



CS311 Assignment 1

“Operating System”

Sem 1, 2025

Group Members

1. Shamal Prasad - S11219545
2. Adrian Obadiah - S11198024
3. Shoneel Kumar - S11219651

TABLE OF CONTENT

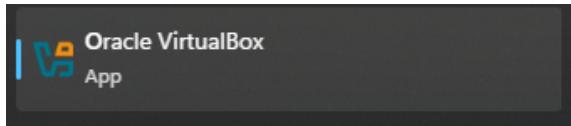
Part A: Virtualization Technology.....	2
Part 1: Installation of virtual machines and Setup.....	2
Installing Windows Server OS CLI Edition.....	2
Installing Windows Server OS GUI Edition.....	11
Installing Kali OS 2025.....	20
Setting Up Windows Server OS CLI IP Address.....	33
Setting Up Windows Server OS GUI IP Address.....	39
Setting Up Kali OS IP Address.....	44
Establishing connection between Various OS.....	46
Pinging from Windows Server OS CLI To Windows Server OS GUI.....	46
Pinging from Windows Server OS CLI To Kali OS.....	47
Pinging from Kali OS To Windows Server OS GUI.....	47
Pinging from Kali OS To Windows Server OS CLI.....	48
Pinging from Windows Server OS GUI To Kali OS.....	49
Pinging from Windows Server OS GUI To Windows Server OS CLI.....	49
Setup Primary domain for Windows Server GUI.....	50
Setting Up Users In Windows GUI Server OS.....	67
Creating a Shared Folder in GUI Server OS.....	70
Accessing the Shared Folder of Server OS GUI from Kali OS.....	75
Creating Users, Shared Folder In Kali OS and Accessing from GUI Server OS.....	77
Creating Users, Shared Folder In CLI Server OS and Accessing from Kali OS.....	84
Part 2: Kali LInux Versus Windows Server OS.....	86
Security Features.....	86
User Management And Access Control.....	87
Design and User Interface.....	88
Part B: Payroll System.....	89
Code: Below is the code for the payroll system.....	89
Result (Kali OS).....	91
Result (Ubuntu OS).....	91
Part C: Critical Thinking & Analysis.....	93
Question 1.....	93
Question 2.....	94
Reference.....	95

Part A: Virtualization Technology.

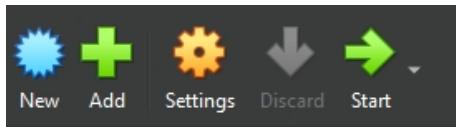
Part 1: Installation of virtual machines and Setup.

Installing Windows Server OS CLI Edition

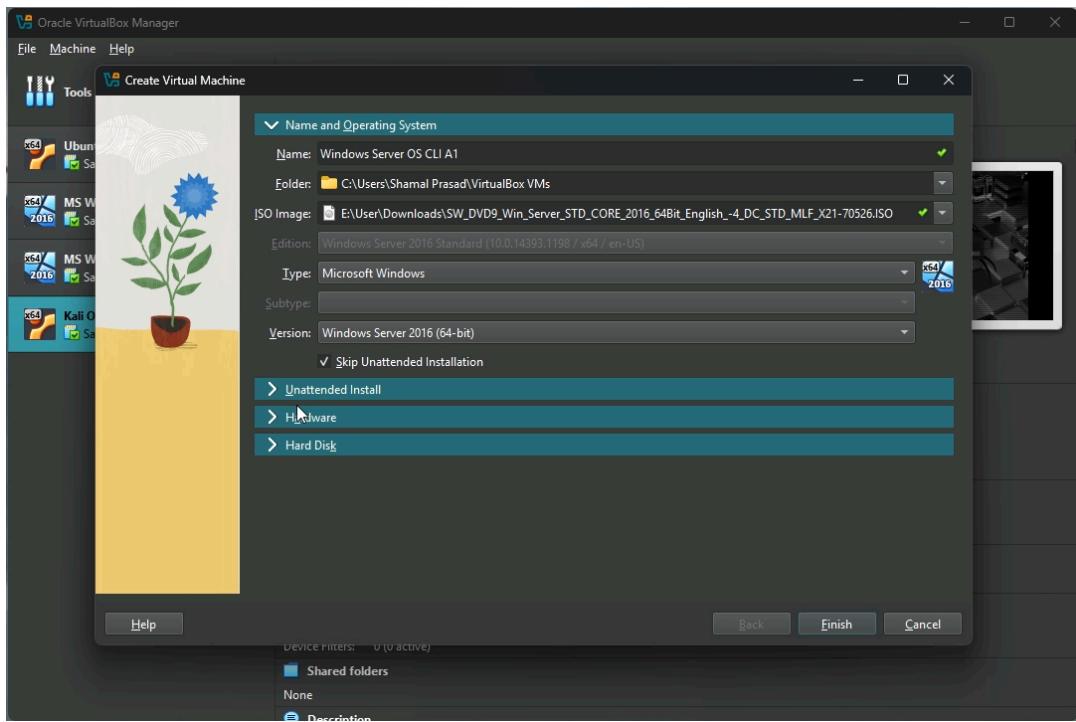
Step 1: Launch Oracle Virtualbox.



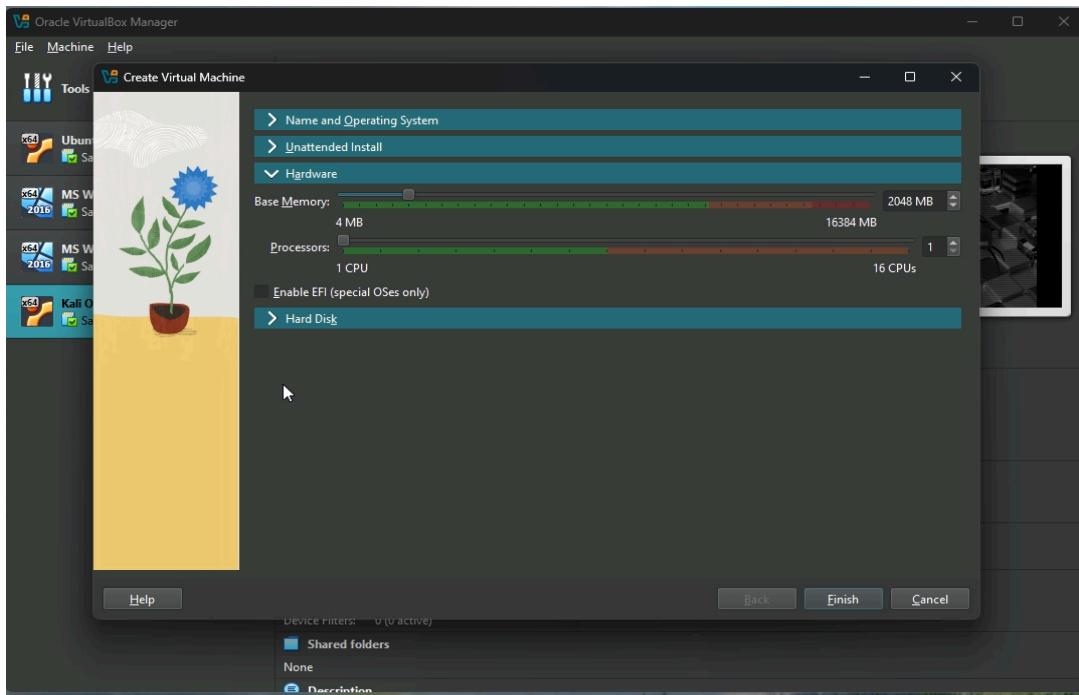
Step 2: In the VirtualBox Manager window, click the "New" button to start the process of creating a new virtual machine.



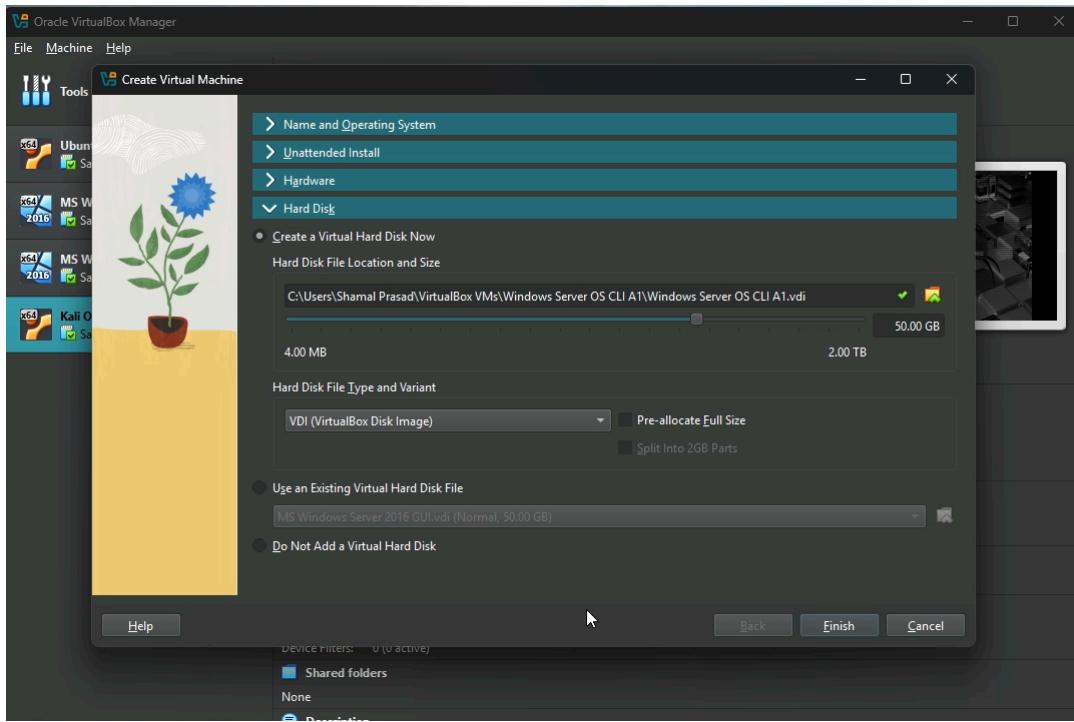
Step 3: In the Create Virtual Box Machine, enter the name of the guest OS. Click the drop down menu for the ISO image and select your desired ISO image that contains the OS that you want to install. Ensure that the Skip Unattended Installation is checked.



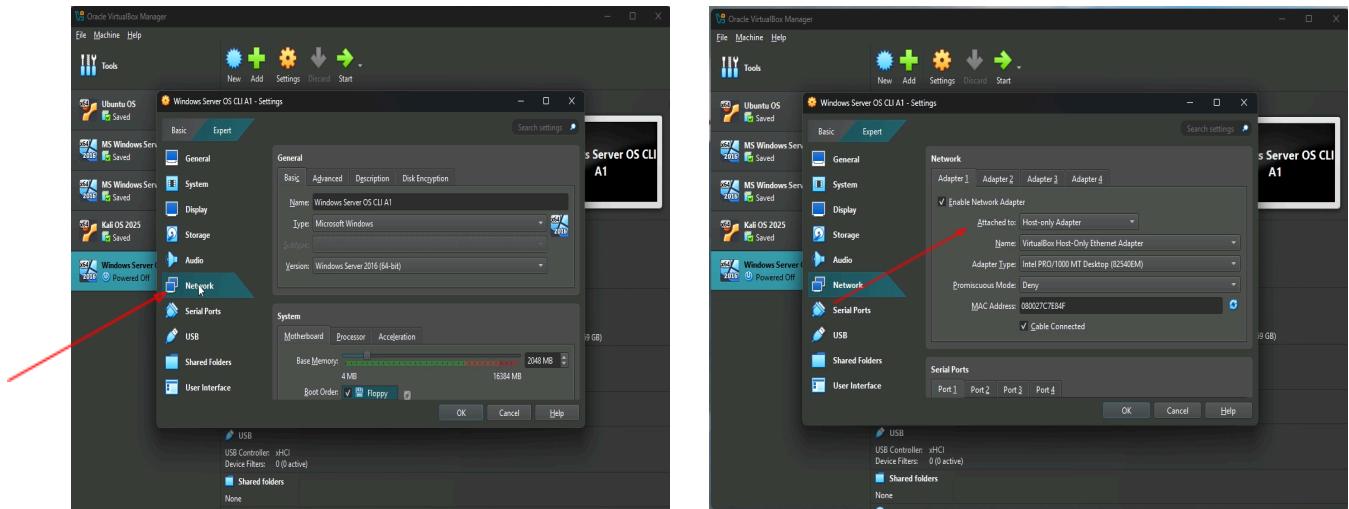
Step 4: Configure the hardware by choosing the right amount of memory and CPU cores to ensure the system runs smoothly. Minimum amount of required RAM should be 1GB. But you can allocate more than that as you see fit. Just ensure to not over allocate as it will hinder the performance of your main computer. The core can be left at 1 as it is enough.



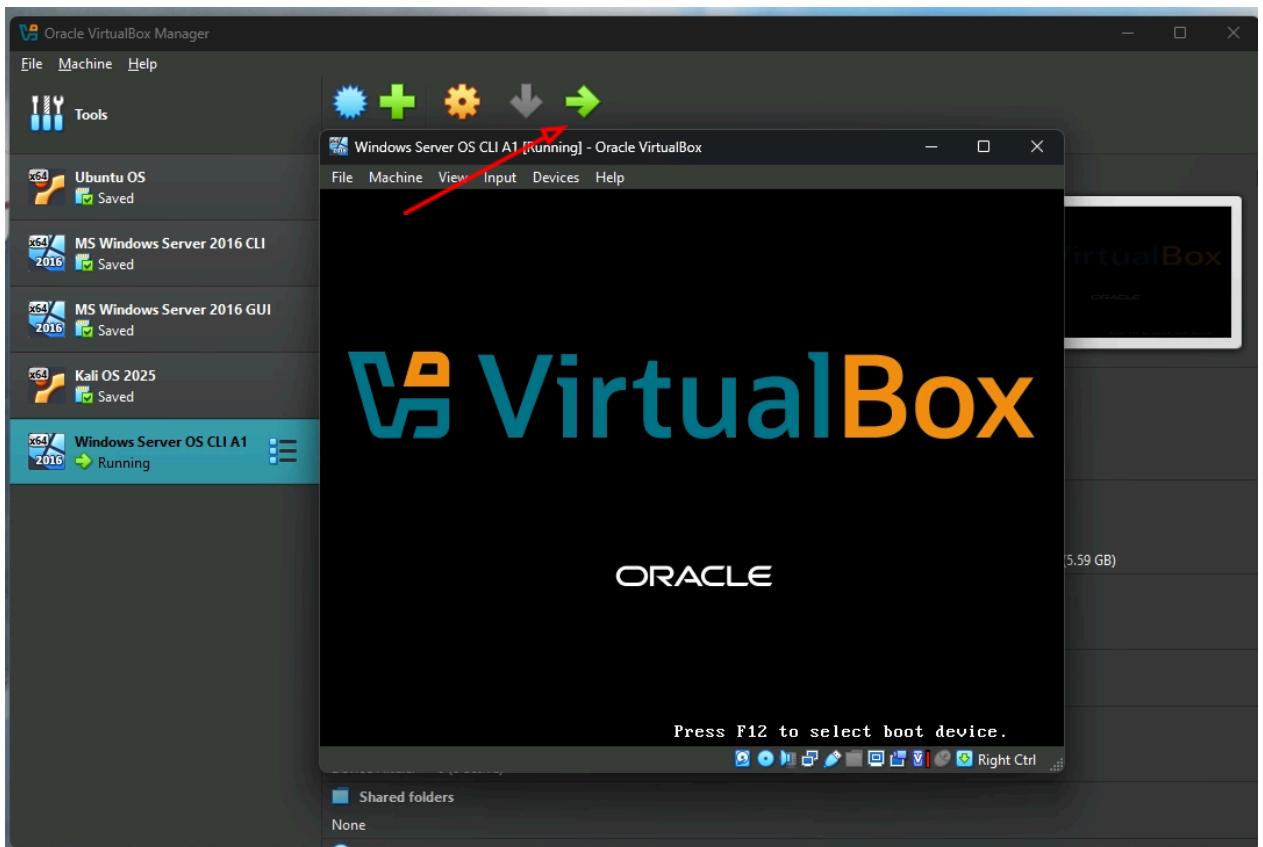
Step 5: Configure the size of the hard drive that the OS can use, changing the slider or typing the value out. Around 50 Gb of space should be enough for the server OS CLI Edition as it does not require a lot of resources to function properly. Once allocated, click Finish.



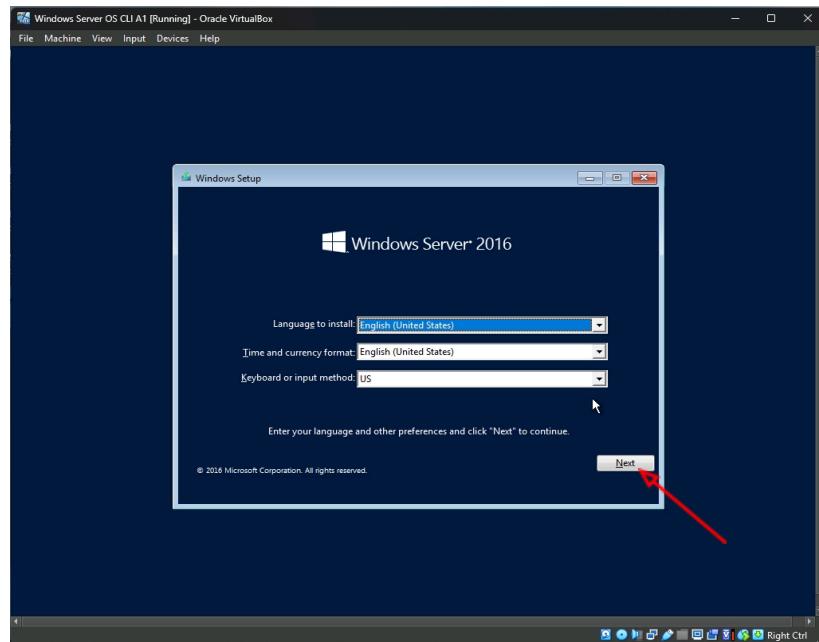
Step 6: Before running the Server OS, some settings need to be changed. Open the Settings in the VirtualBox Manager window and locate the network setting and ensure you are on Expert Mode. Once on the Network tab, change the network adapter to Host-only Adapter and click OK.



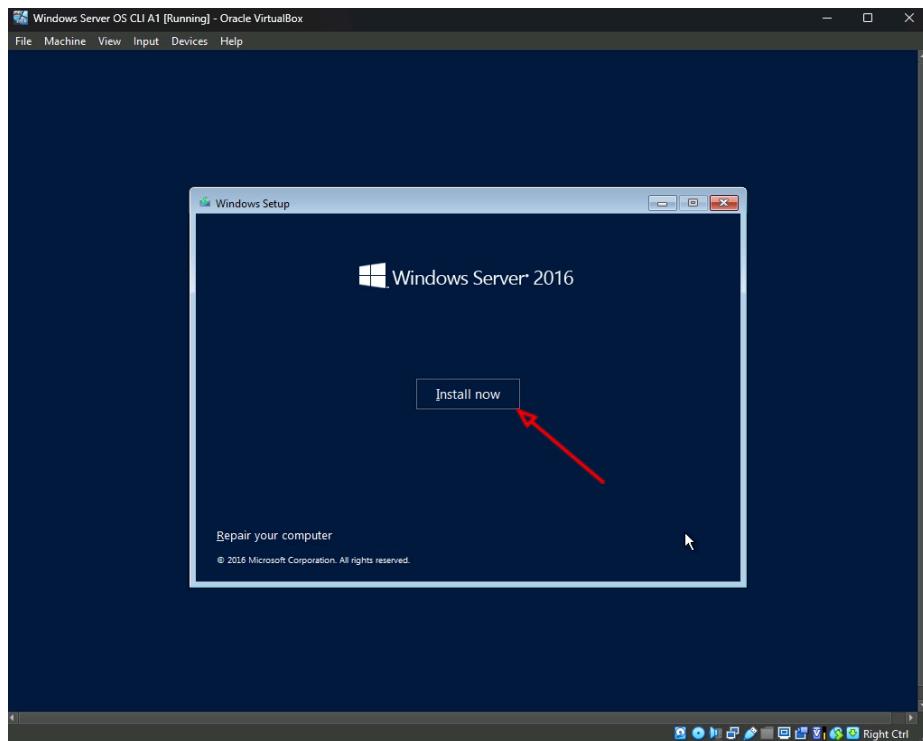
Step 7: Once you have configured the setting, you can now go ahead and start the OS by clicking the start Arrow in the VirtualBox Manager window at the top.



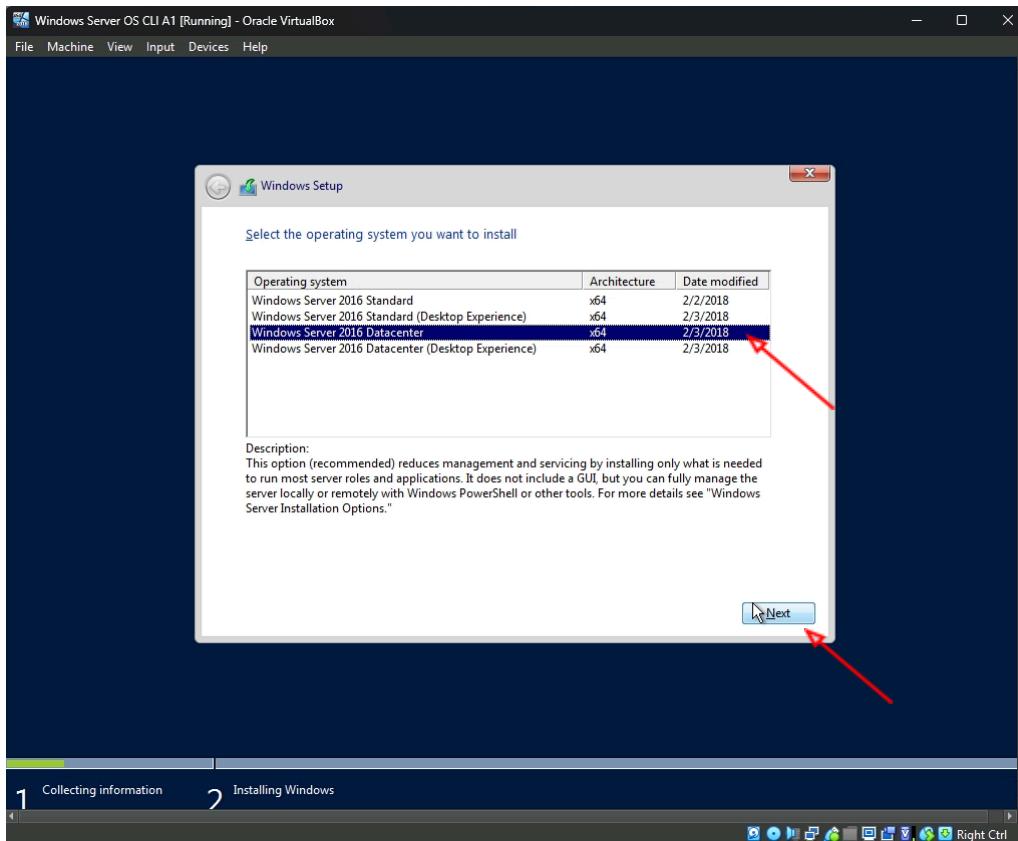
Step 8: Now the OS setup and installation process will begin. Follow the setup instructions and fill in information as required. Click Next.



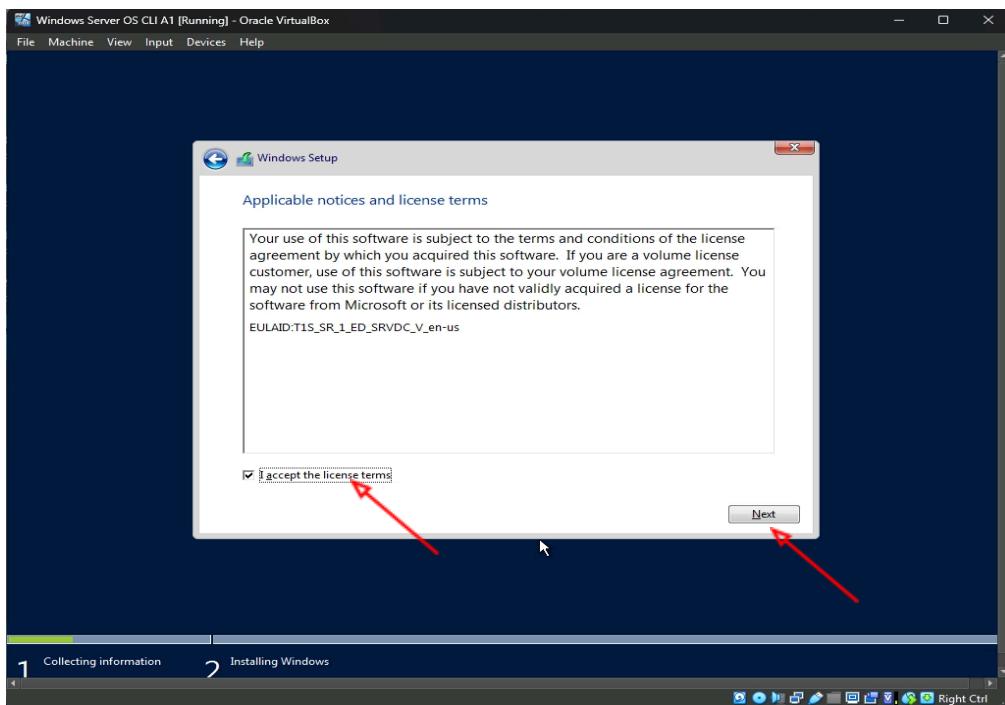
Step 9: Press Install Now.



