throughout this process, I encountered many problems. Mainly with the installation of VirtualBox and getting the VMs to run properly. The first major problem I encountered was that I had installed the wrong version of VirtualBox (7.0 instead of 6.1) and a lot of the old features that I need have been scrapped in the latest version. The second problem I encountered was after I tried starting the virtual machine. I was greeted by the errors: Not in a hypervisor partition (HVP=0) (VERR_NEM_NOT_AVAILABLE) and VT-x is disabled in the BIOS for all CPU modes (VERR_VMX_MSR_ALL_VMX_DISABLED). After some research I discovered that this error meant that virtualization was disabled on my machine and that I had to go into the BIOS to reenable it.

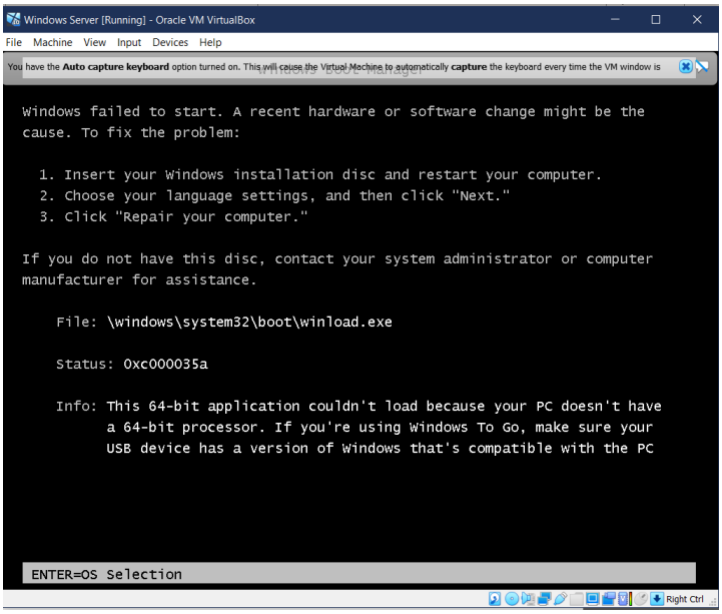


Enabling Virtualization:

My computer, the Legion Y7000P-1060 has a weird boot quirk that doesn't use the keys, instead you have to press a button with a paper clip, after navigating through and enabling virtualization the error didn't appear anymore. By shutting down my laptop and pressing this button, it booted into the BIOS menu.	
You can see here is the BIOS screen, I then navigated to configuration.	