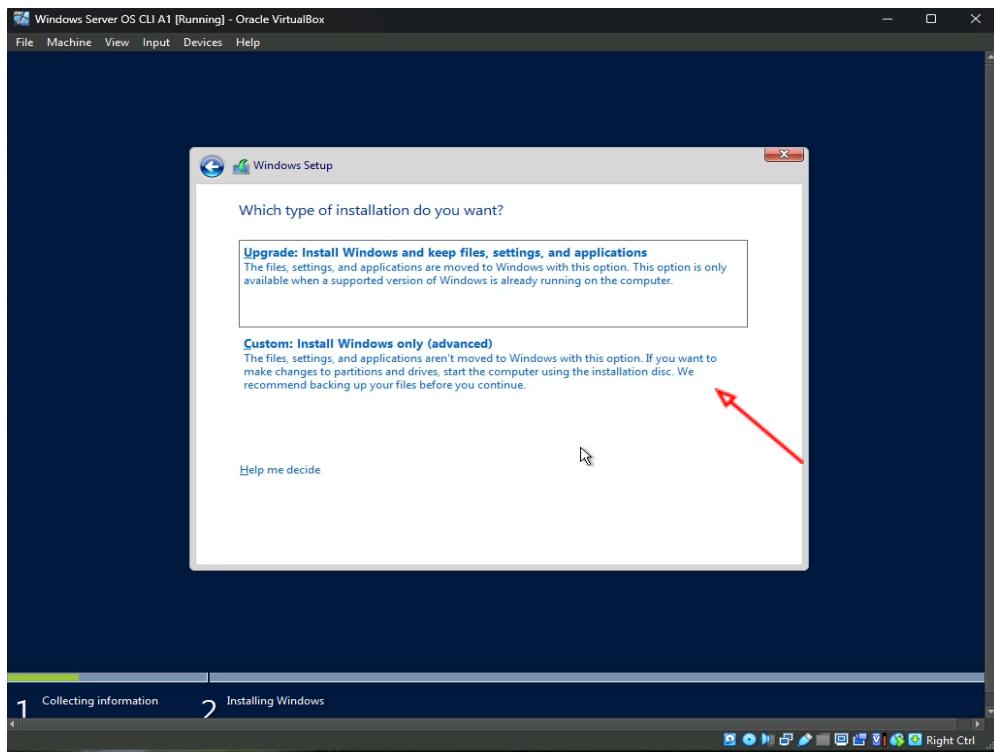
Step 10: Select Windows Server 2016 Datacenter Edition and click Next.



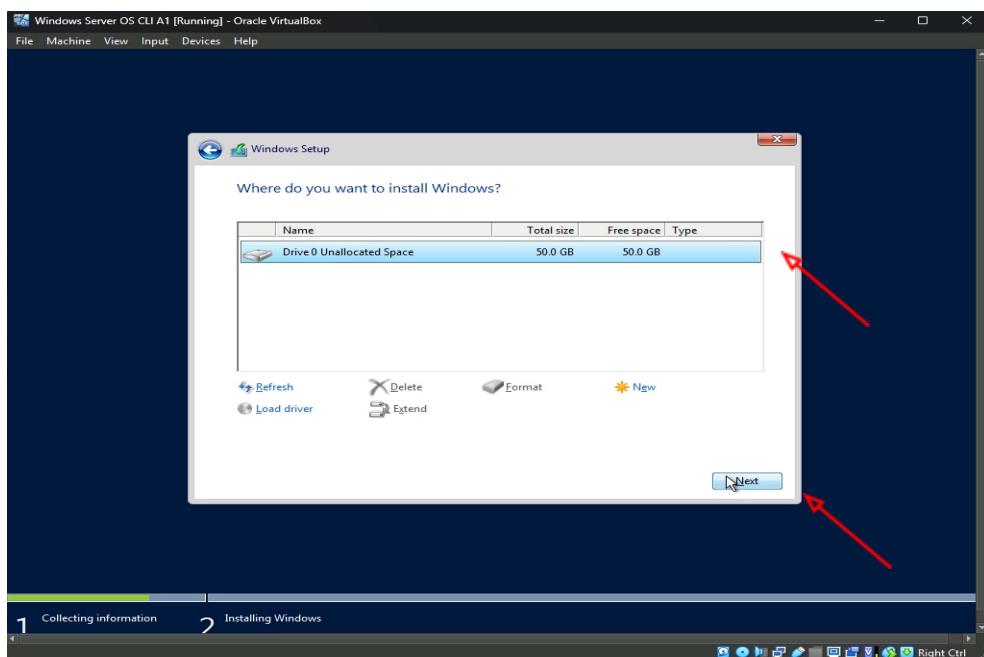
Step 11: Check the License Agreement Box and Click Next.



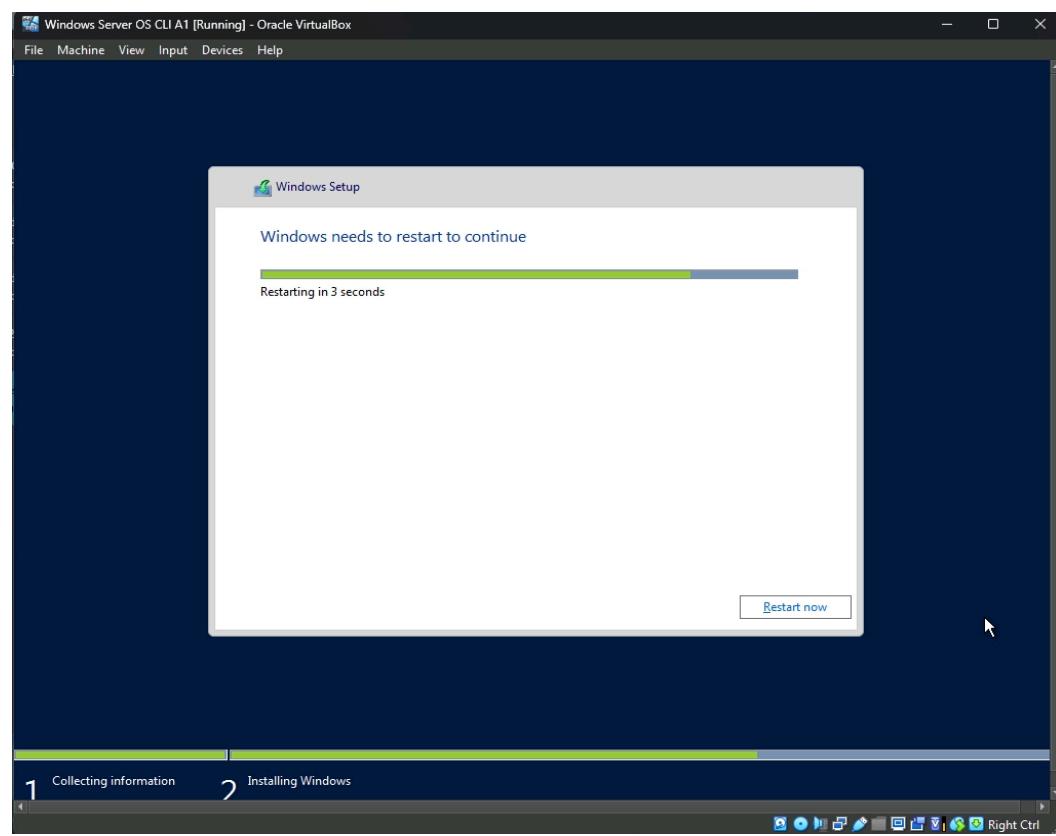
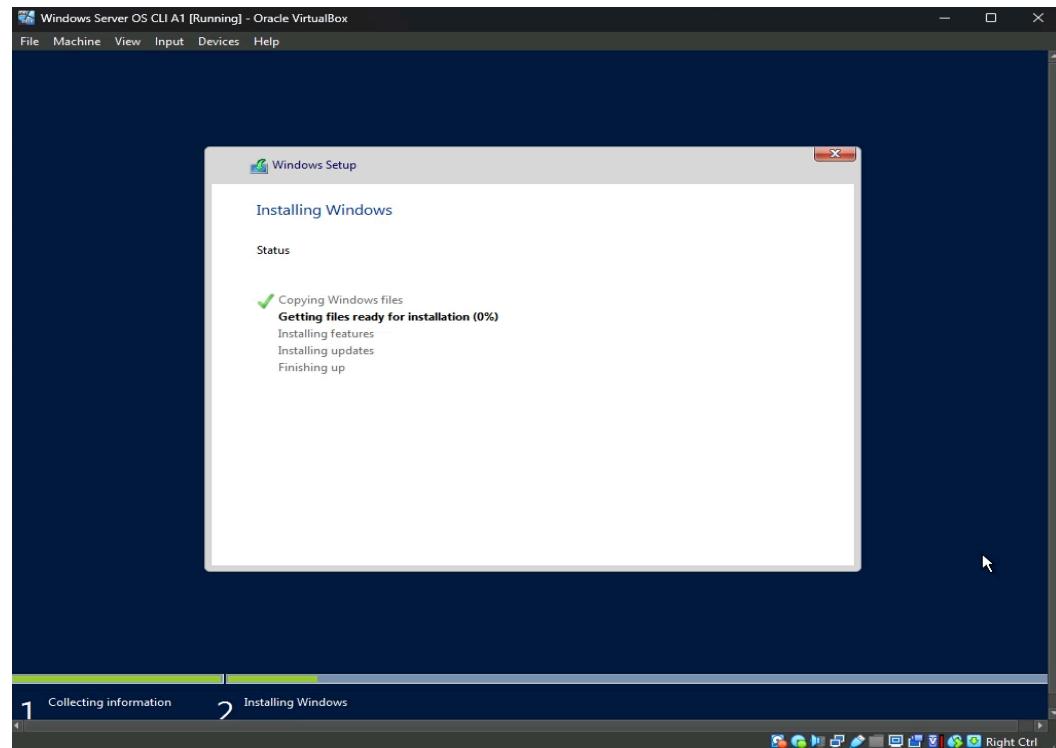
Step 12: Choose the custom installation type as it will install the OS from the very beginning, the other option is repair the OS if it has encountered an error. You can select the following option by double clicking on the option.



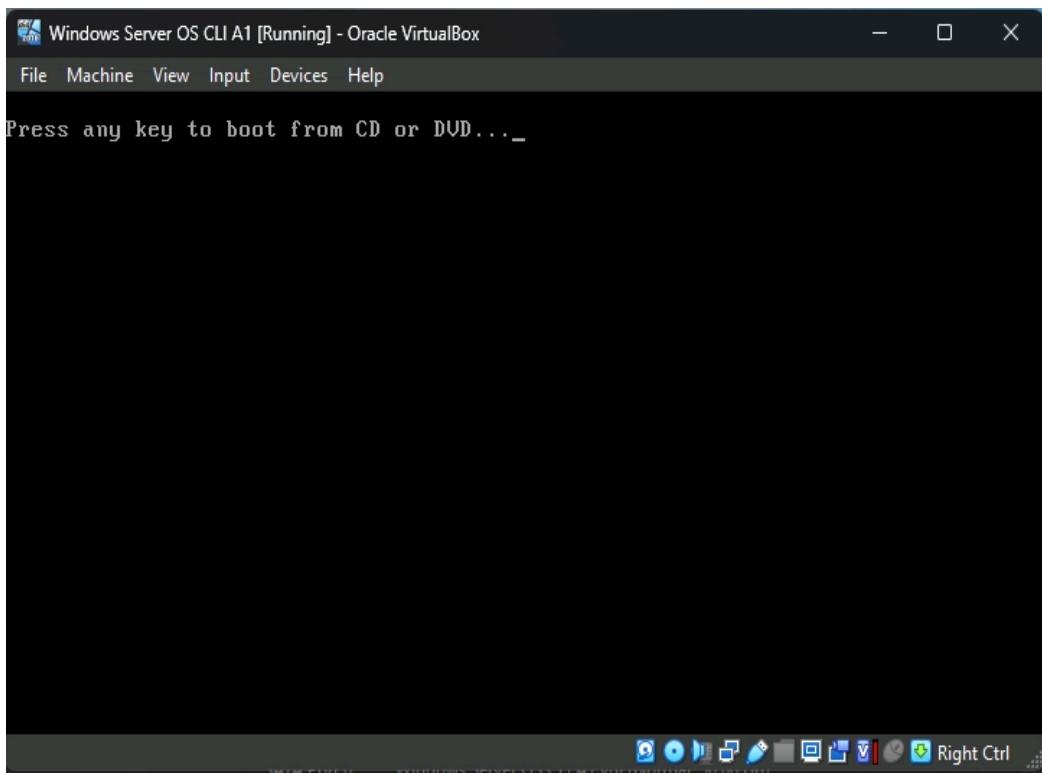
Step 13: Select your main drive partition, which should only be one , and click Next.



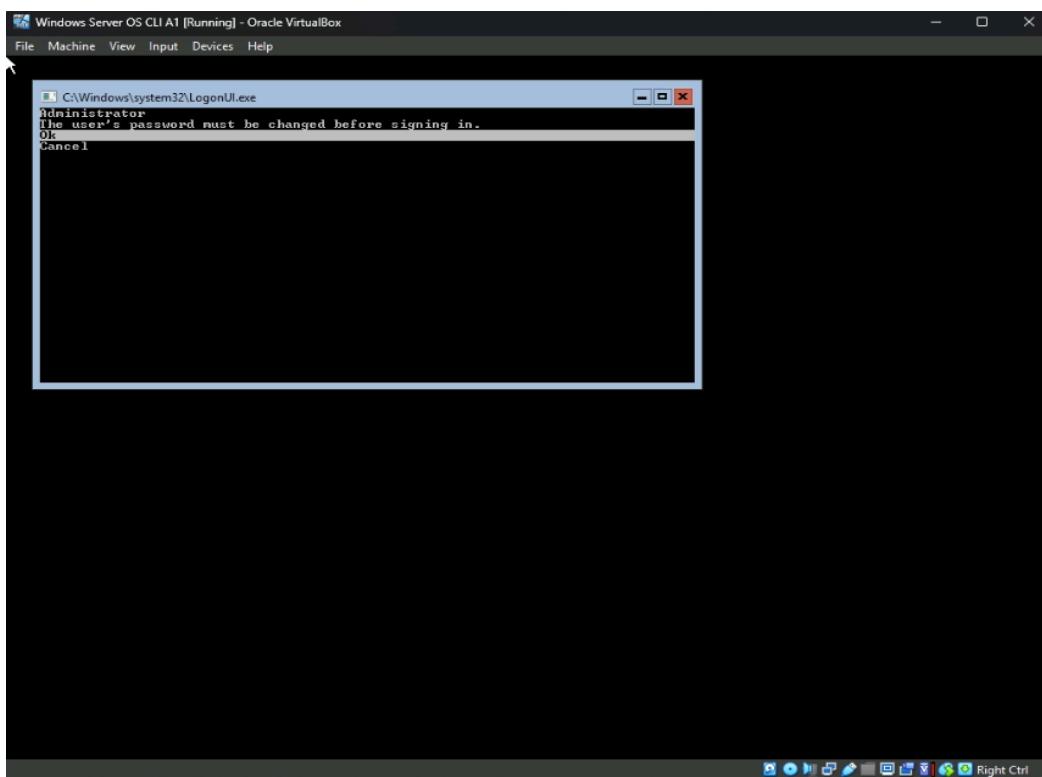
Step 14: Wait for the installation to complete and at the end of the installation, the OS will reboot itself.



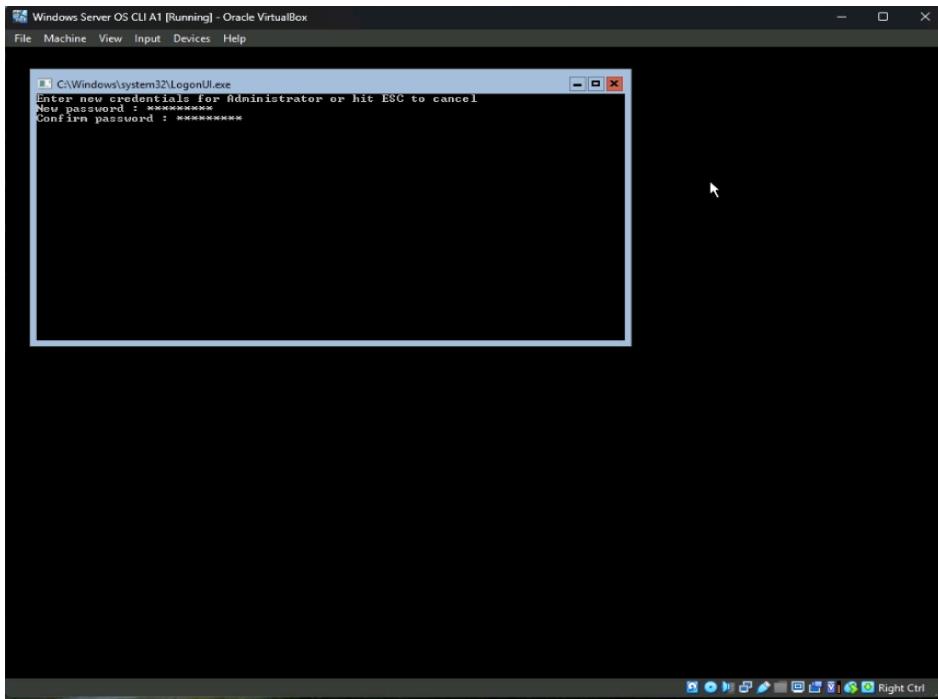
Step 15: During the Screen shown below, DO NOT press anything and wait, the OS startup will begin automatically.



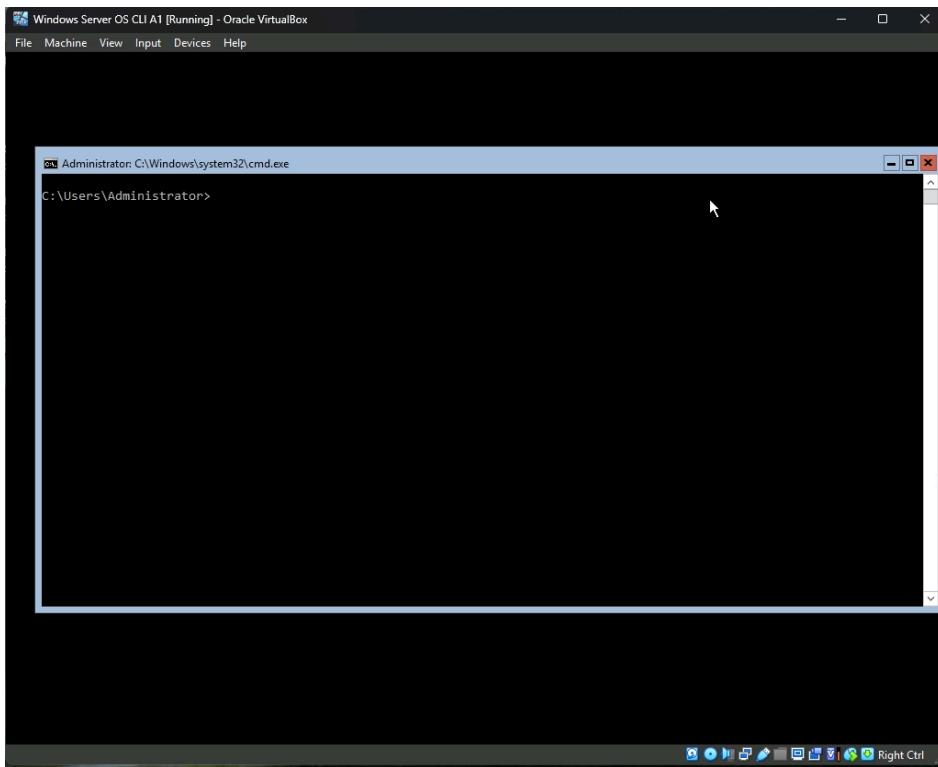
Step 16: The following first time logon window will inform you that you need to set up a password for your Administrator account. Press Enter on your keyboard



Step 17: Create a password for your account.

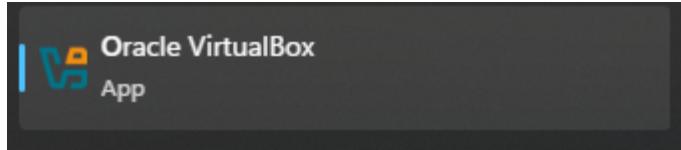


Step 18: You will now be presented with the command line prompt which will be the only way to interact with the OS since in the datacenter edition the OS is stripped down to include core features and will be less resource intensive.

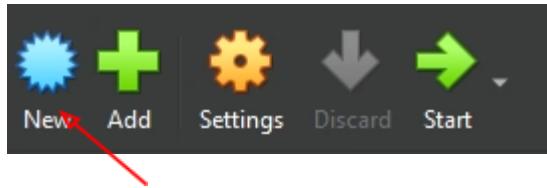


Installing Windows Server OS GUI Edition

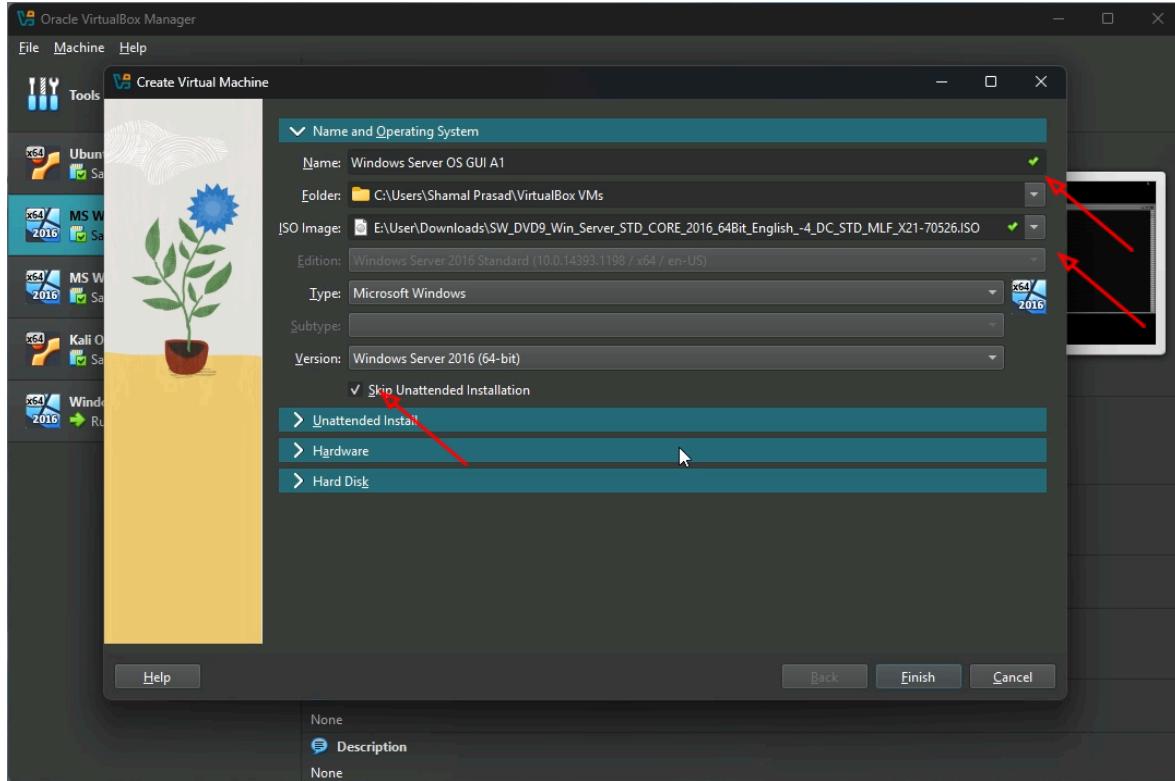
Step 1: Launch Oracle Virtualbox.



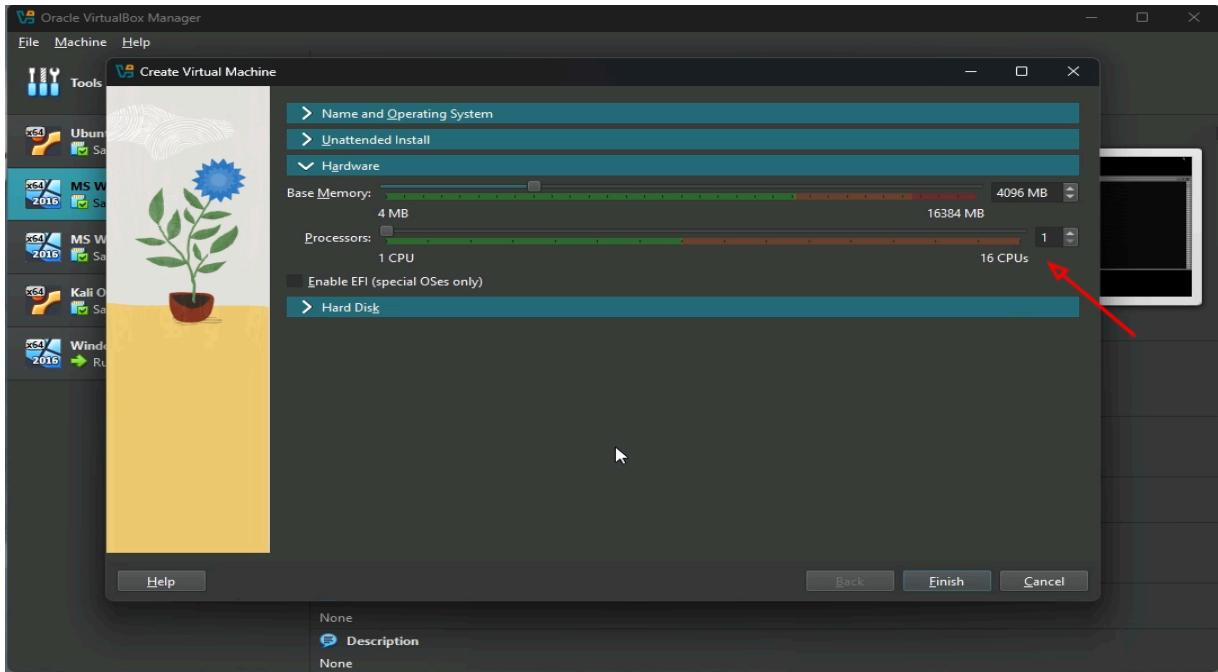
Step 2: In the VirtualBox Manager window, click the "New" button to start the process of creating a new virtual machine.



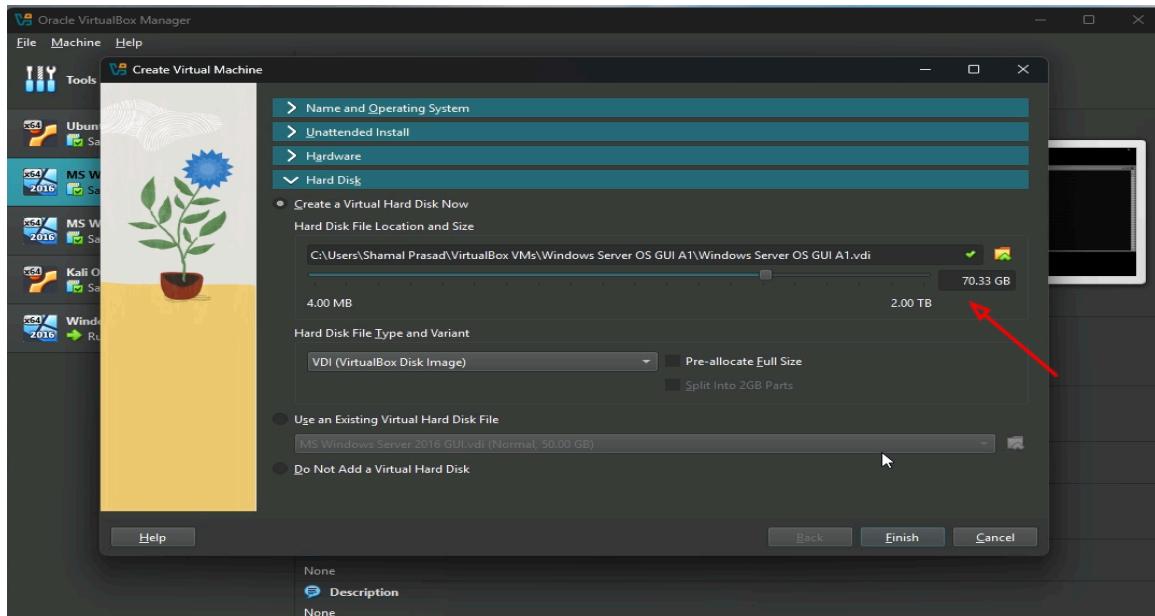
Step 3: In the Create Virtual Box Machine, enter the name of the guest OS. Click the drop down menu for the ISO image and select your desired ISO image that contains the OS that you want to install. Ensure that the Skip Unattended Installation is checked.



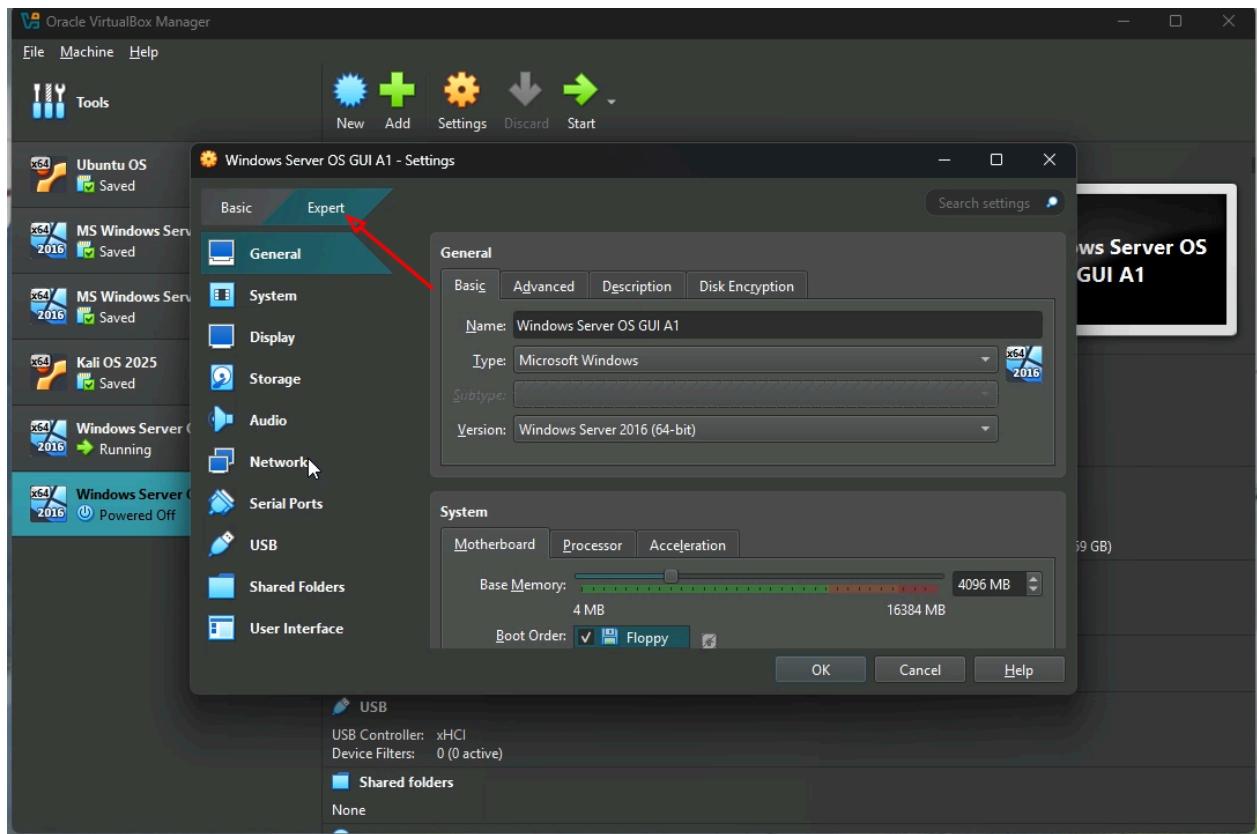
Step 4: Configure the hardware by choosing the right amount of memory and CPU cores to ensure the system runs smoothly. Minimum amount of required RAM should be 2GB, but you can allocate more than that as you see fit, but ensure to not over allocate as it will hinder the performance of your main computer. The core can be left at 1 as it is enough.



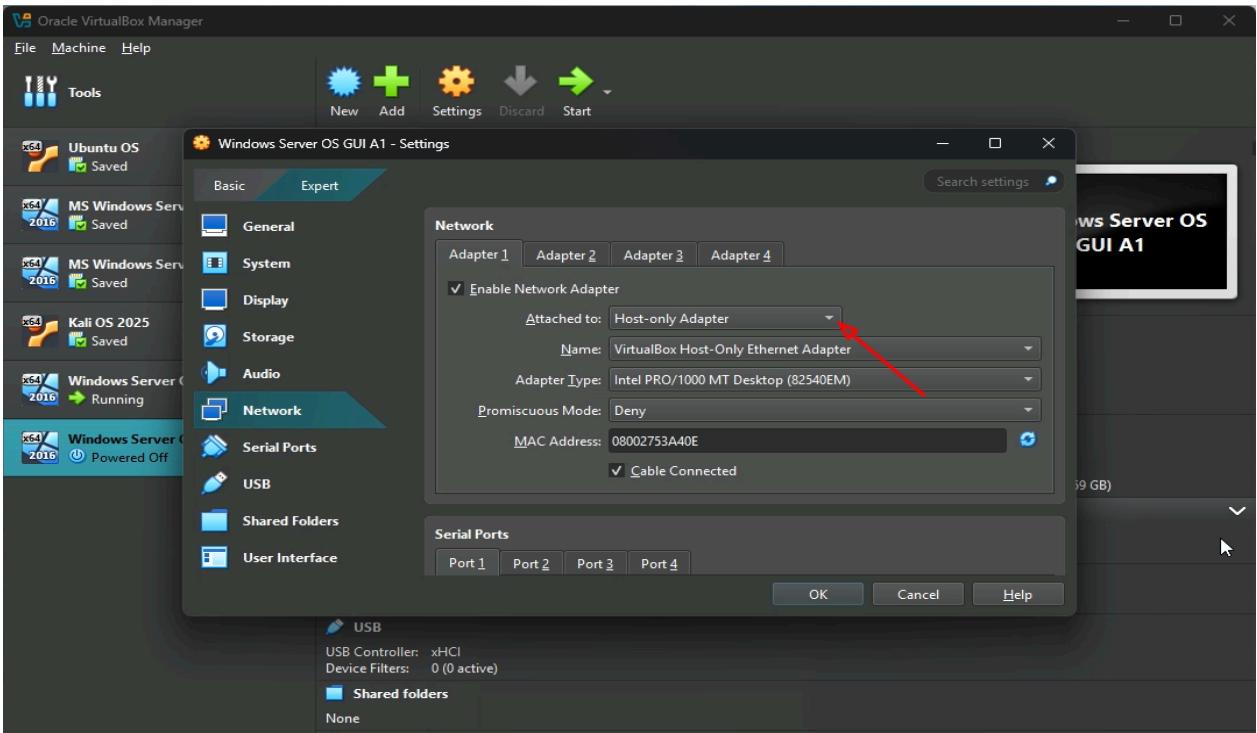
Step 5: Configure the size of the hard drive that the OS can use changing the slider or typing the value out. Around 70 Gb of space should be enough for the server OS GUI Edition. Once allocated, click Finish.



Step 6: before running the Server OS, some settings need to be changed. Open the Settings in the VirtualBox Manager window and locate the network setting and ensure you are on Expert Mode.



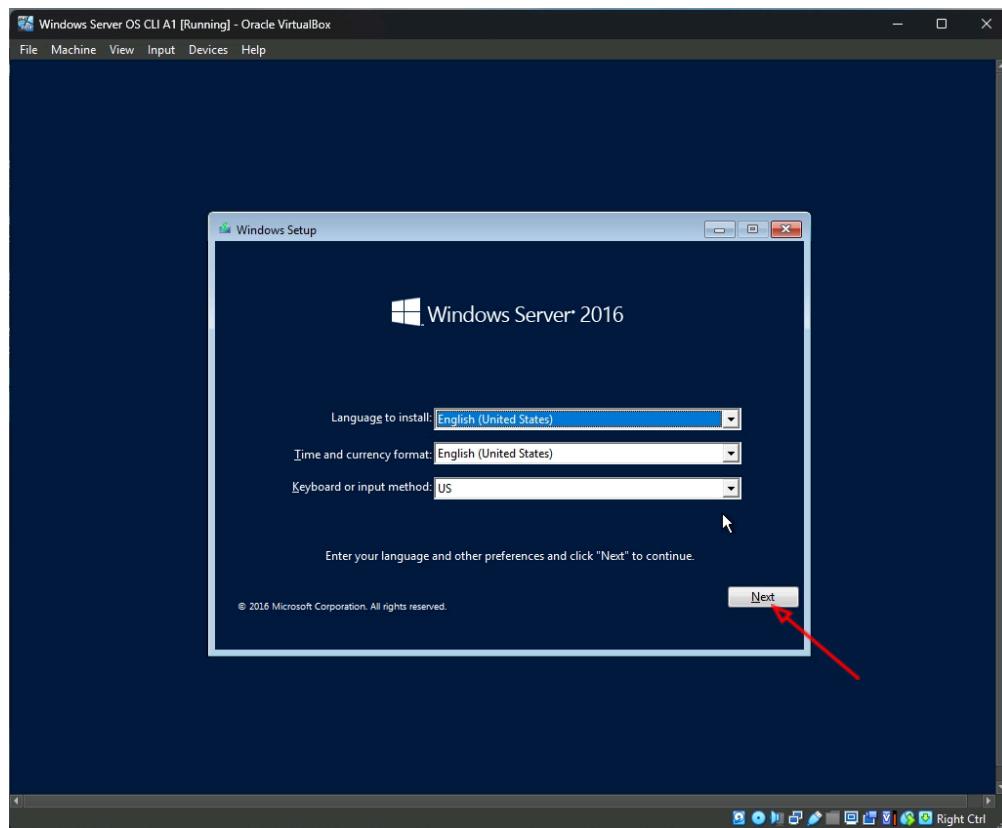
Once on the Network tab, change the network adapter to Host-only Adapter. Click OK.



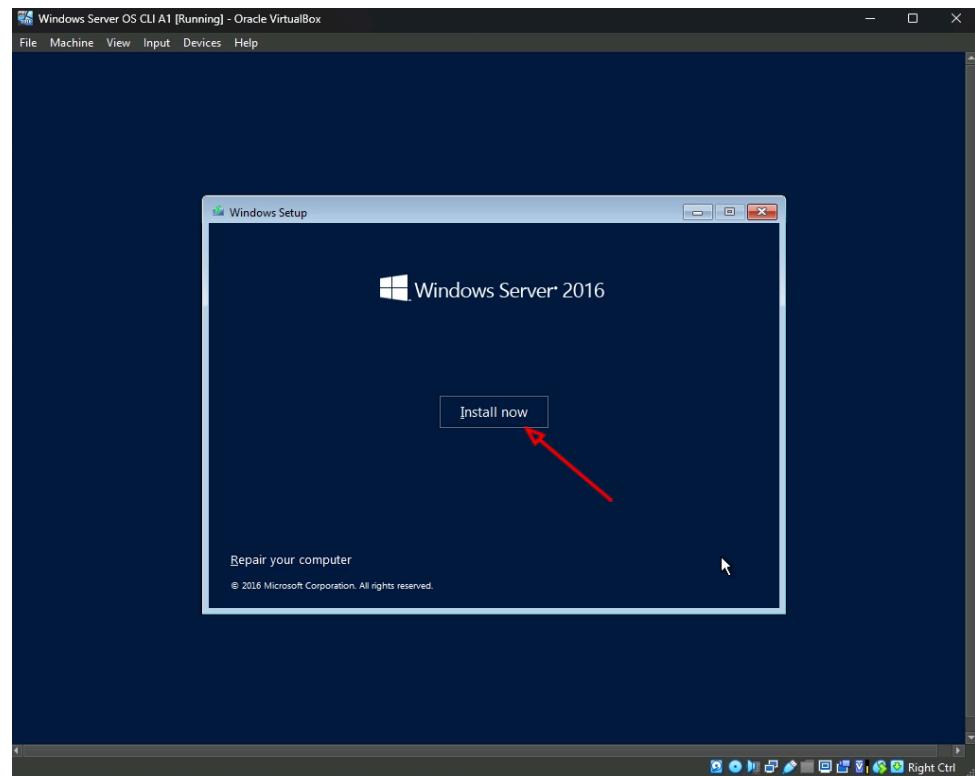
Step 7: Once you have configured the setting, you can now go ahead and start the OS by clicking the start Arrow in the VirtualBox Manager window at the top.



Step 8: Now the OS setup and installation process will begin. Follow the setup instructions and fill in information as required. Click Next.

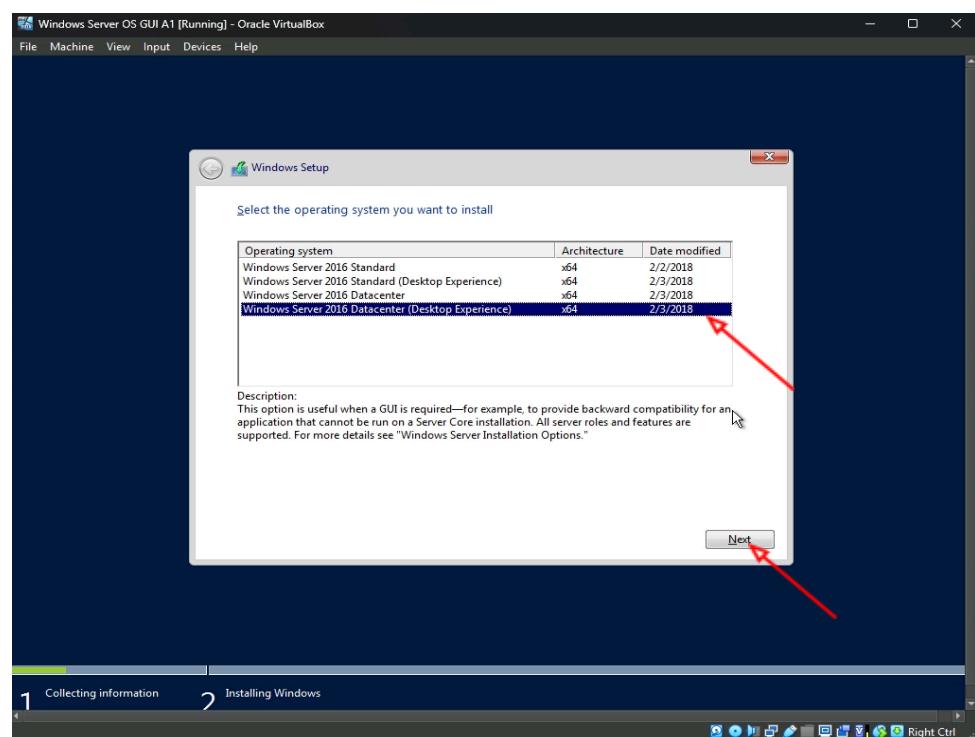


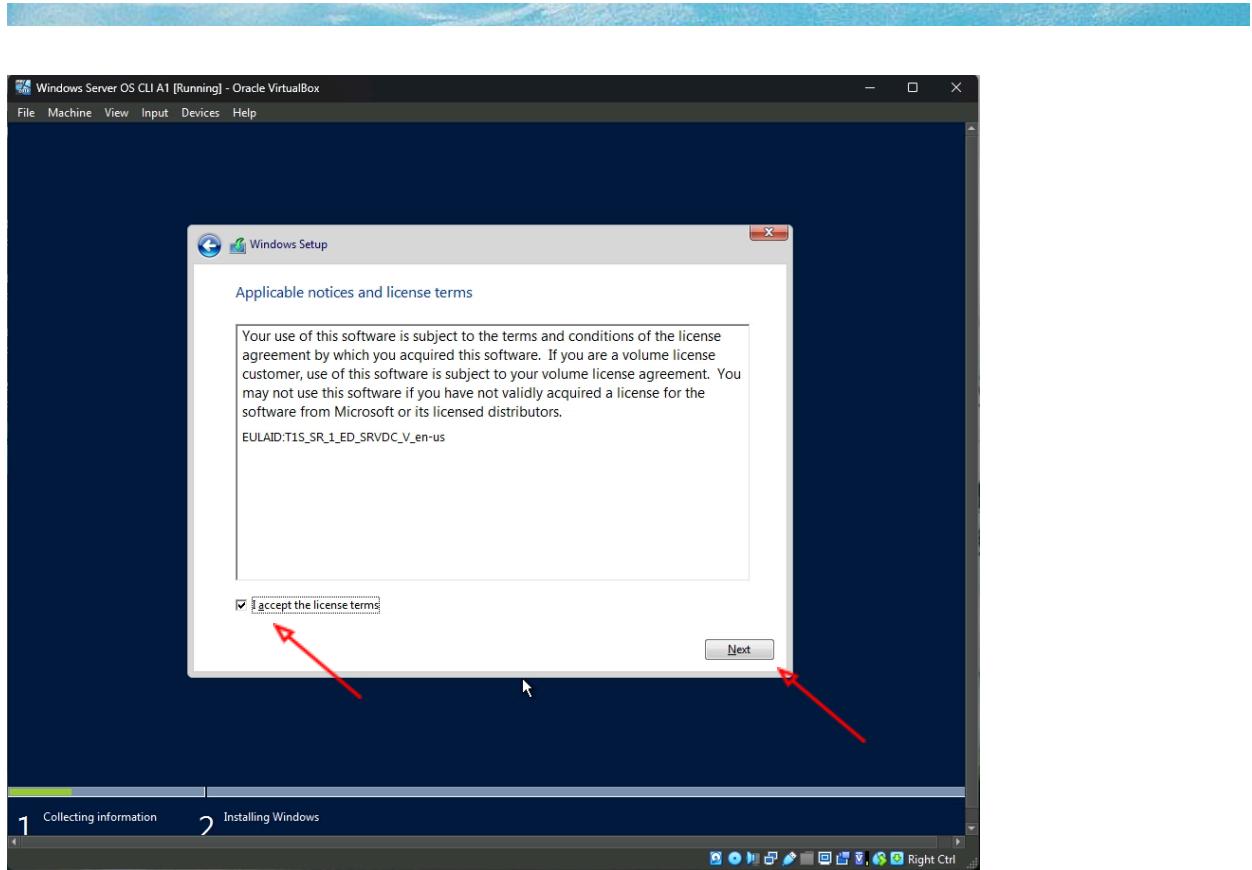
Step 9: Press Install Now.



Step 10: Select Windows Server 2016 Datacenter Edition/Desktop Experience) and click Next.

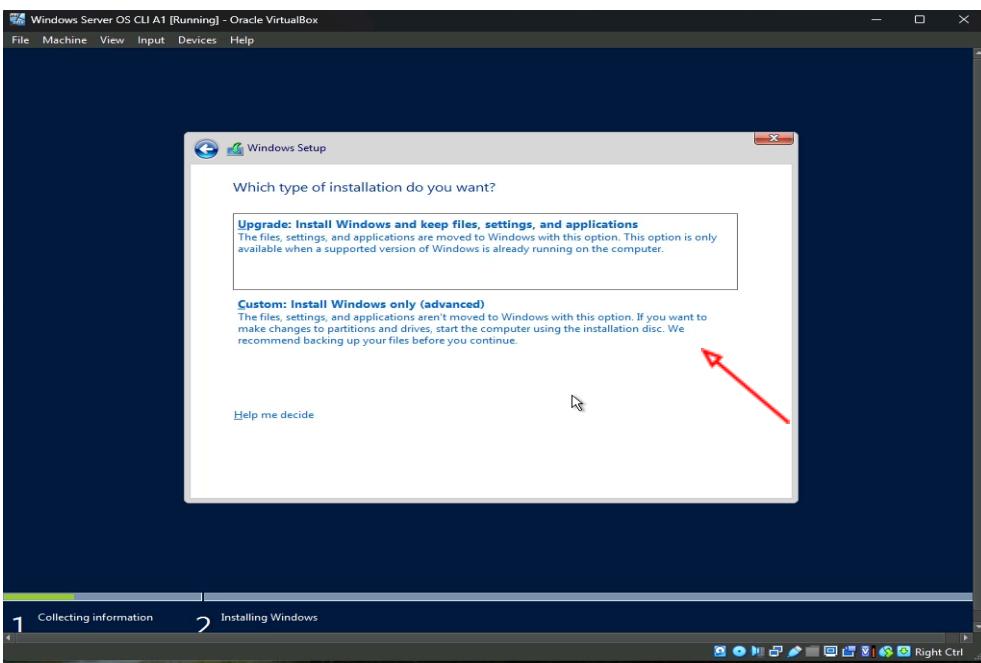
Step 11: Check the License Agreement Box and Click Next.

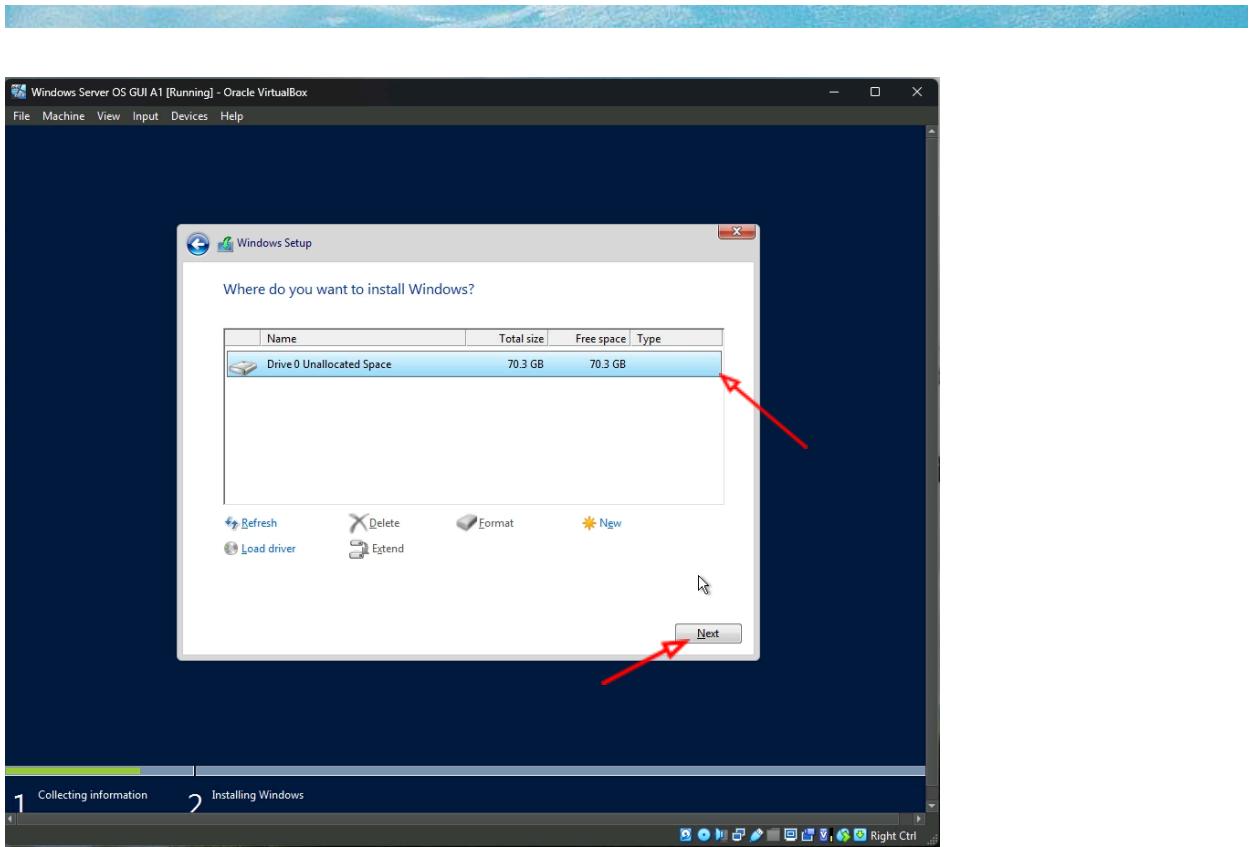




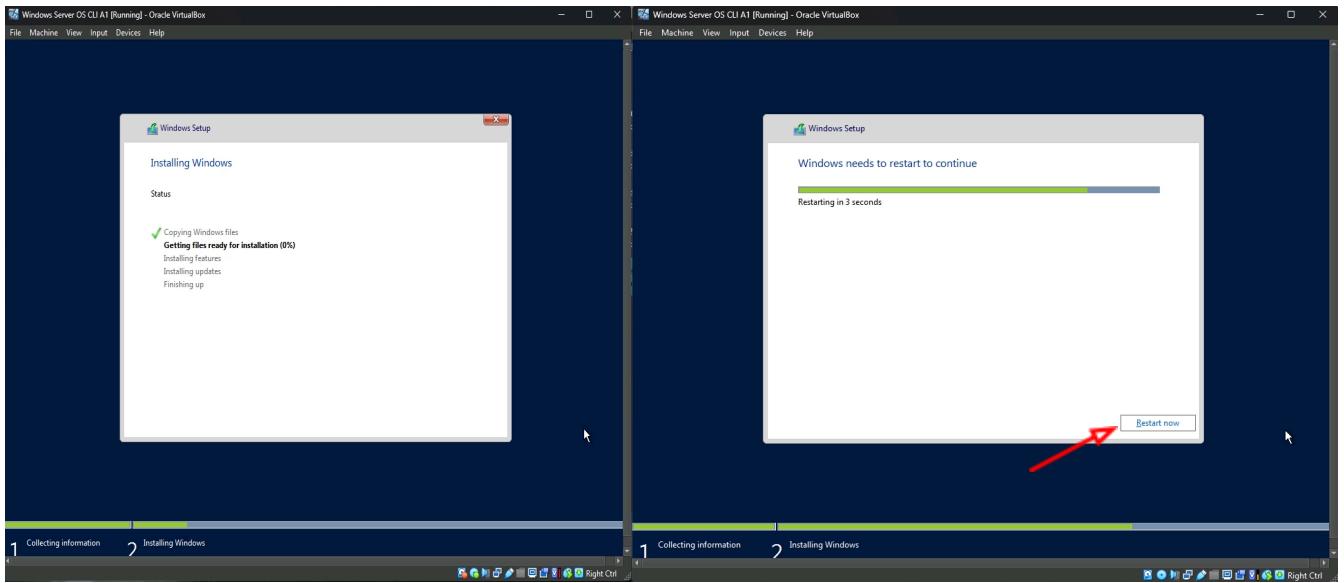
Step 12: Choose the custom installation type as it will install the OS from the very beginning, the other option is repair the OS if it has encountered an error. You can select the following option by double clicking on the option using your mouse.