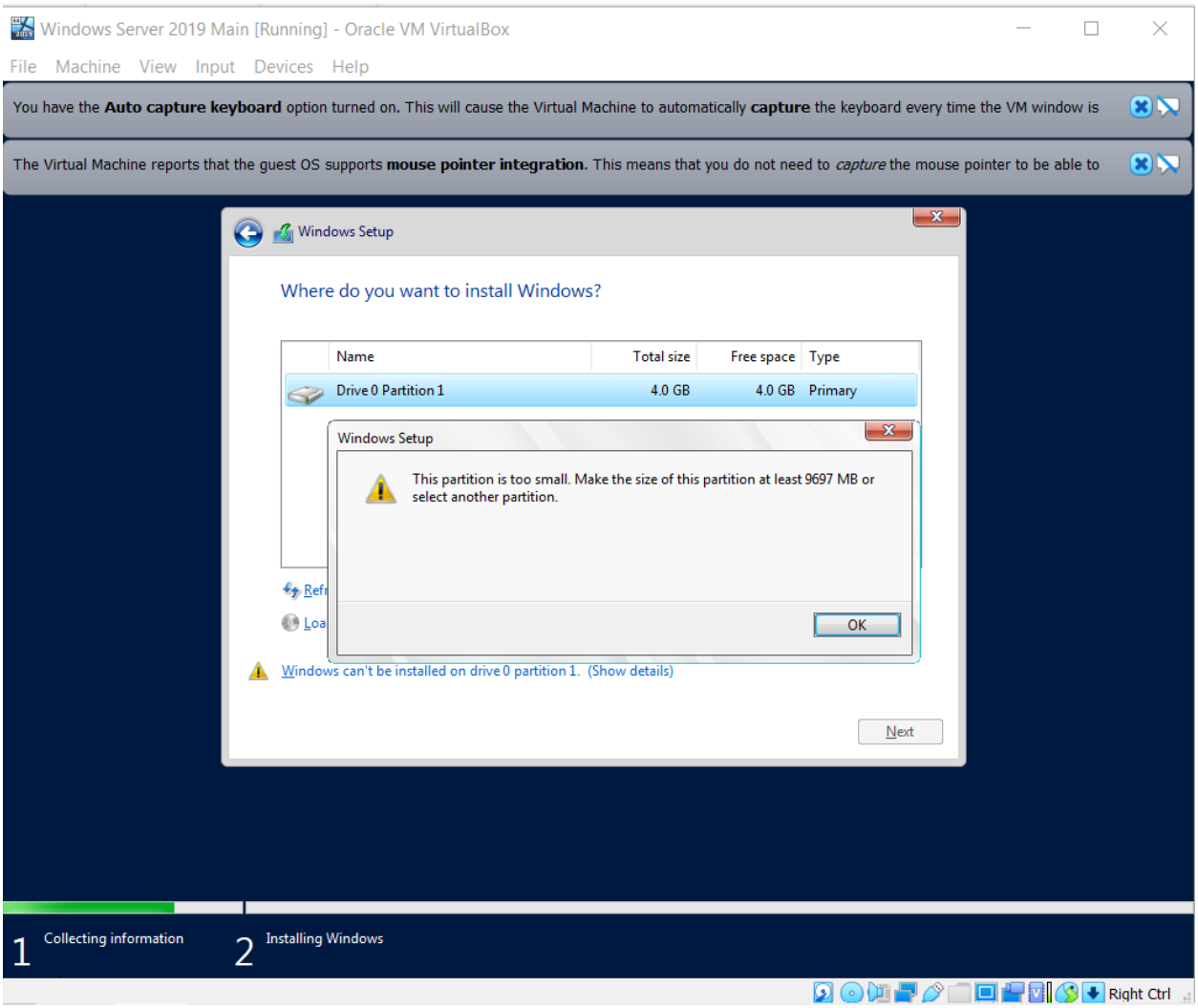
In configuration, I located Intel Virtual Technology. Virtual technology = virtualization for Intel.	
I enabled the Intel Virtual Technology option.	
Saved my changes, which partly resolve the problem.	

The next error I encountered was one of misconfiguration of the Microsoft Server VM. When I tried starting the VM again, it gave me an error stating that “This 64-bit application couldn’t load because your PC doesn’t have a 64-bit processor”. I knew this couldn’t be a problem with my hardware because my computer had like 12 cores and way more than enough processing power since it was made to be a gaming PC. I then got to thinking, and hypothesized that maybe when I created this VM when virtualization was still disabled, it got allocated a really small amount of processing power because the rest of it was locked if virtualization was disabled. Based on that hunch, I decided to delete this VM and create a new one. Immediately I discovered that there were new options available to me including a Windows Server 2019 option that wasn’t there before. When I chose that and started the new VM, I was able to start it up without a problem.



The final configuration error I encountered had to do with disk space. Because I also you this PC as storage for my photography, there was very little space, like less than 4 GB when I started. So I decided to only allocate 4 GB. However, I soon learned that this was way too little space for a Windows Server to run on, and it gave me the respective error of partition is too small. When I created a 20 GB partition instead once I had moved a lot of photos and deleted a ton of apps, the error resolved itself, and I was able to create the server.

Also note, I ran into trouble when I tried to join the domain as I was using the credential for my local PC, not the credentials of the Domain Controller which are required. Once I fixed that I was able to join without a problem.



Conclusion:

Installing a Windows Machine to be a domain controller on a VM is a very effective and efficient way of testing Active directory and group policy objects in a testing environment, where you can work out the problems and bugs in production instead of after the mass implementation rolls out. Active Directory is a crucial part in any organization's management and efficiency. Knowing how to keep them secure is a crucial part of cybersecurity which we will cover in more depth in our next lab about Group Policy Objects.