Step 13: Select your main drive partition, which should only be one, and click Next.

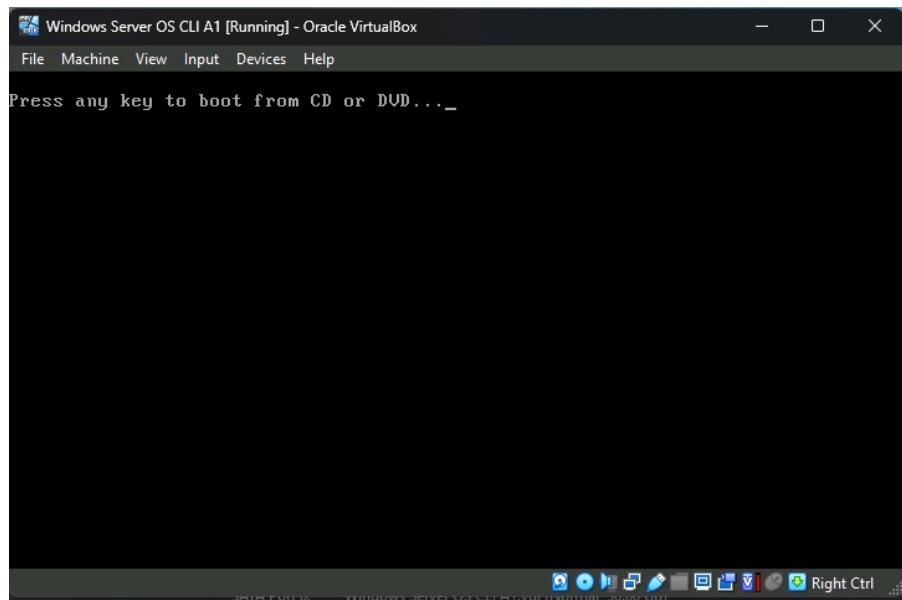




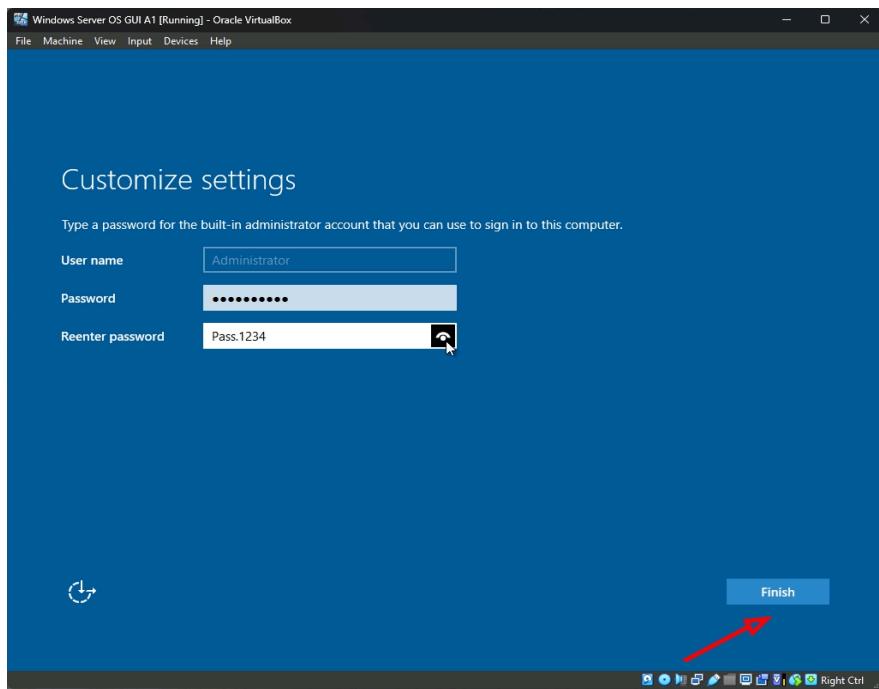
Step 14: Wait for the installation to complete and at the end of the installation, the OS will reboot itself.



Step 15: During the Screen shown below, **DO NOT** press anything and wait, the OS startup will begin automatically.



Step 16: After the setup process you will be directed to the screen shown below where you will need to set up a password for the Administrator account. Create your password and press Finish.



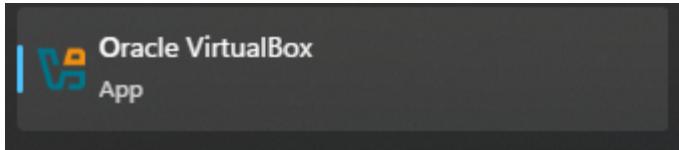


Step 17: Sign in to your Admin Account.

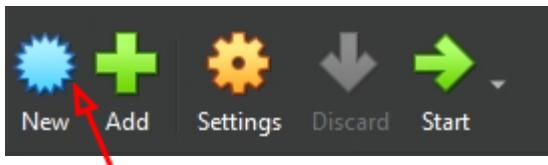


Installing Kali OS 2025

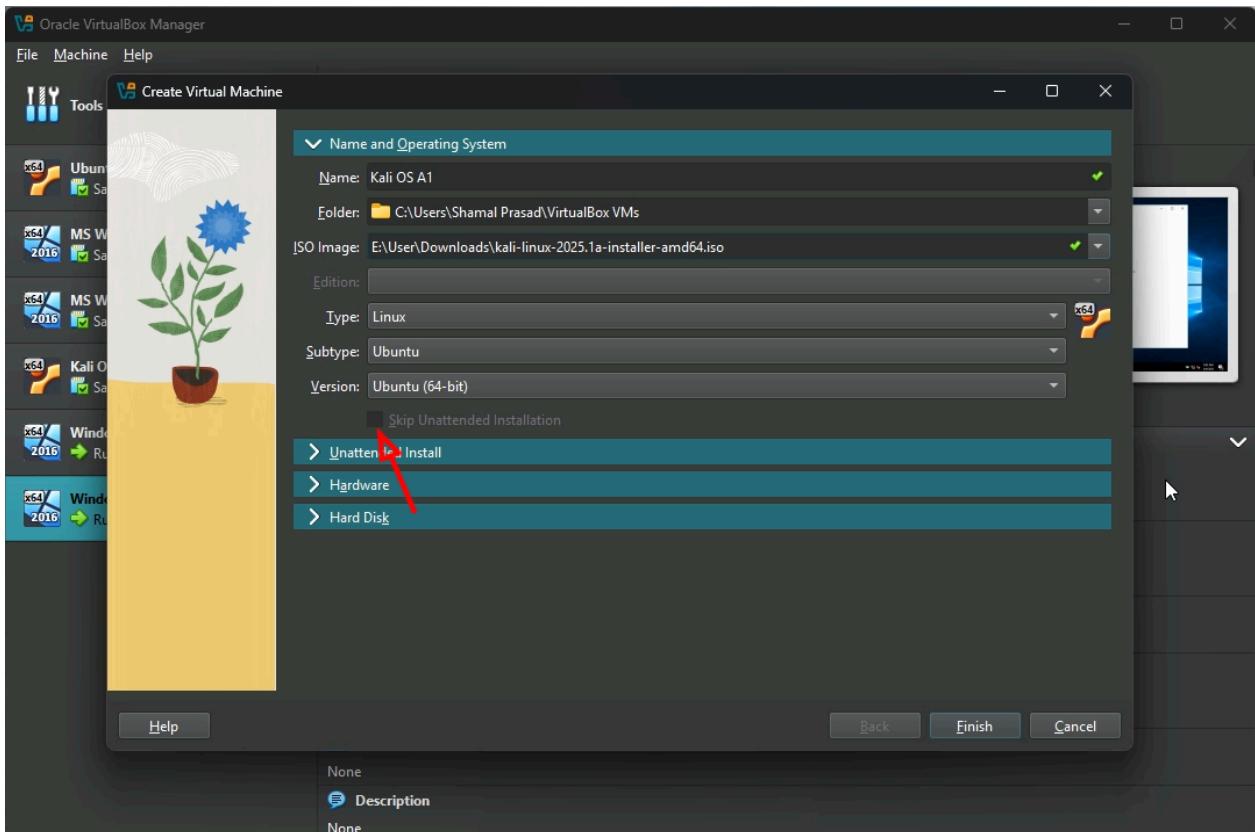
Step 1: Launch Oracle VirtualBox.



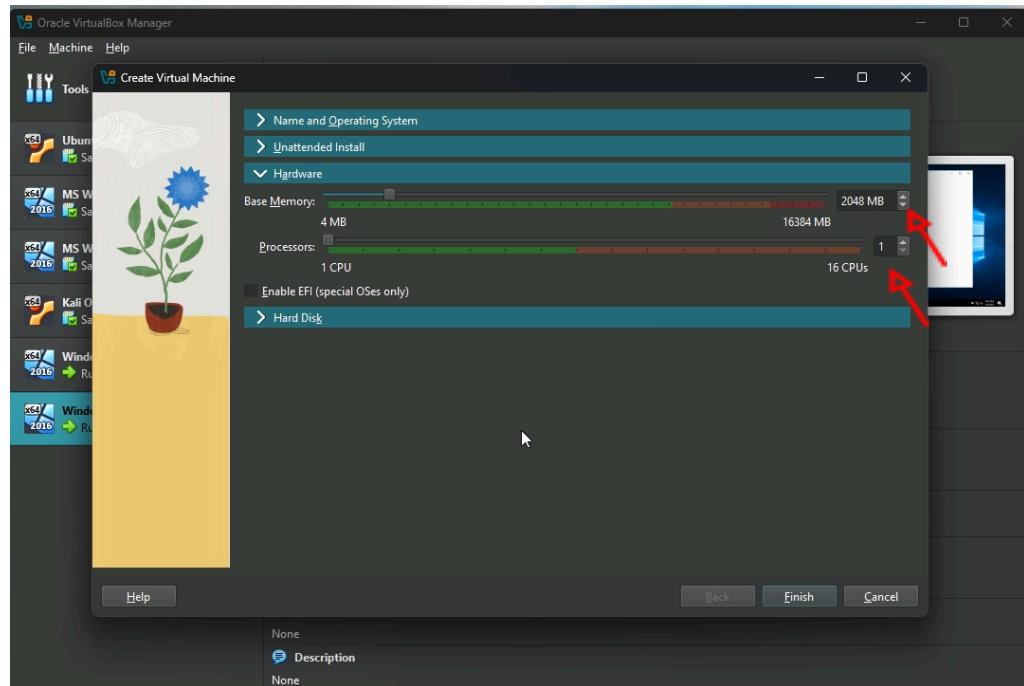
Step 2: In the VirtualBox Manager window, click the "New" button to start the process of creating a new virtual machine.



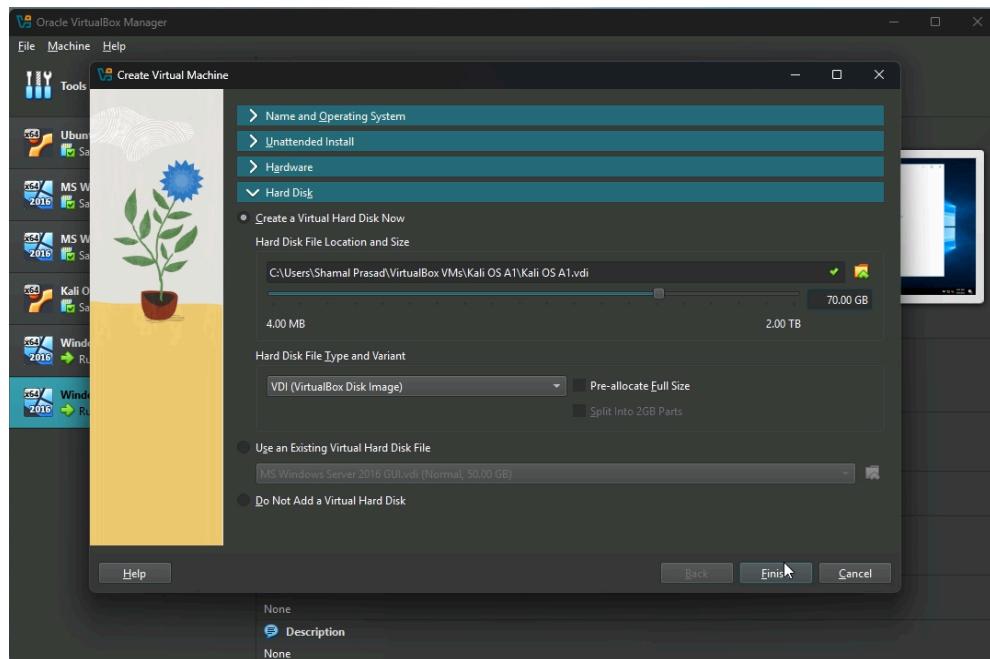
Step 3: In the Create Virtual Box Machine, enter the name of the guest OS. Click the drop down menu for the ISO image and select your desired ISO image that contains the OS that you want to install. Ensure that the Skip Unattended Installation is checked, if it isn't greyed out.



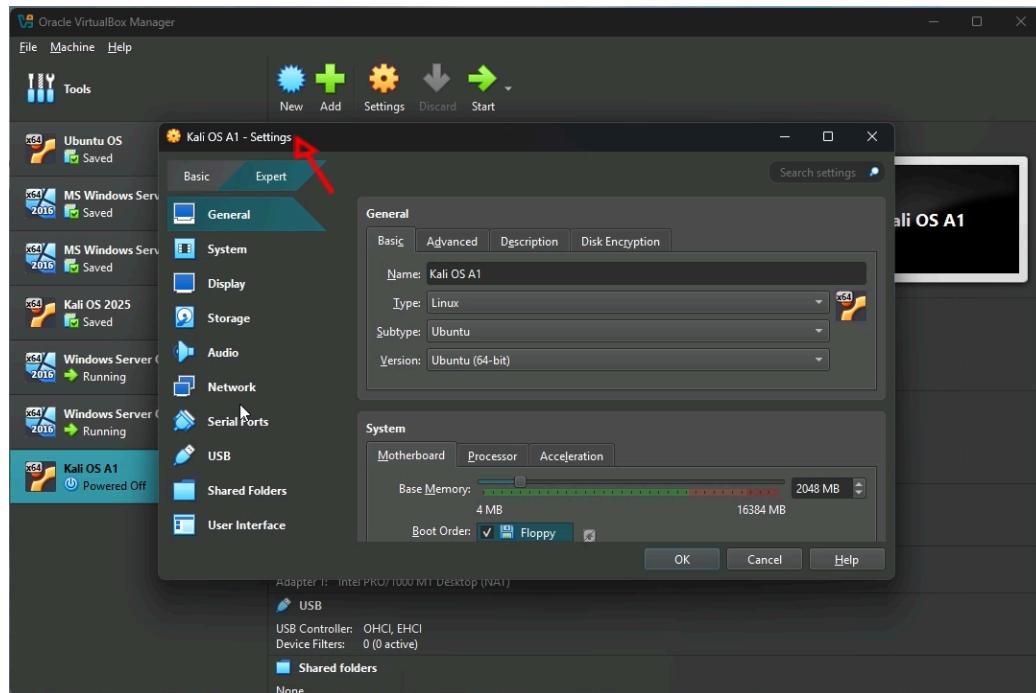
Step 4: Configure the hardware by choosing the right amount of memory and CPU cores to ensure the system runs smoothly. Minimum amount of required RAM should be 2GB, but you can allocate more than that as you see fit, but ensure to not over allocate as it will hinder the performance of your main computer. The core can be left at 1 as it is enough.



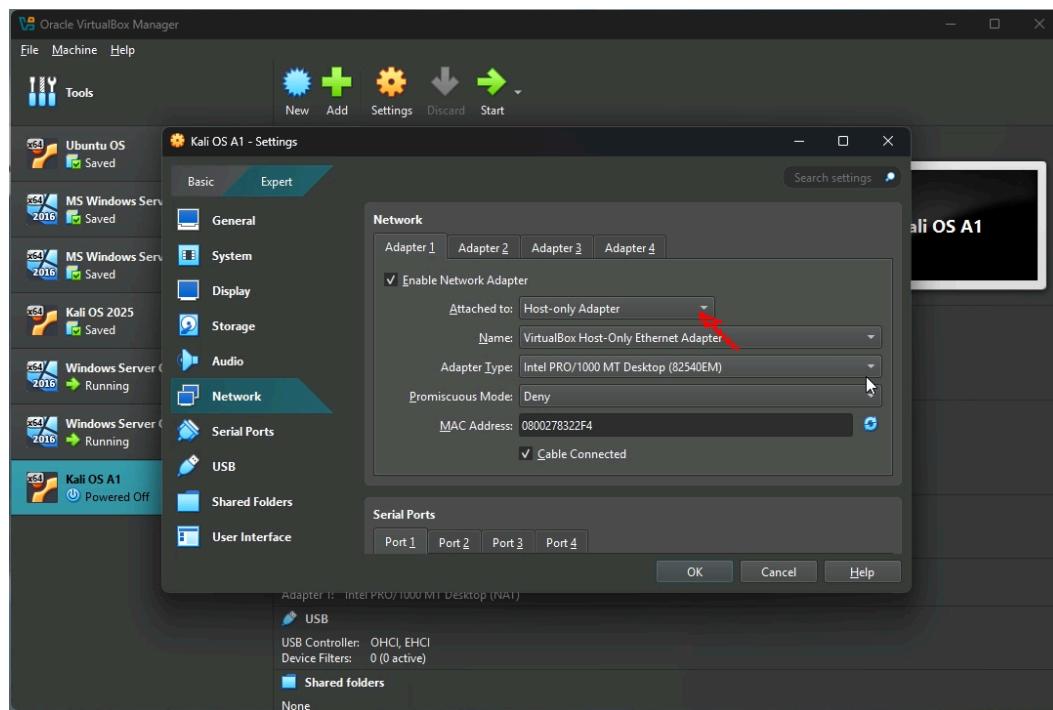
Step 5: Configure the size of the hard drive that the OS can use changing the slider or typing the value out. Around 70 Gb of space should be enough for the Kali OS. Once allocated, click Finish.



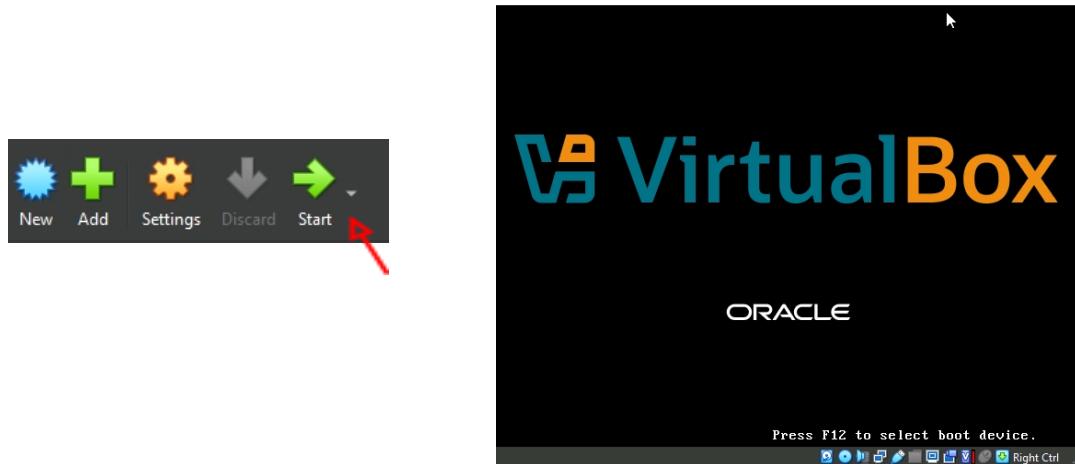
Step 6: Before running the Kali OS, some settings need to be changed. Open the Settings in the VirtualBox Manager window and locate the network setting and ensure you are on Expert Mode.



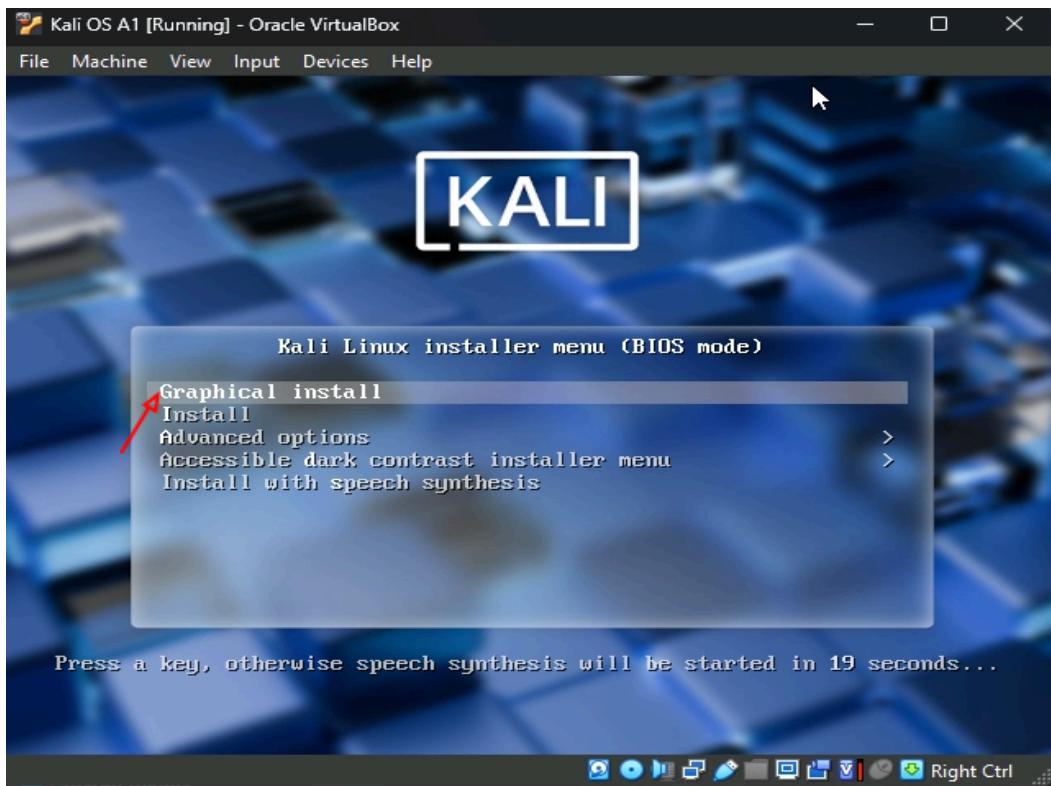
Once on the Network tab, change the network adapter to Host-only Adapter. Click OK.



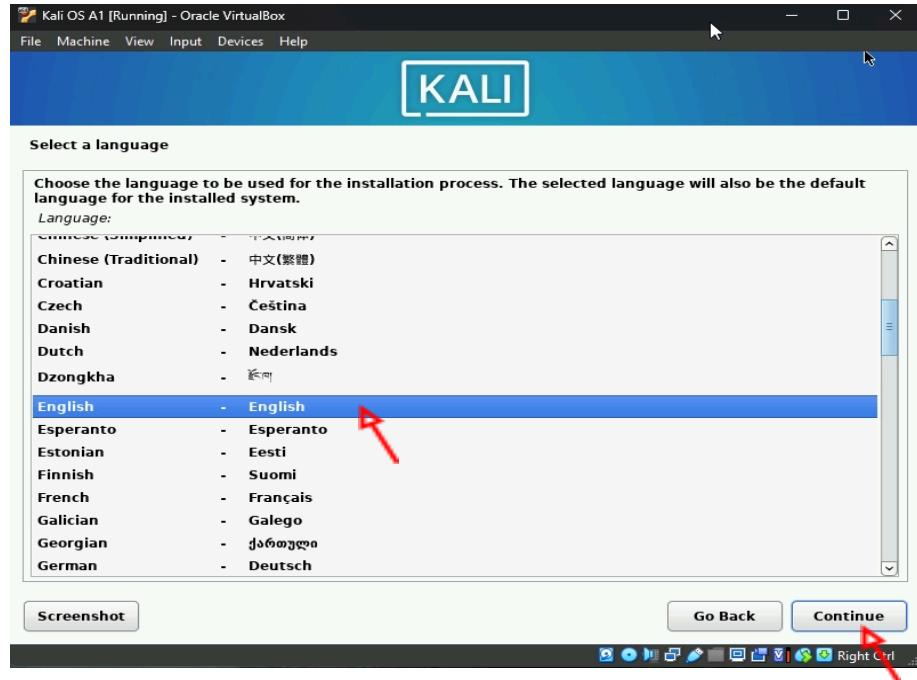
Step 7: Once you have configured the setting, you can now go ahead and start the OS by clicking the start Arrow in the VirtualBox Manager window at the top.



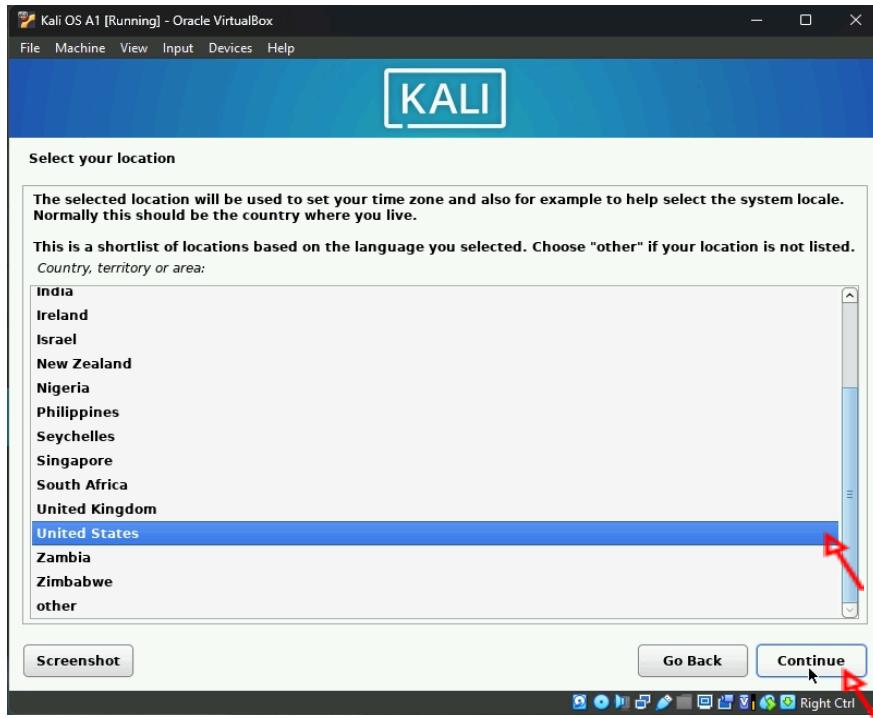
Step 8: Select Graphical Install and press enter on your keyboard to start the installation and setup process of the OS.



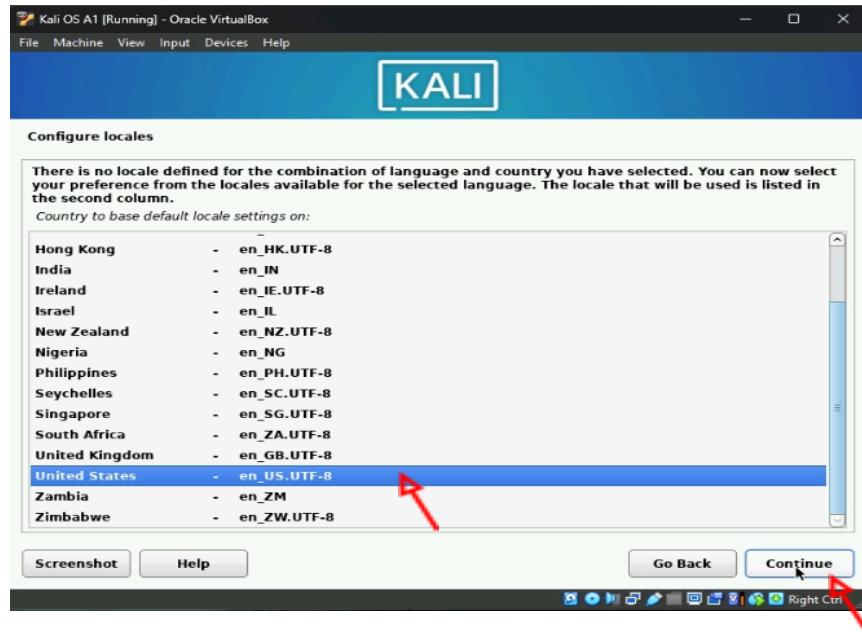
Step 9: Select the Language and press Continue.



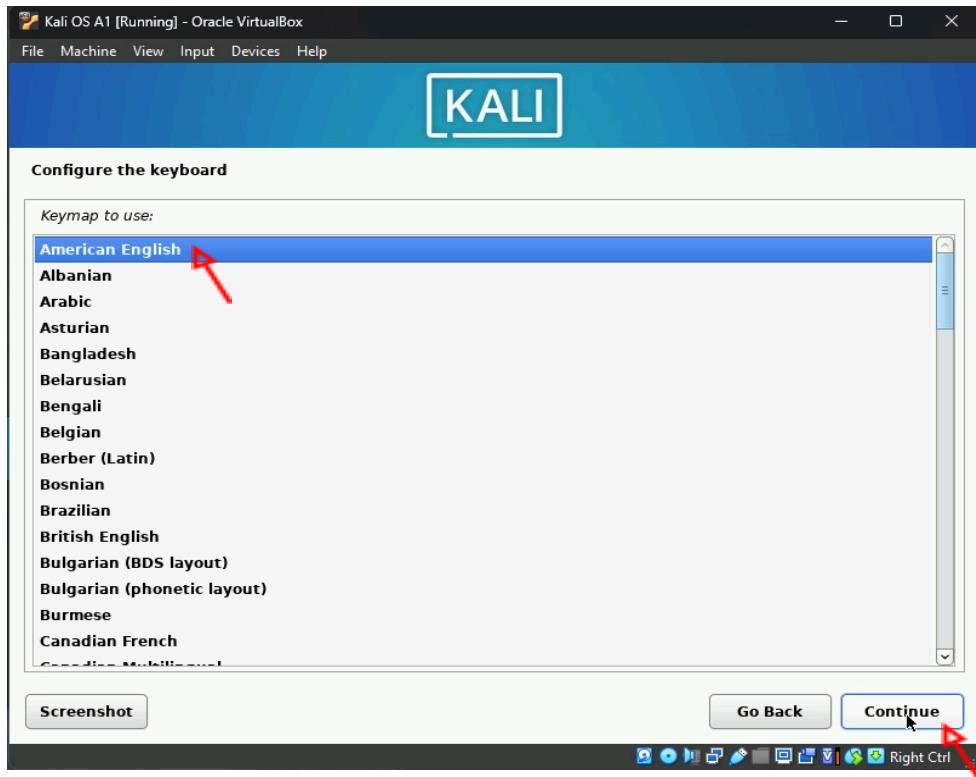
Step 10: Select your Location and press Continue.



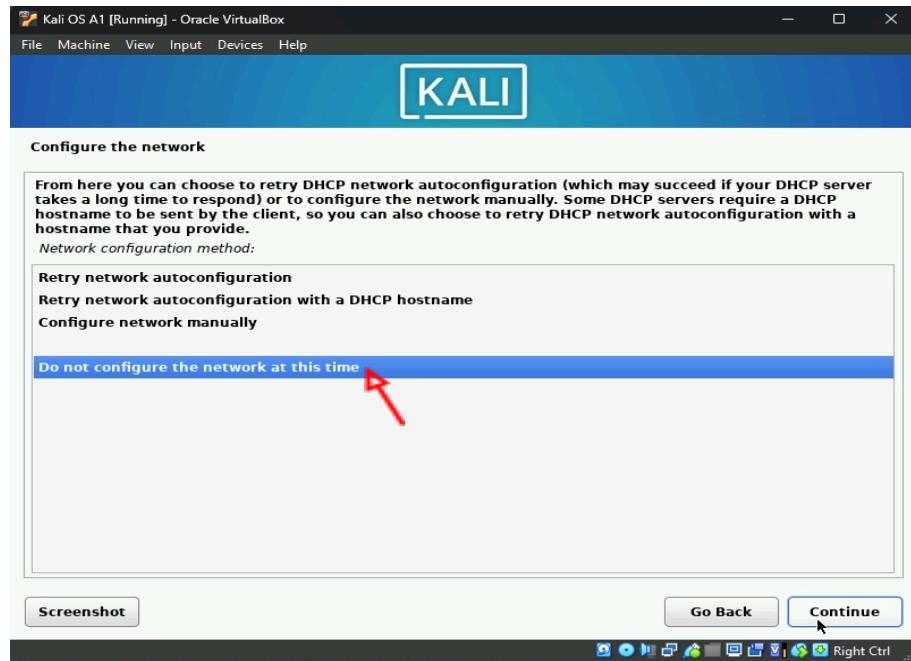
Step 11: Select your locale. Just choose the United States, it's not that important for our use case.



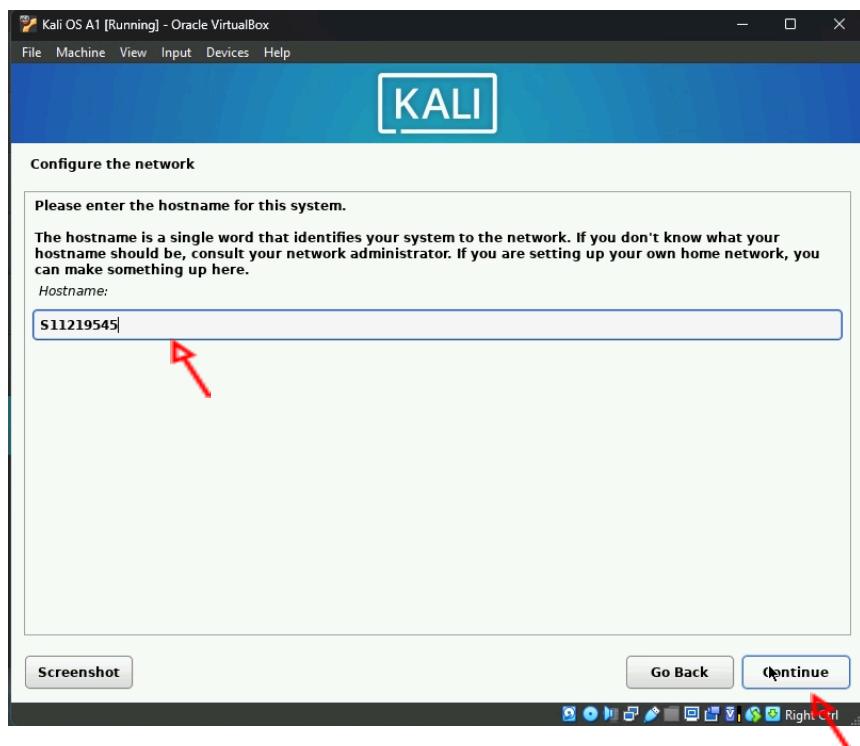
Step 12: Select your keyboard configuration, which would be American English.



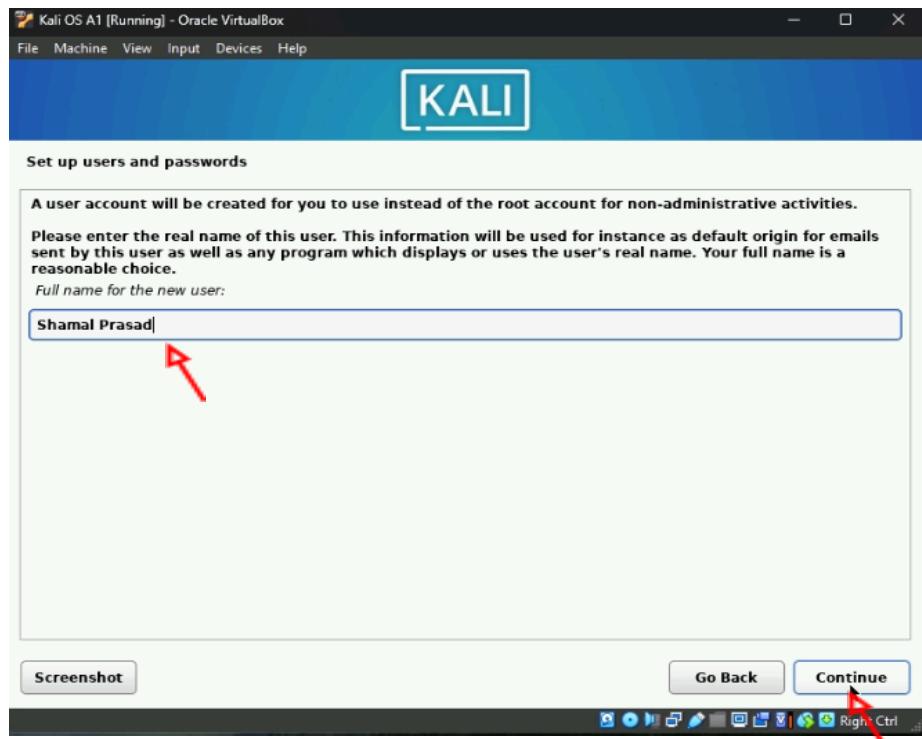
Step 13: The installation process will ask you to set up your network, but select “Do not configure network at this time” and press Continue.



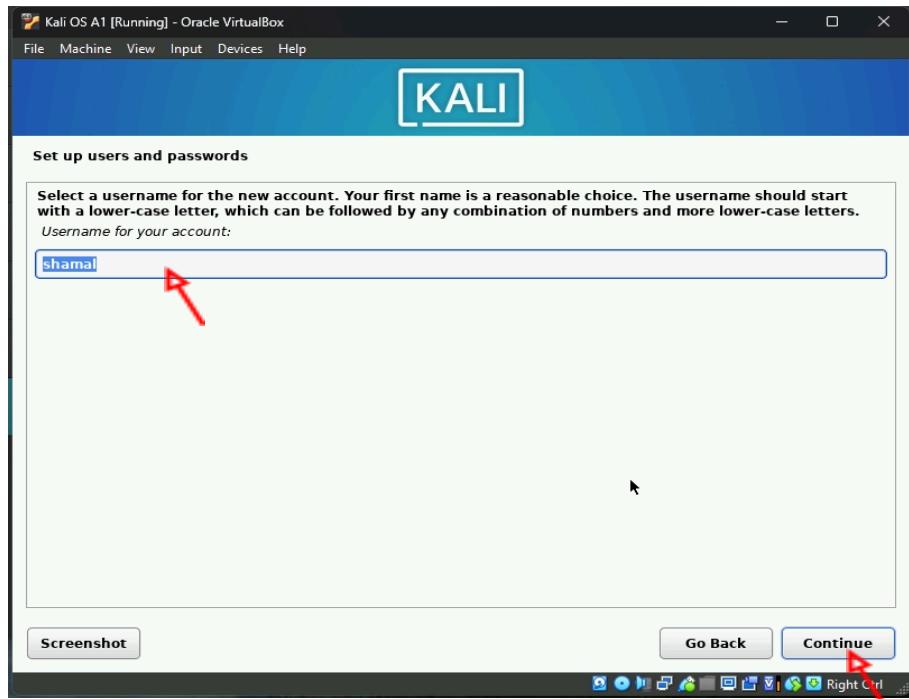
Step 14: you will now be prompted to enter the hostname of the OS. This can be your name or ID. In our case it will be a student number. Then press continue.



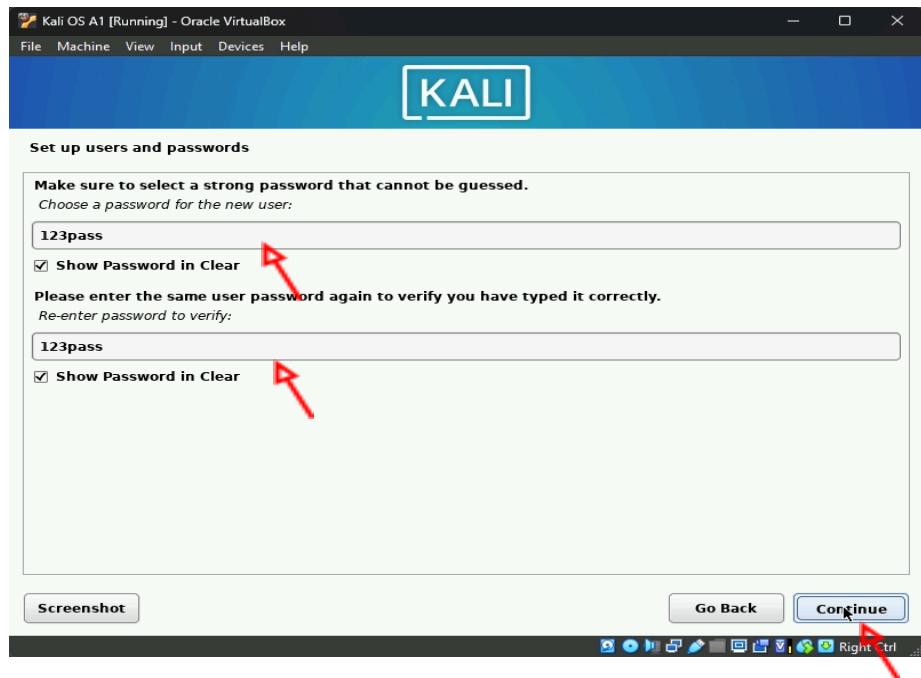
Step 15: Enter your full name and press Continue.



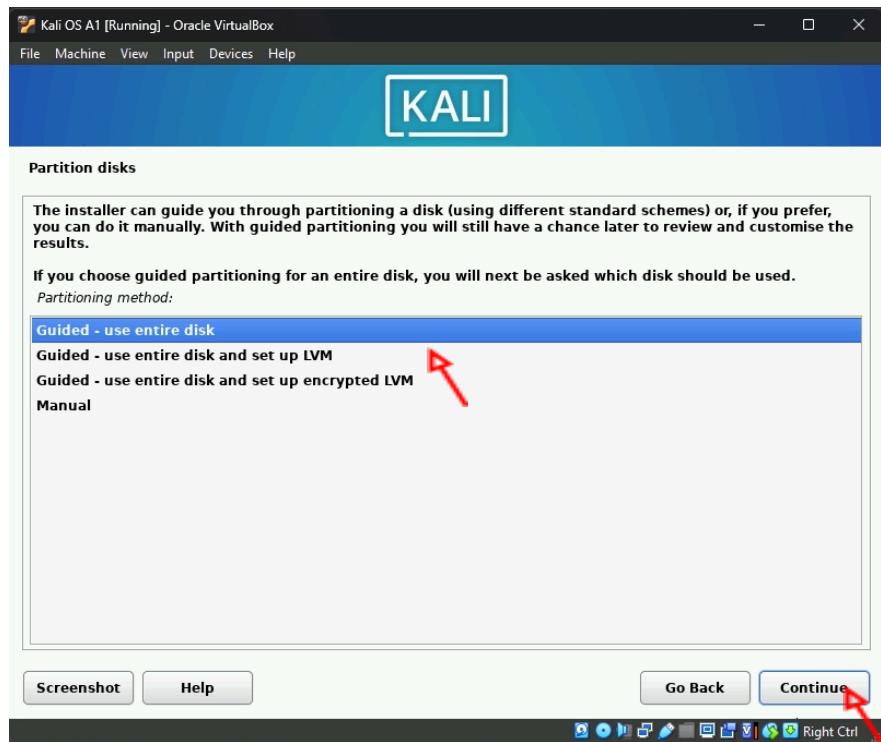
Step 16: You will now be required to enter a username for your account, ensure your username are all lower-case letters. Then press continue.



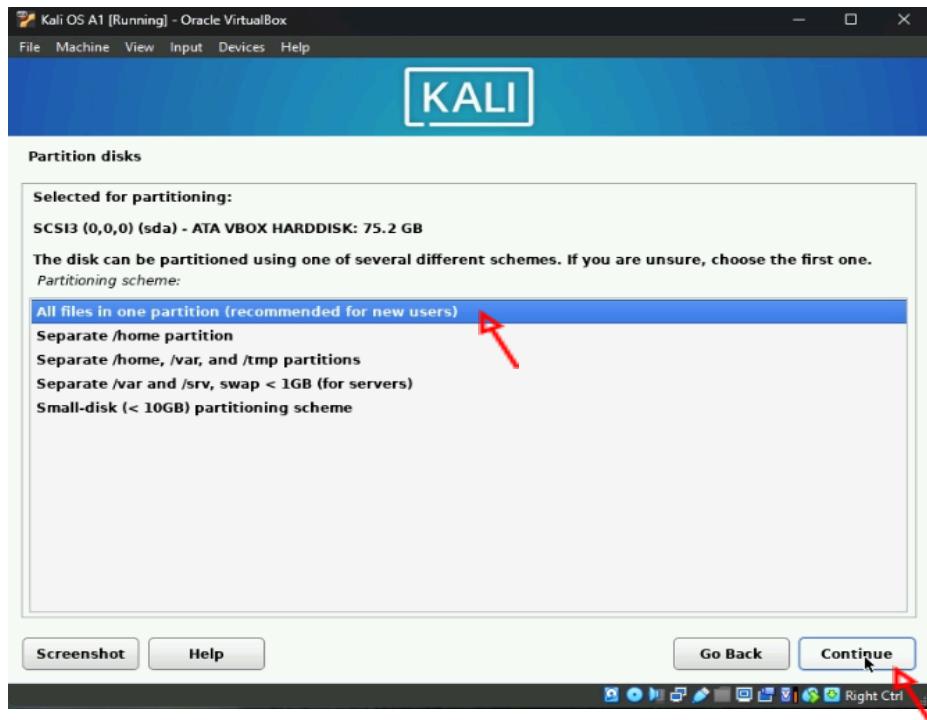
Step 17: After choosing a suitable username for your account, you will now need to set up a password for the same account.



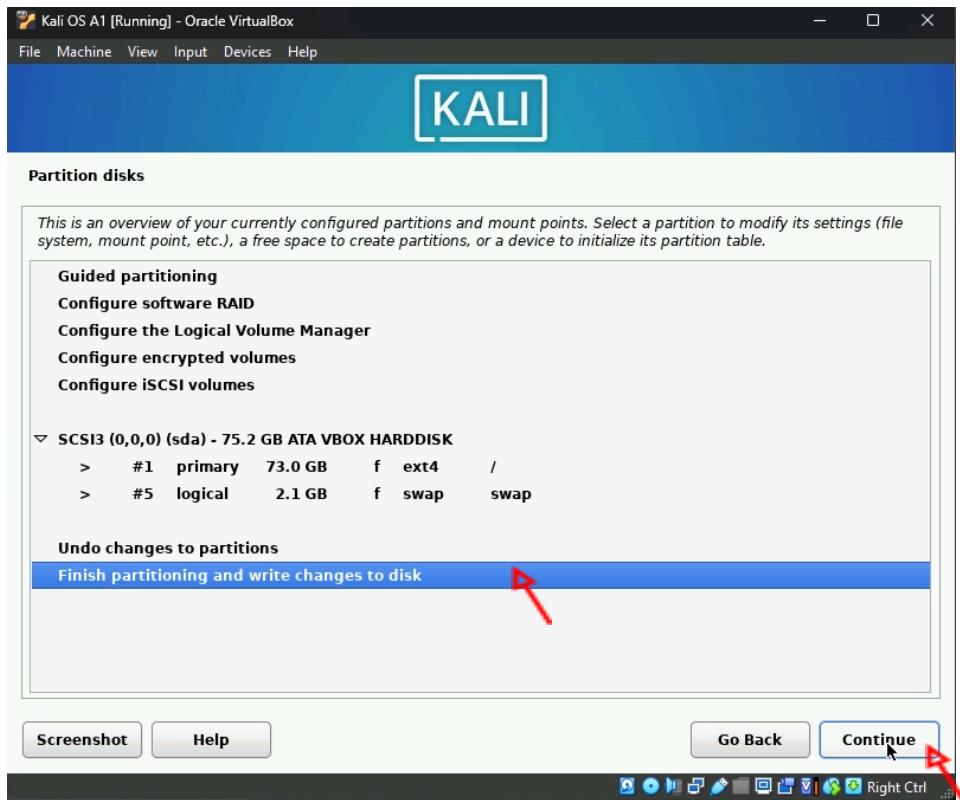
Step 18: The following screen requires you to determine the partitioning method of your harddrive. Select “Guided- use the entire disk” and then press Continue.



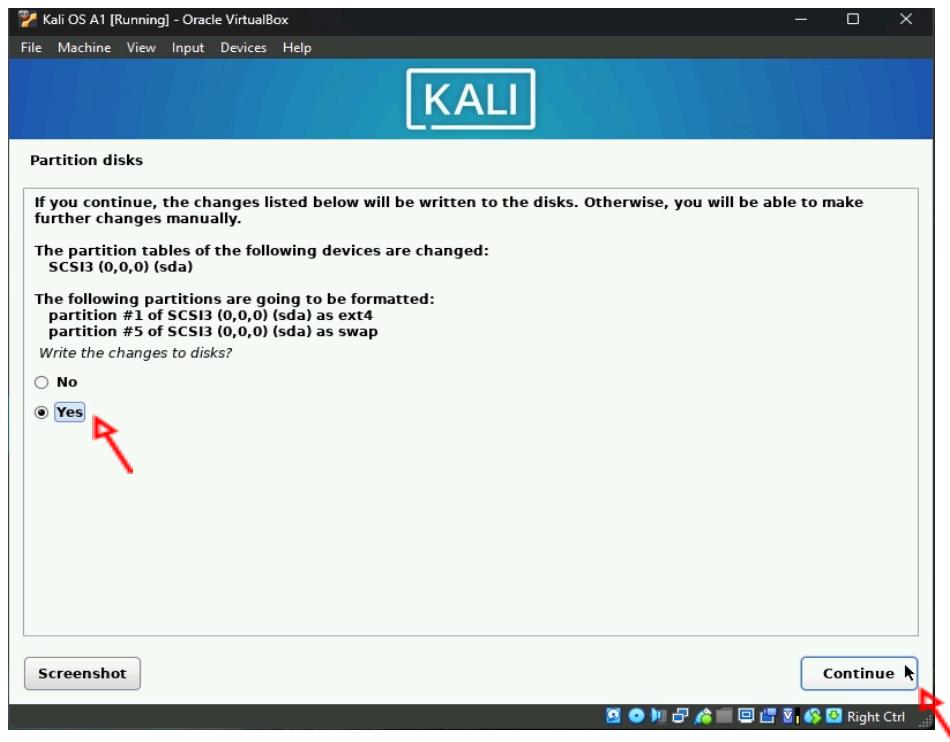
Step 19: In the following screen, choose the partitioning scheme as “All file in one partition” and press Continue.



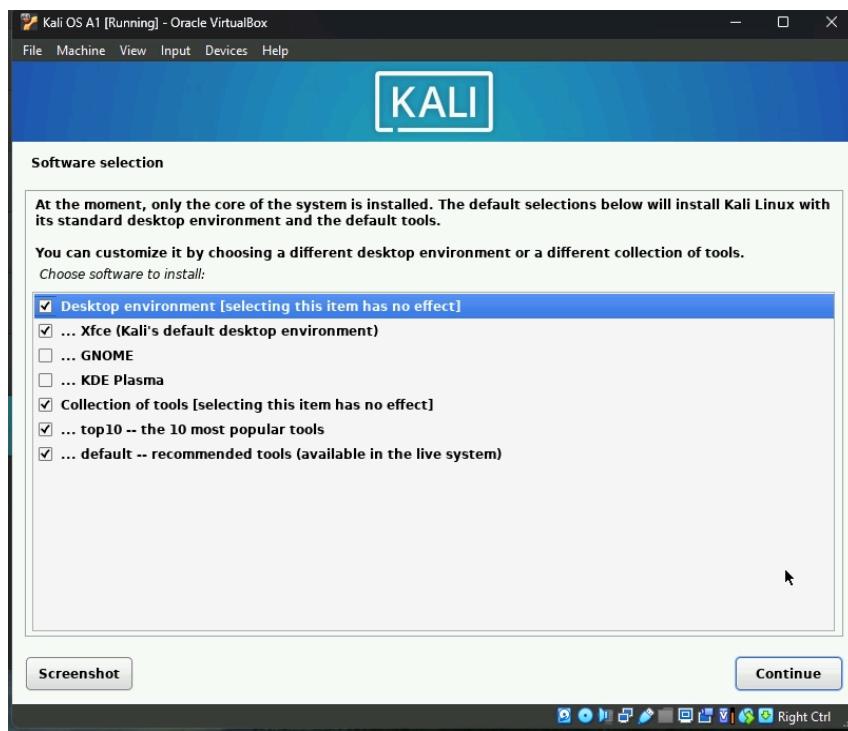
Step 20: The following screen just shows you the options you have previously chosen for partitioning your disk. Press Continue.



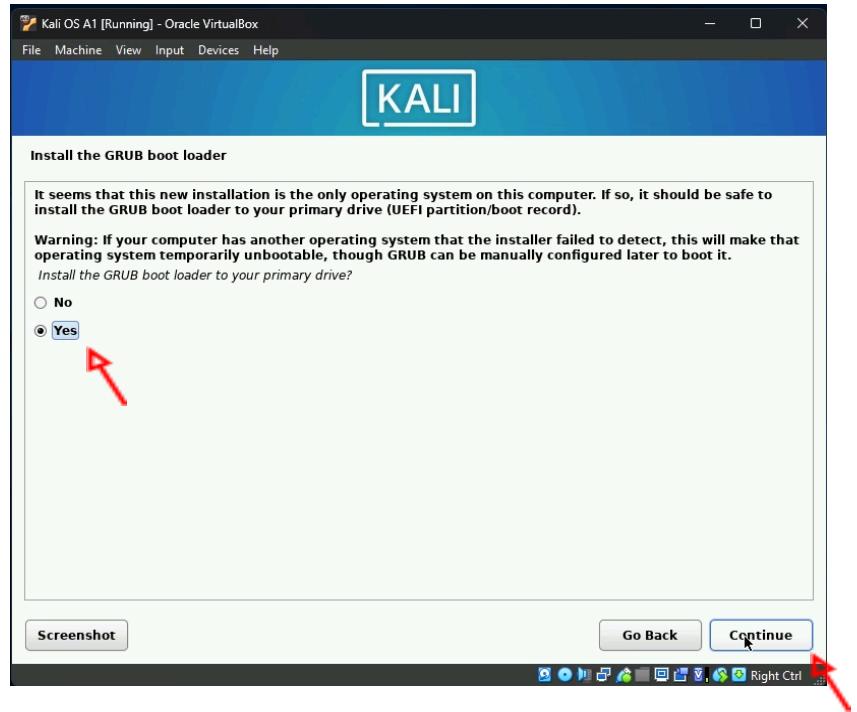
Step 21: The following screen confirms the hard drive partitioning options. Select yes and Press Continue.



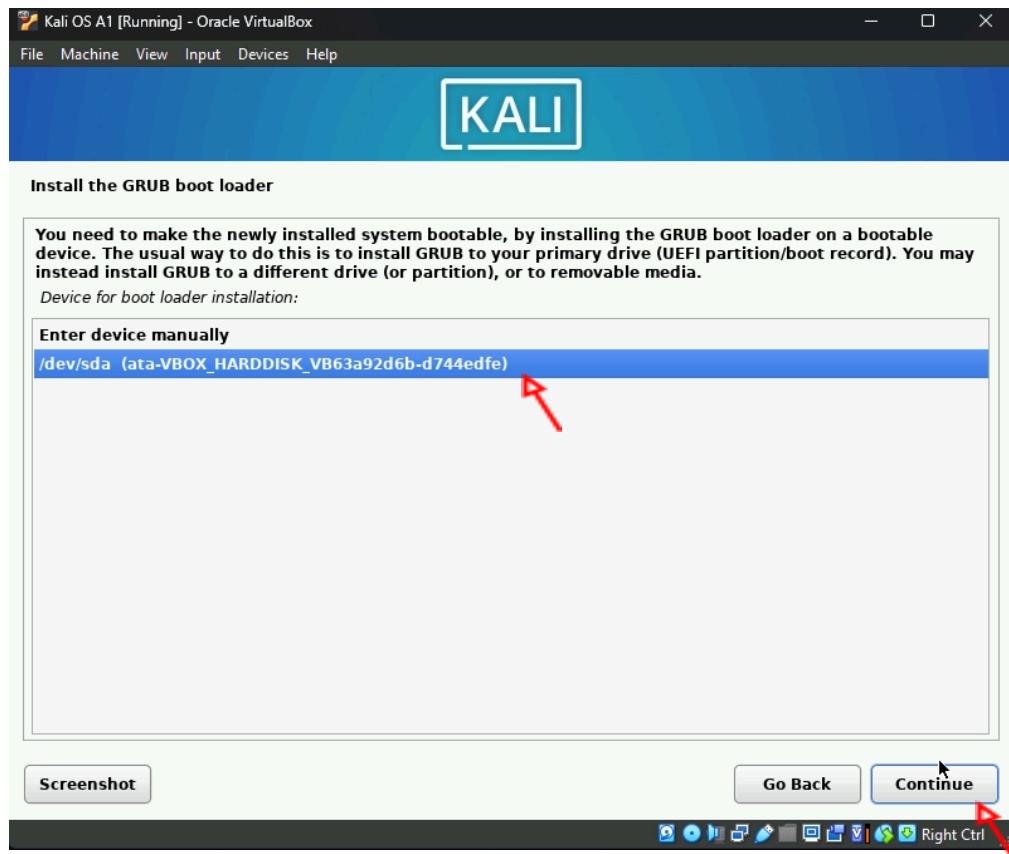
Step 22: The following screen gives option what additional software you want to install with the OS, by default some will be selected and some won't be. Do not worry, just press Continue.



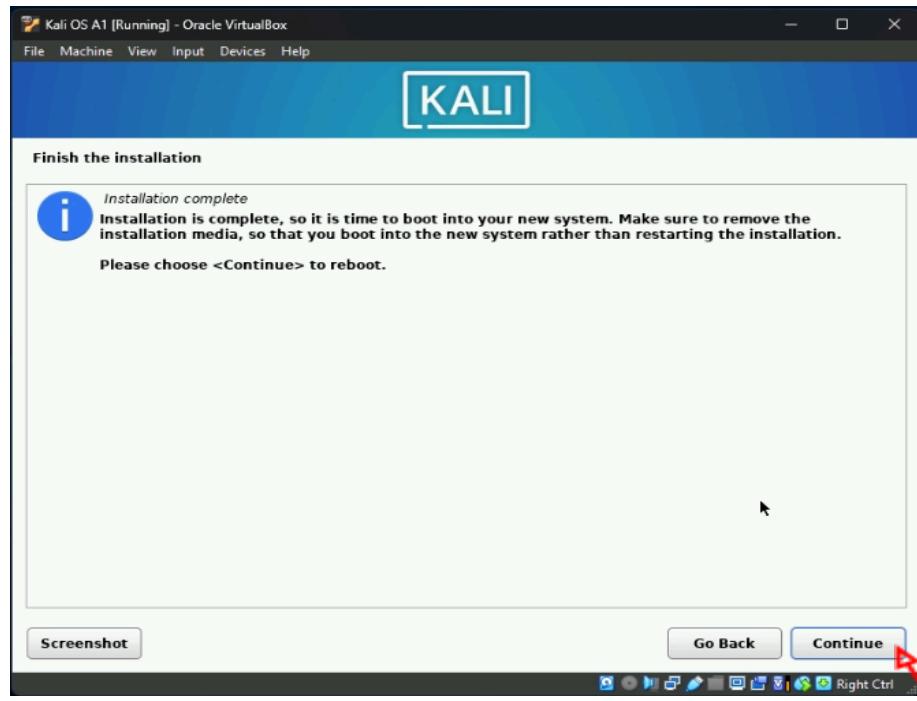
Step 23: This screen gives you a choice to install the bootloader of the OS to your primary harddrive. Select Yes and press Continue.



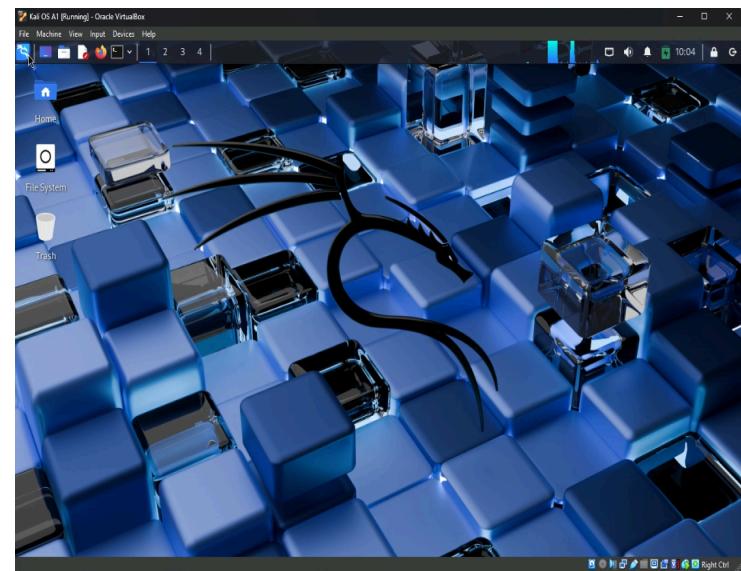
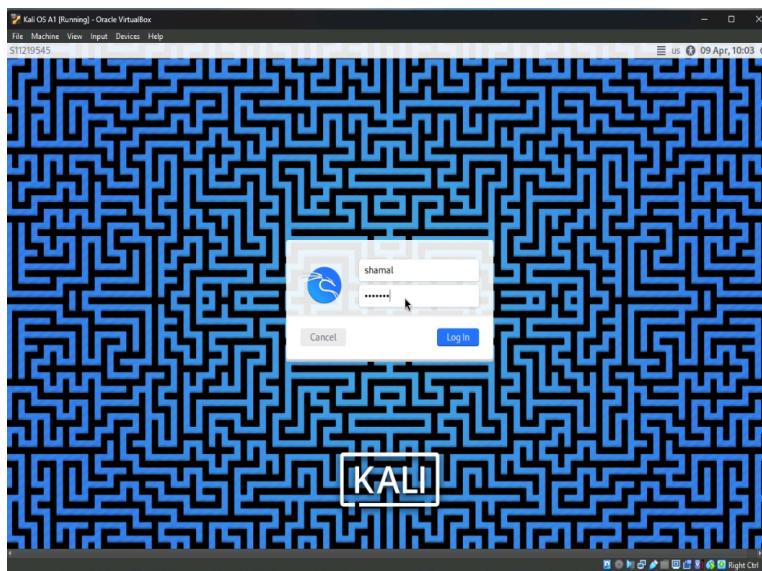
Step 24: This screen shows you a list of drives that you have and you can choose which one you want to install the bootloader into. Select the desired drive and press Continue.



Step 25: This screen informs you that inorder to finish the installation process the OS will need to be rebooted. You can go ahead and press Continue.

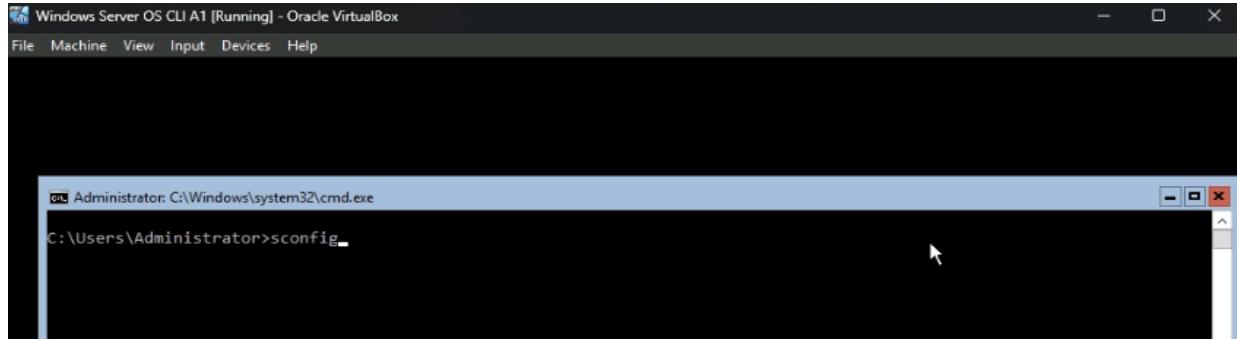


Step 26: You can now login into your OS system. Simply enter your username that you chose the password you created earlier to login.

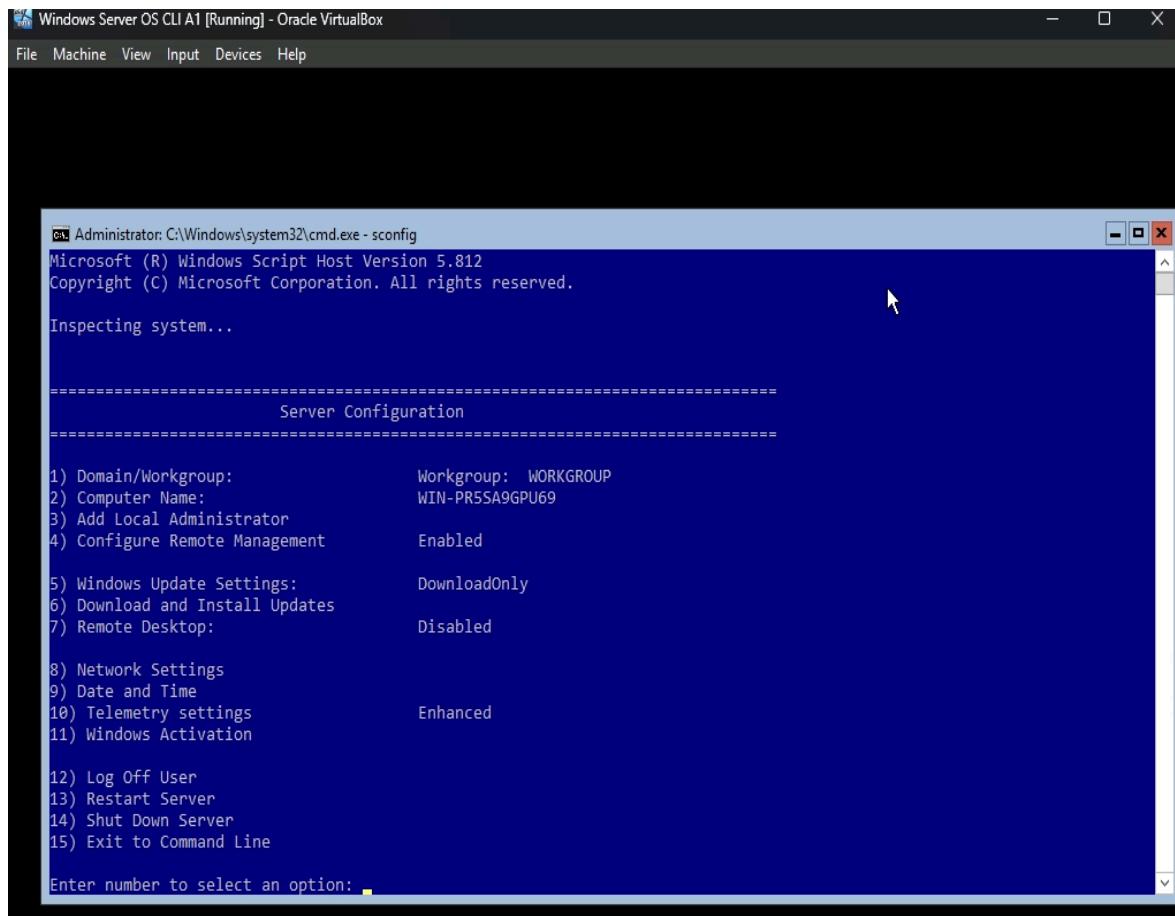


Setting Up Windows Server OS CLI IP Address

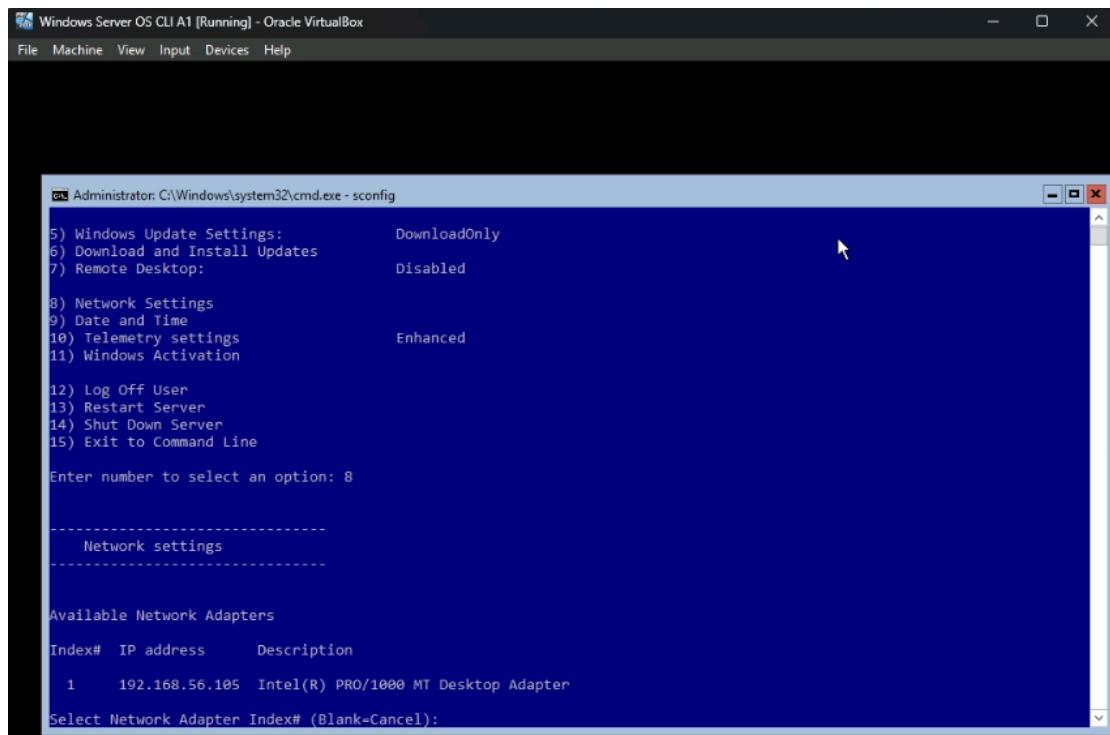
Step 1: In the cmd, which is your command prompt, enter the following command, “sconfig”, which opens up system configuration in the command prompt.



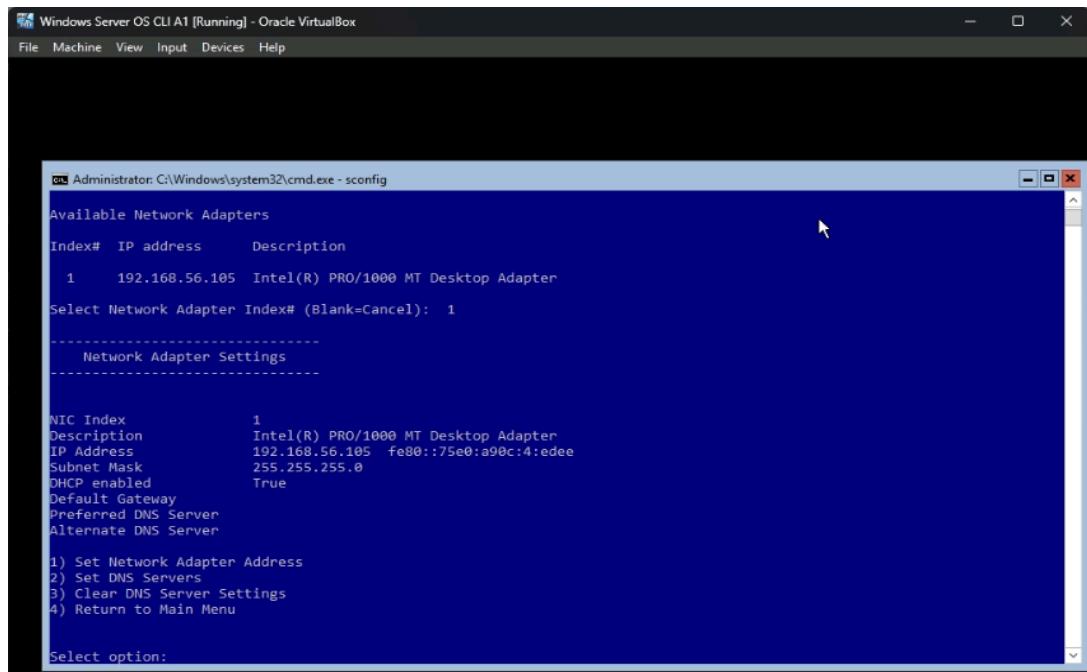
After which you will be presented with the screen shown below.



Step 2: Type option number 8 and press enter on your keyboard to start configuring your network setting.



Step 3: Enter your network adapter index which in this case is 1 and press enter on your keyboard.



Step 4: Select option 1 and start setting up your network adapter address.

```
Windows Server OS CLI A1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help

Administrator: C:\Windows\system32\cmd.exe - sconfig
Index# IP address Description
1 192.168.56.105 Intel(R) PRO/1000 MT Desktop Adapter
Select Network Adapter Index# (Blank=Cancel): 1

Network Adapter Settings

NIC Index      1
Description    Intel(R) PRO/1000 MT Desktop Adapter
IP Address     192.168.56.105 fe80::75e0:a90c:4:edee
Subnet Mask    255.255.255.0
DHCP enabled   True
Default Gateway
Preferred DNS Server
Alternate DNS Server

1) Set Network Adapter Address
2) Set DNS Servers
3) Clear DNS Server Settings
4) Return to Main Menu

Select option: 1

Select (D)HCP, (S)static IP (Blank=Cancel):
```

Step 5: Enter letter 's' and press enter to configure the static IP address of the OS.

```
Windows Server OS CLI A1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help

Administrator: C:\Windows\system32\cmd.exe - sconfig
Select Network Adapter Index# (Blank=Cancel): 1

Network Adapter Settings

NIC Index      1
Description    Intel(R) PRO/1000 MT Desktop Adapter
IP Address     192.168.56.105 fe80::75e0:a90c:4:edee
Subnet Mask    255.255.255.0
DHCP enabled   True
Default Gateway
Preferred DNS Server
Alternate DNS Server

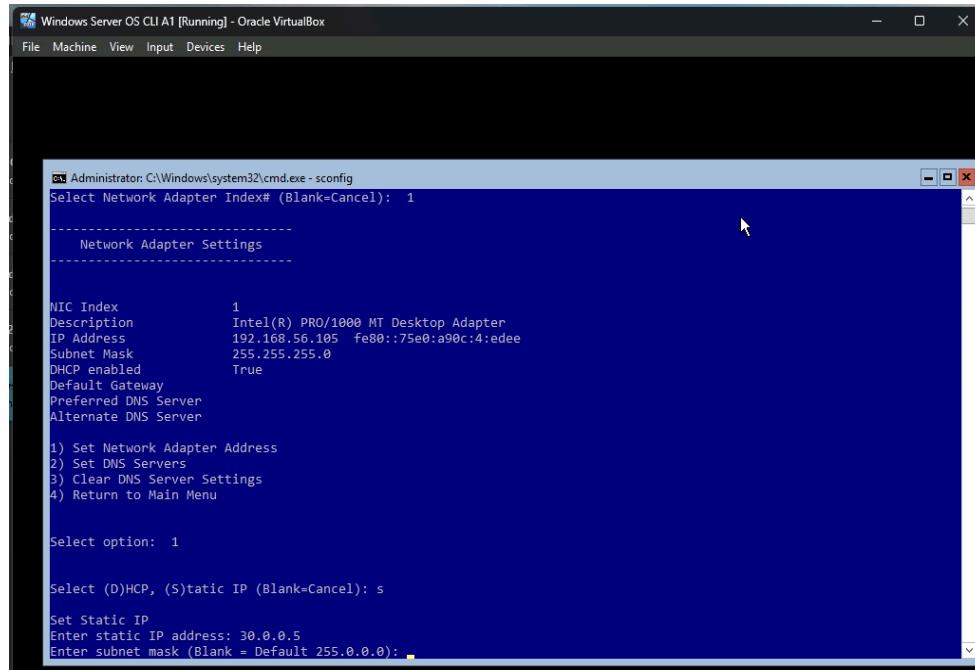
1) Set Network Adapter Address
2) Set DNS Servers
3) Clear DNS Server Settings
4) Return to Main Menu

Select option: 1

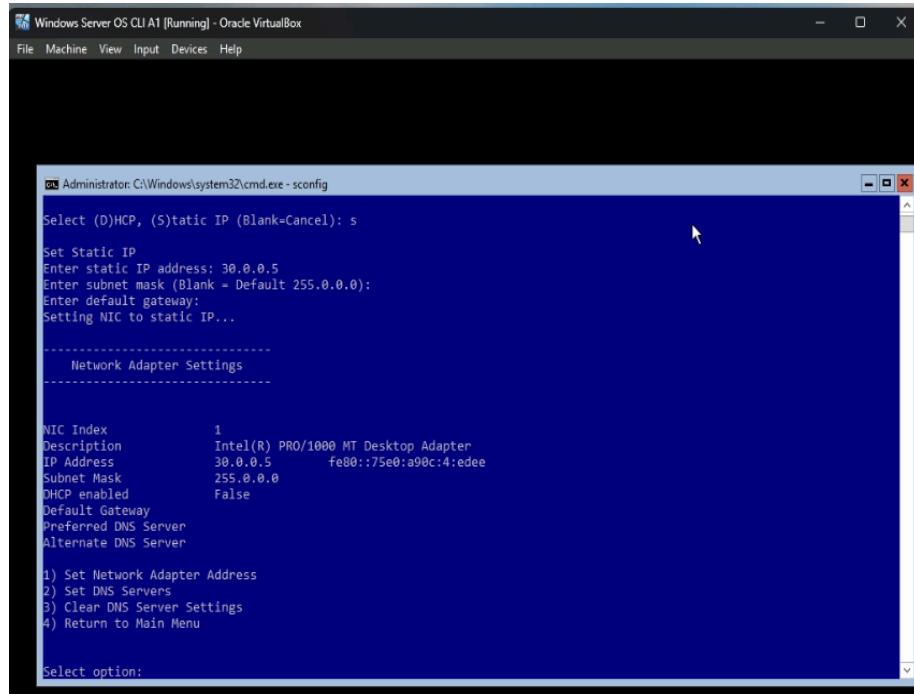
Select (D)HCP, (S)static IP (Blank=Cancel): s

Set Static IP
Enter static IP address:
```

Step 6: Enter your IP address which in our case will be “30.0.0.5” and press enter to save the new IP address.



Step 7: Now you must enter your subnet mask, the default value is “255.0.0.0”. You can either type that or just press enter and it will automatically save the default subnet mark as “255.0.0.0”.



Step 8: After setting up the subnet mask, you need to set your default gateway, but for us we don't have to set it, so just press enter.

```

Windows Server OS CLI A1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help

Administrator: C:\Windows\system32\cmd.exe - sconfig

Network Adapter Settings

NIC Index      1
Description    Intel(R) PRO/1000 MT Desktop Adapter
IP Address     192.168.56.105  fe80::75e0:a90c:4:edee
Subnet Mask   255.255.255.0
DHCP enabled   True
Default Gateway
Preferred DNS Server
Alternate DNS Server

1) Set Network Adapter Address
2) Set DNS Servers
3) Clear DNS Server Settings
4) Return to Main Menu

Select option: 1

Select (D)HCP, (S)tatic IP (Blank=Cancel): s

Set Static IP
Enter static IP address: 30.0.0.5
Enter subnet mask (Blank = Default 255.0.0.0):
Enter default gateway:

```

Step 9: You need to establish connection with remote computers or OS's. which means you need to let other devices be able to ping your device. To make this possible, while we are in the sconfig screen, enter option '4' to configure remote management.

```

Windows Server OS CLI A1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help

Administrator: C:\Windows\system32\cmd.exe - sconfig

7) Remote Desktop:           Disabled
8) Network Settings
9) Date and Time
10) Telemetry settings:     Enhanced
11) Windows Activation

12) Log Off User
13) Restart Server
14) Shut Down Server
15) Exit to Command Line

Enter number to select an option: 4

Configure Remote Management

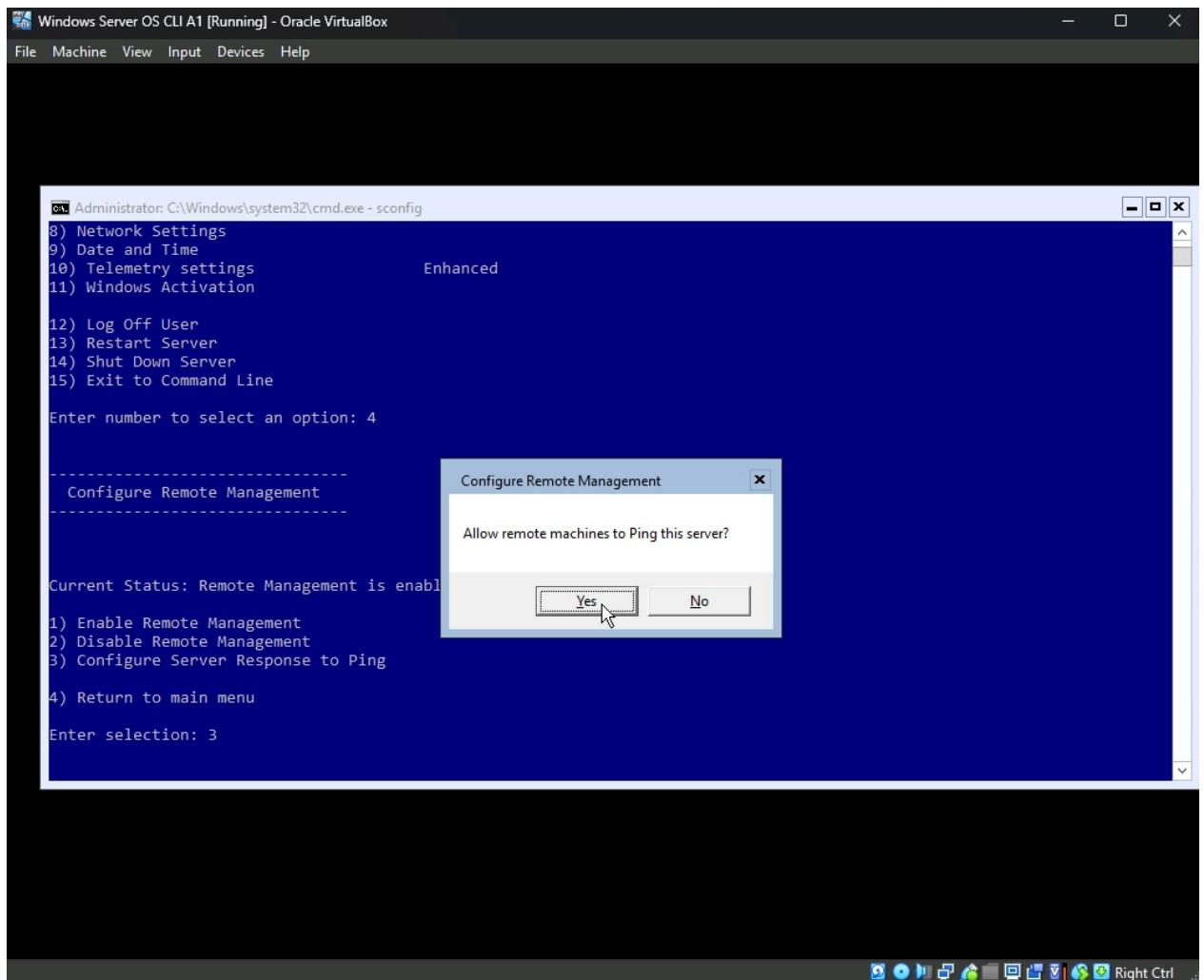
Current Status: Remote Management is enabled

1) Enable Remote Management
2) Disable Remote Management
3) Configure Server Response to Ping
4) Return to main menu

Enter selection:

```

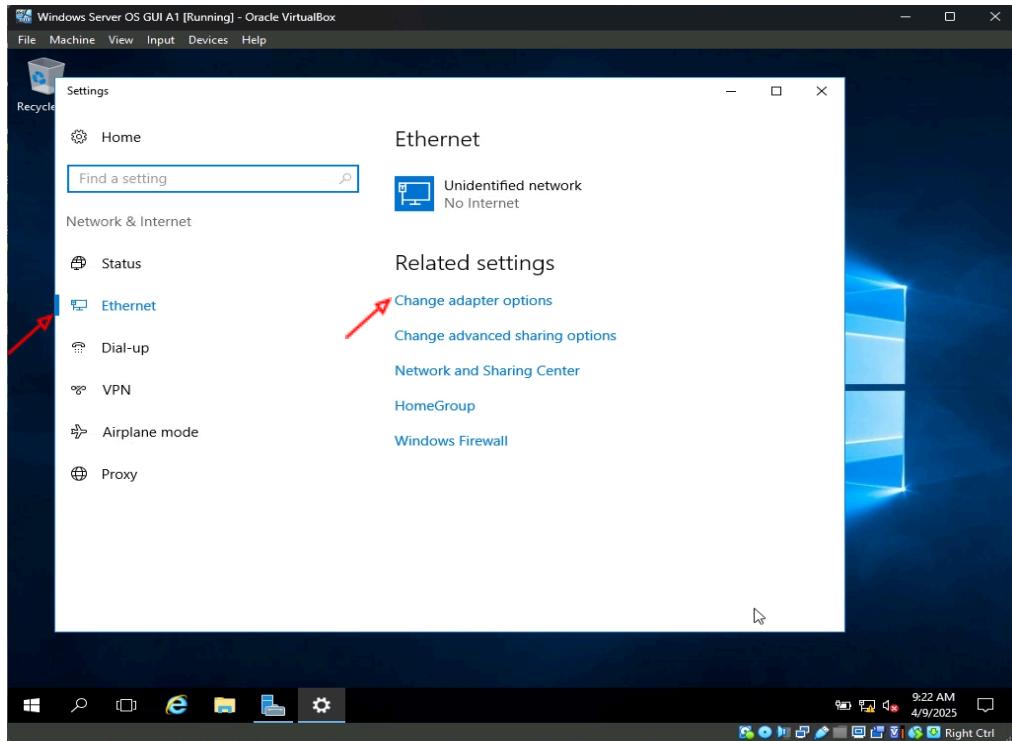
Step 10: Enter option '3' to configure the ping response to the server.



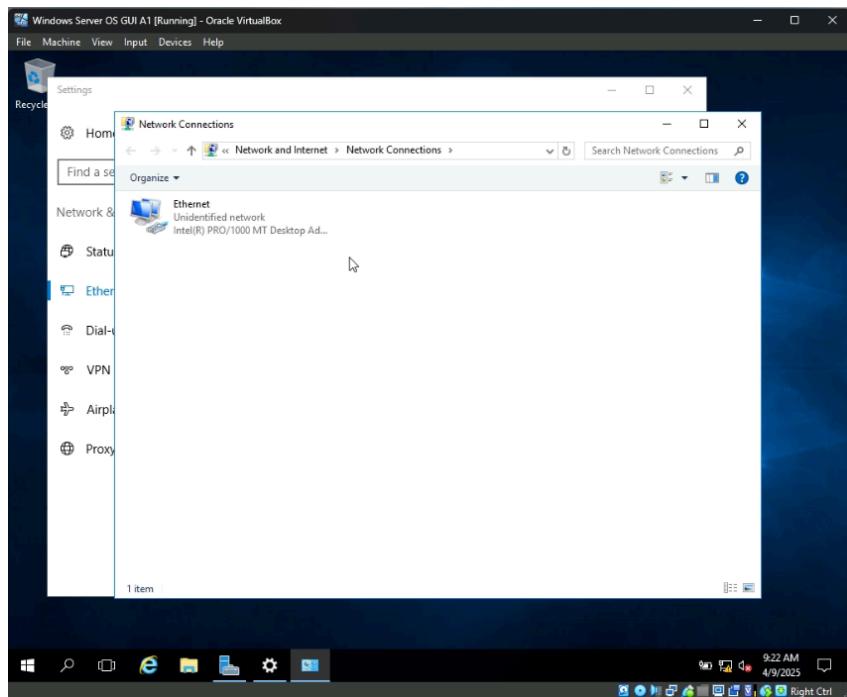
Press yes to allow other remote devices to ping to the server. After which you can enter option '15' to exit back to the command line.

Setting Up Windows Server OS GUI IP Address

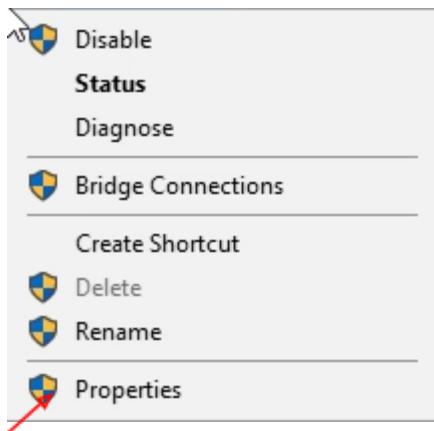
Step 1: Open your settings of the windows OS and navigate or search in the search bar Ethernet settings. You should reach a screen as shown below.



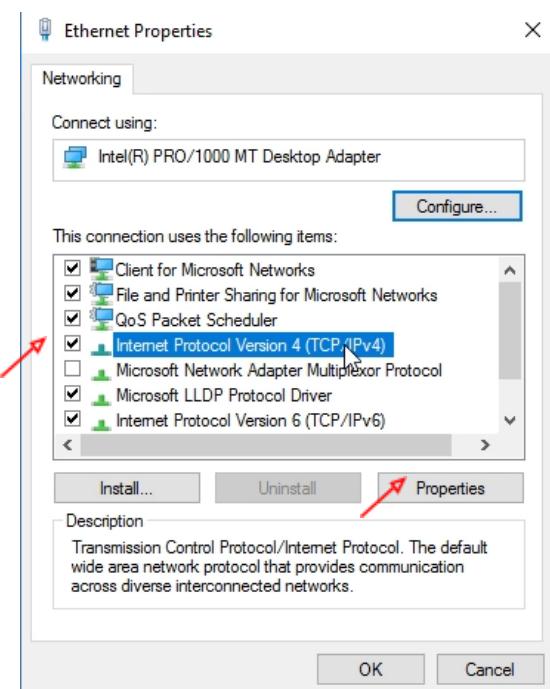
Step 2: Select the change adapter option by clicking on and you will be presented with a screen shown below.



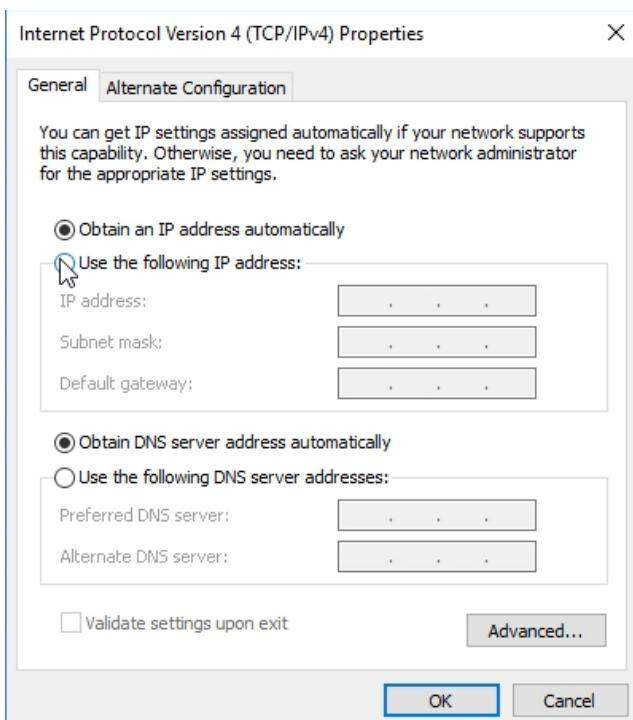
Step 3: Position your mouse over the Ethernet icon and press right click on your mouse. You should see a menu similar to the one below.



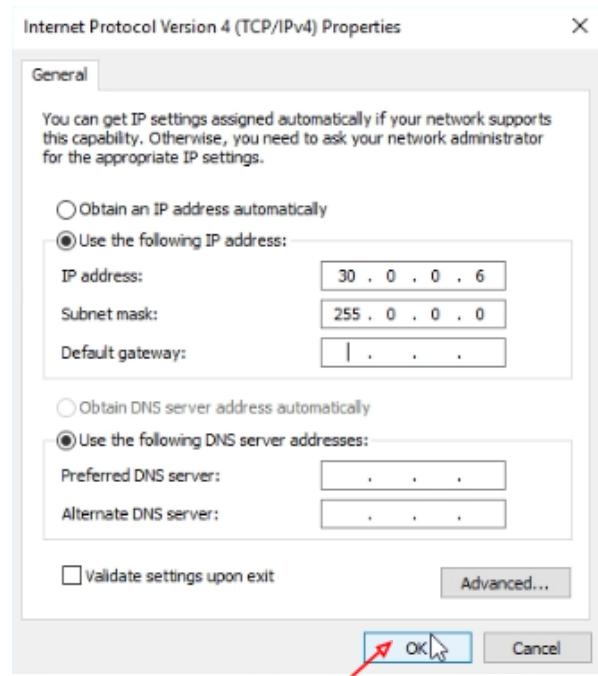
Click on Properties to view and change certain properties of the Ethernet network adapter. You should now see the window box as shown below.



Click on “internet Protocol Version 4”(IPv4) and then click on the properties button.



Step 4: On the last screen shown above, type in your IP address and subnet mask.

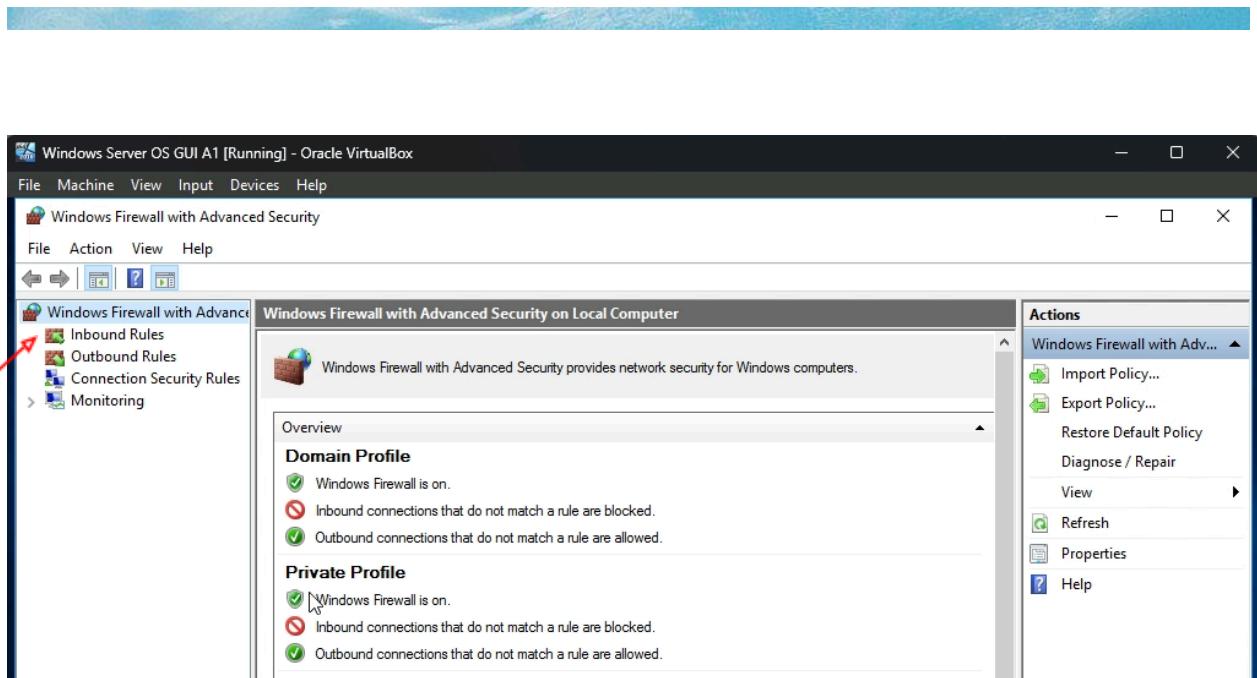


Once you have entered it, you can press OK to save the entered information.

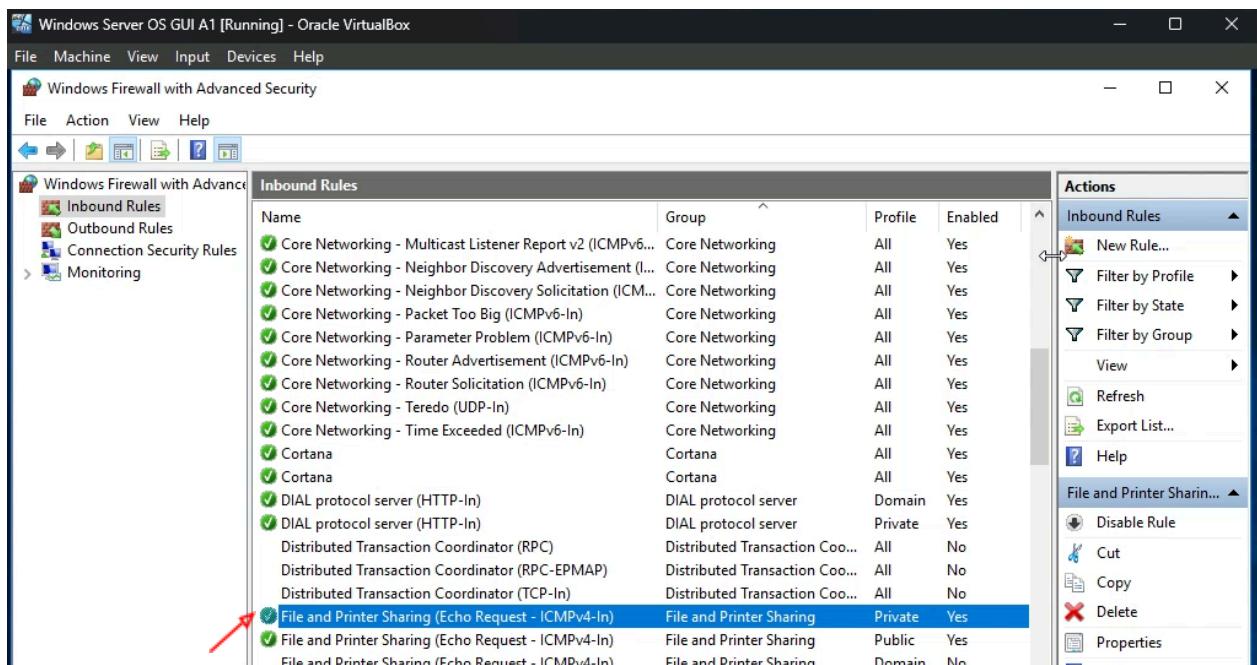
Step 5: You must enable your server OS to be able to respond to ping. One option is to do it the same way as it was done in the server OS CLI mode using the cmd. The other option is to enable this through the Windows Defender Firewall. Open the settings application and search in the search bar Windows Firewall. You should see a screen similar to the one below.



Press the Advance setting that is present on the left side of the control panel.

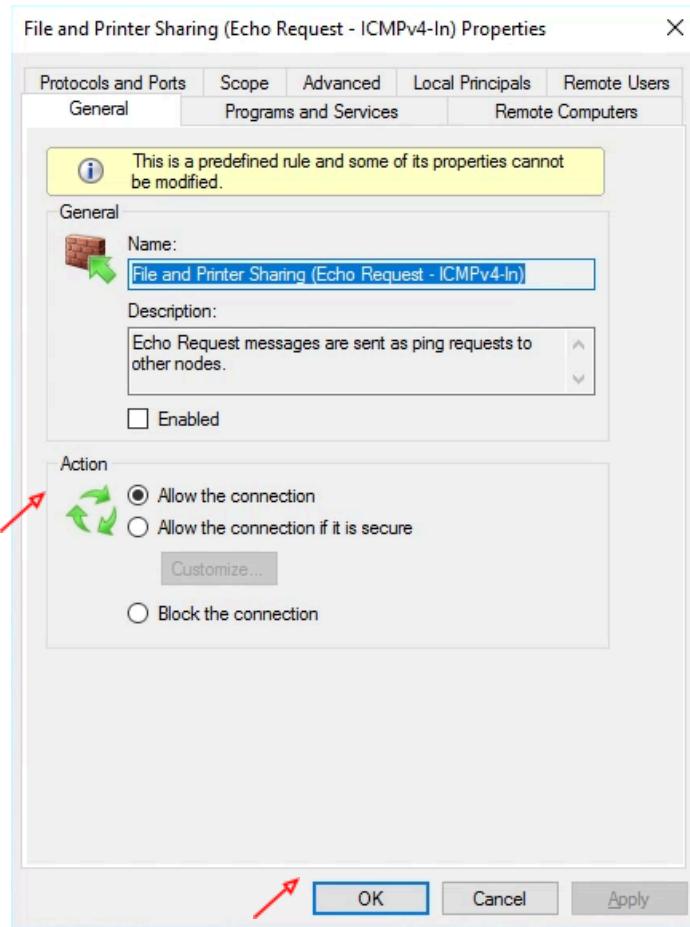


In the sidebar navigate to inbound Rules.



Look for file and printer sharing and this is where you have to enable the response to server ping. Beside each file and printer sharing for IPv4, a green tick must exist, if there isn't any then you want to enable it.

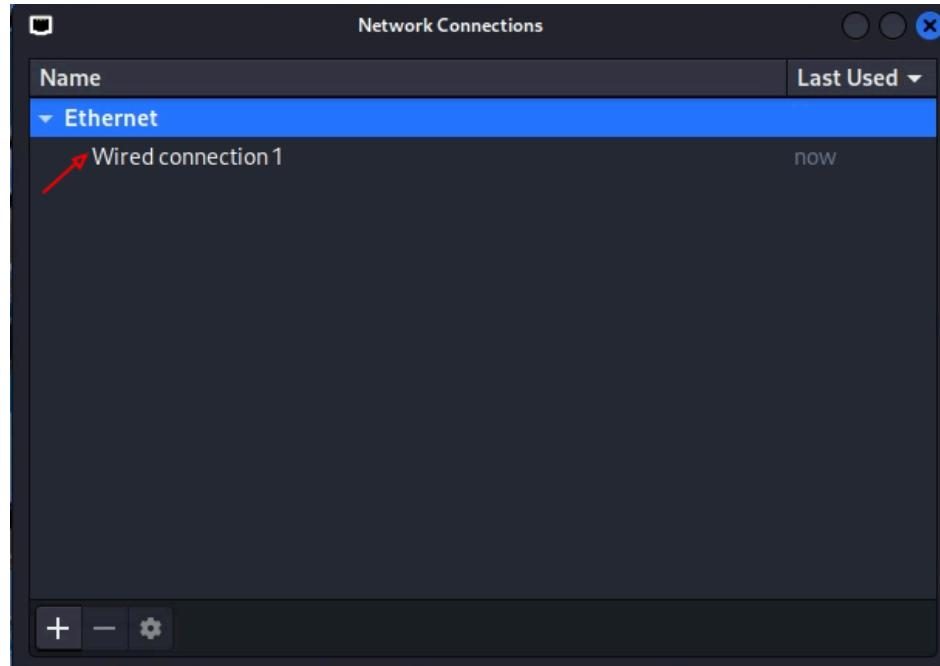
Double click on it and you should see the screen as shown below.



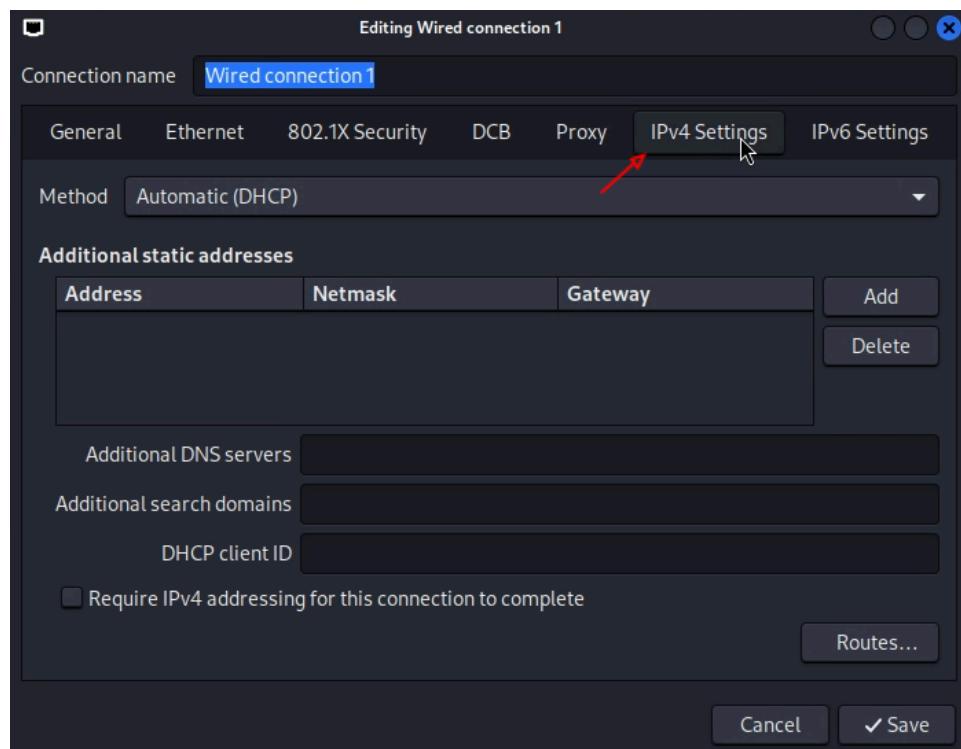
You must enable it, press apply and then press OK. Now repeat this process for all the file and printer sharing for IPv4 till it has a green tick beside it.

Setting Up Kali OS IP Address.

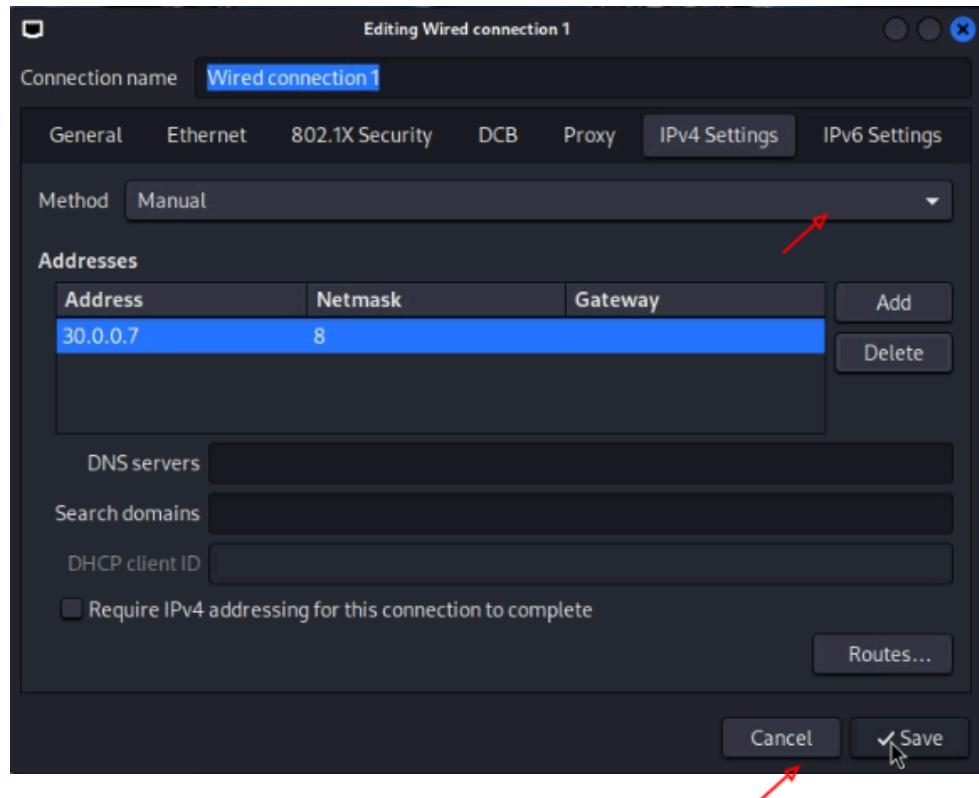
Step 1: Open your settings and look for Advanced Network Configuration and click on it, you should see a screen similar to the one shown below.



Step 2: Double click on Wired connection 1 to open its properties.



Step 3: Change the Method from Automatic to Manual and click on add additional static address.

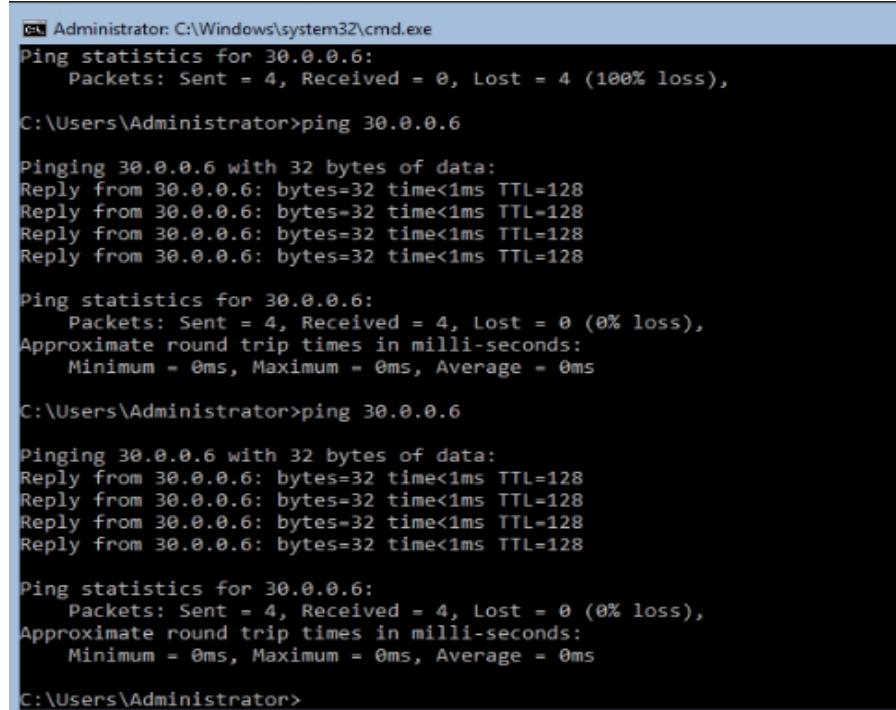


Fill in the information as shown in the above screen and press on Save.

Establishing connection between Various OS

Pinging from Windows Server OS CLI To Windows Server OS GUI

Step 1: While on the command line enter command “ping 30.0.0.6” to ping to GUI server OS. If there is a connection, the output of the command will look something similar to the screenshot shown below.



```
Administrator: C:\Windows\system32\cmd.exe
Ping statistics for 30.0.0.6:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\Users\Administrator>ping 30.0.0.6

Pinging 30.0.0.6 with 32 bytes of data:
Reply from 30.0.0.6: bytes=32 time<1ms TTL=128

Ping statistics for 30.0.0.6:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Administrator>ping 30.0.0.6

Pinging 30.0.0.6 with 32 bytes of data:
Reply from 30.0.0.6: bytes=32 time<1ms TTL=128

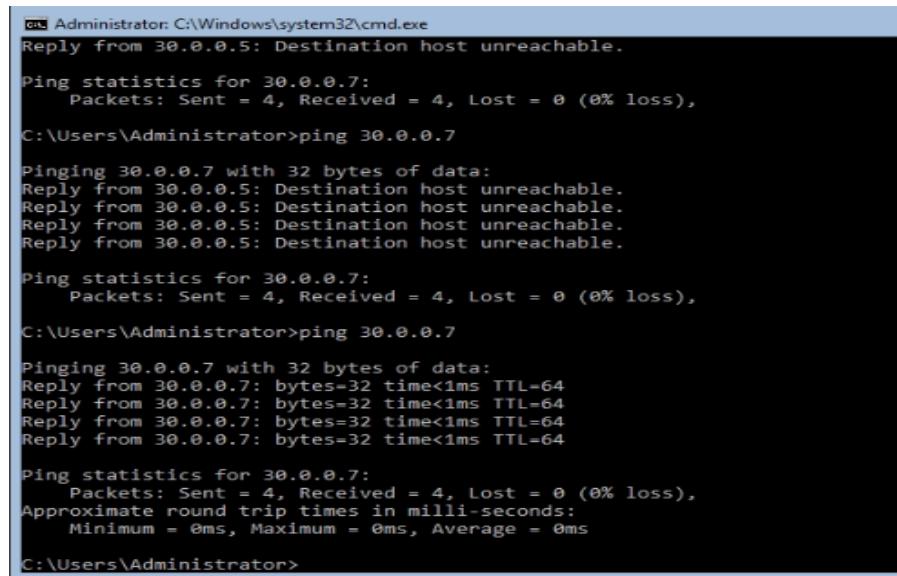
Ping statistics for 30.0.0.6:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Administrator>
```

This shows that the CLI Server OS was able to successfully send a message to the GUI server OS.

Pinging from Windows Server OS CLI To Kali OS

Step 1: While on the command line enter command “ping 30.0.0.7” to ping to Kali OS. If there is a connection, the output of the command will look something similar to the screenshot shown below.



```

Administrator: C:\Windows\system32\cmd.exe
Reply from 30.0.0.5: Destination host unreachable.

Ping statistics for 30.0.0.7:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
C:\Users\Administrator>ping 30.0.0.7

Pinging 30.0.0.7 with 32 bytes of data:
Reply from 30.0.0.5: Destination host unreachable.

Ping statistics for 30.0.0.7:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
C:\Users\Administrator>ping 30.0.0.7

Pinging 30.0.0.7 with 32 bytes of data:
Reply from 30.0.0.7: bytes=32 time<1ms TTL=64

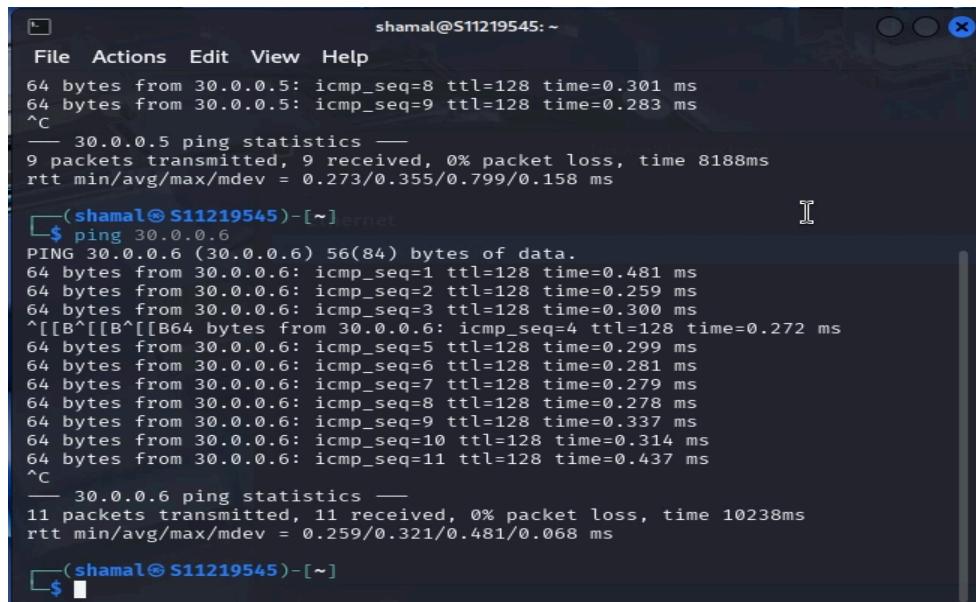
Ping statistics for 30.0.0.7:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\Users\Administrator>

```

This shows that the CLI Server OS was able to successfully send a message to the Kali OS.

Pinging from Kali OS To Windows Server OS GUI

Step 1: Open the terminal and enter command “ping 30.0.0.6” to ping to GUI server OS. If there is a connection, the output of the command will look something similar to the screenshot shown below.



```

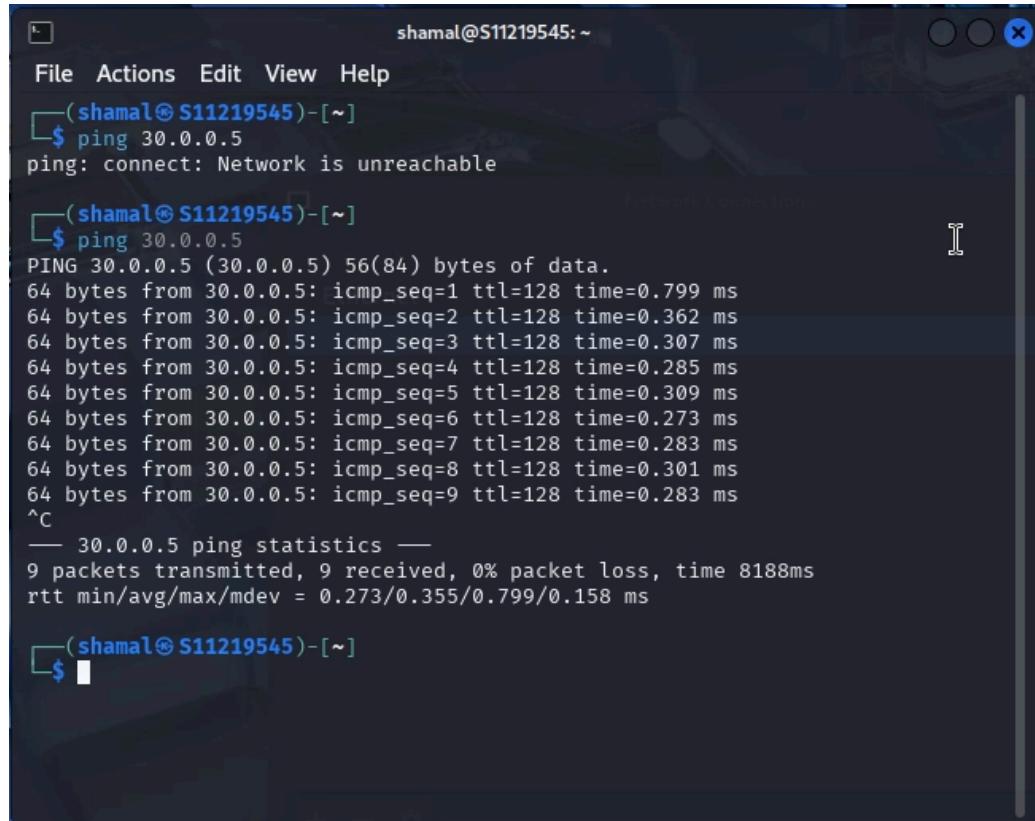
shamal@S11219545: ~
File Actions Edit View Help
64 bytes from 30.0.0.5: icmp_seq=8 ttl=128 time=0.301 ms
64 bytes from 30.0.0.5: icmp_seq=9 ttl=128 time=0.283 ms
^C
--- 30.0.0.5 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8188ms
rtt min/avg/max/mdev = 0.273/0.355/0.799/0.158 ms
(shamal@S11219545)-[~]net
$ ping 30.0.0.6
PING 30.0.0.6 (30.0.0.6) 56(84) bytes of data.
64 bytes from 30.0.0.6: icmp_seq=1 ttl=128 time=0.481 ms
64 bytes from 30.0.0.6: icmp_seq=2 ttl=128 time=0.259 ms
64 bytes from 30.0.0.6: icmp_seq=3 ttl=128 time=0.300 ms
^[[B^[[B64 bytes from 30.0.0.6: icmp_seq=4 ttl=128 time=0.272 ms
64 bytes from 30.0.0.6: icmp_seq=5 ttl=128 time=0.299 ms
64 bytes from 30.0.0.6: icmp_seq=6 ttl=128 time=0.281 ms
64 bytes from 30.0.0.6: icmp_seq=7 ttl=128 time=0.279 ms
64 bytes from 30.0.0.6: icmp_seq=8 ttl=128 time=0.278 ms
64 bytes from 30.0.0.6: icmp_seq=9 ttl=128 time=0.337 ms
64 bytes from 30.0.0.6: icmp_seq=10 ttl=128 time=0.314 ms
64 bytes from 30.0.0.6: icmp_seq=11 ttl=128 time=0.437 ms
^C
--- 30.0.0.6 ping statistics ---
11 packets transmitted, 11 received, 0% packet loss, time 10238ms
rtt min/avg/max/mdev = 0.259/0.321/0.481/0.068 ms
(shamal@S11219545)-[~]
$ 

```

This shows that the Kali OS was able to successfully send a message to the GUI Server OS.

Pinging from Kali OS To Windows Server OS CLI

Step 1: Open the terminal and enter command “ping 30.0.0.5” to ping to CLI server OS. If there is a connection, the output of the command will look something similar to the screenshot shown below.



The screenshot shows a terminal window titled "shamal@S11219545: ~". The window has a menu bar with "File", "Actions", "Edit", "View", and "Help". Below the menu is a command-line interface. The user has run the command "ping 30.0.0.5". The output shows the ping command being sent to the target IP address, with 9 packets transmitted and 9 received, resulting in 0% packet loss. The time taken for the round trip is 8188ms, with an average of 0.273ms, maximum of 0.355ms, and minimum of 0.158ms.

```
shamal@S11219545: ~
File Actions Edit View Help
(shamal@S11219545)-[~]
$ ping 30.0.0.5
ping: connect: Network is unreachable

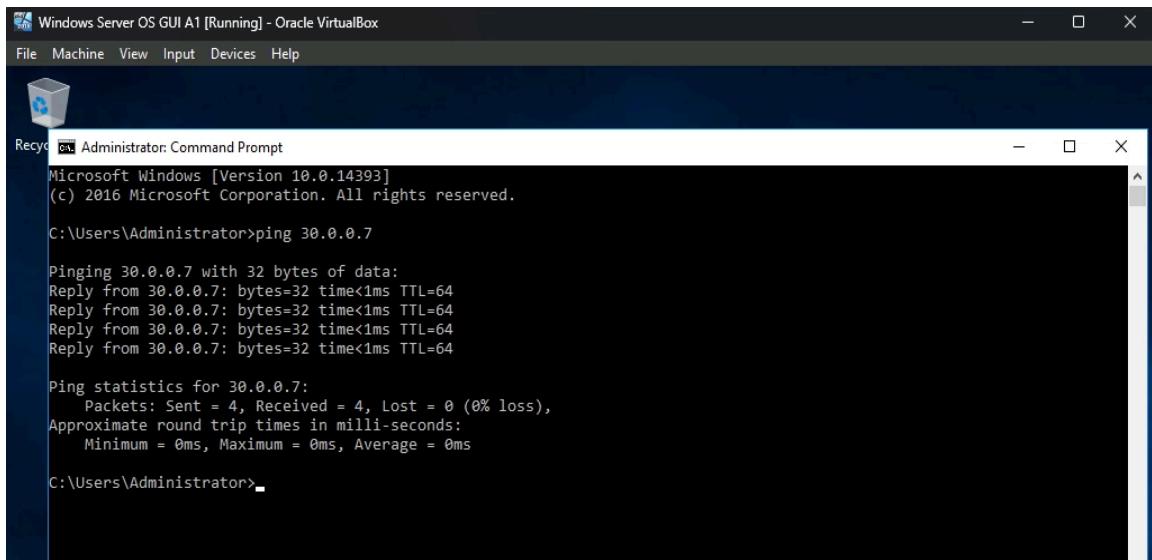
(shamal@S11219545)-[~]
$ ping 30.0.0.5
PING 30.0.0.5 (30.0.0.5) 56(84) bytes of data.
64 bytes from 30.0.0.5: icmp_seq=1 ttl=128 time=0.799 ms
64 bytes from 30.0.0.5: icmp_seq=2 ttl=128 time=0.362 ms
64 bytes from 30.0.0.5: icmp_seq=3 ttl=128 time=0.307 ms
64 bytes from 30.0.0.5: icmp_seq=4 ttl=128 time=0.285 ms
64 bytes from 30.0.0.5: icmp_seq=5 ttl=128 time=0.309 ms
64 bytes from 30.0.0.5: icmp_seq=6 ttl=128 time=0.273 ms
64 bytes from 30.0.0.5: icmp_seq=7 ttl=128 time=0.283 ms
64 bytes from 30.0.0.5: icmp_seq=8 ttl=128 time=0.301 ms
64 bytes from 30.0.0.5: icmp_seq=9 ttl=128 time=0.283 ms
^C
--- 30.0.0.5 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8188ms
rtt min/avg/max/mdev = 0.273/0.355/0.799/0.158 ms

(shamal@S11219545)-[~]
$
```

This shows that the Kali OS was able to successfully send a message to the CLI Server OS.

Pinging from Windows Server OS GUI To Kali OS

Step 1: Open the terminal and enter command “ping 30.0.0.7” to ping to Kali OS. If there is a connection, the output of the command will look something similar to the screenshot shown below.



Administrator: Command Prompt

```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ping 30.0.0.7

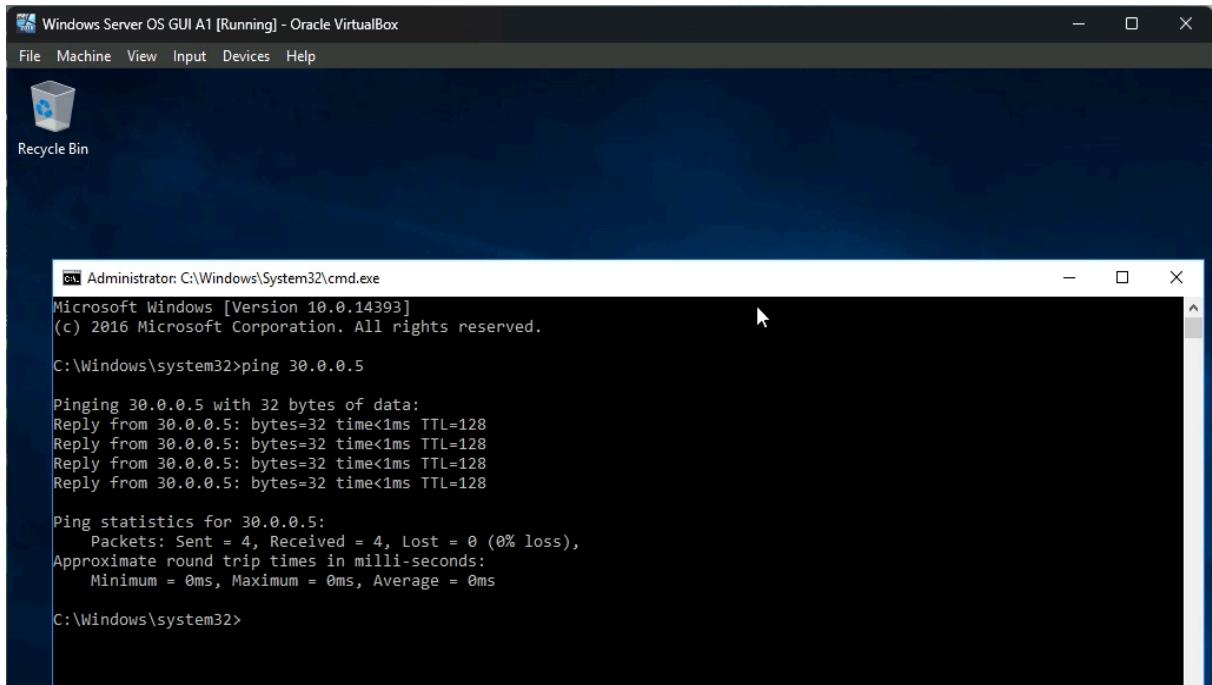
Pinging 30.0.0.7 with 32 bytes of data:
Reply from 30.0.0.7: bytes=32 time<1ms TTL=64

Ping statistics for 30.0.0.7:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Administrator>
```

Pinging from Windows Server OS GUI To Windows Server OS CLI

Step 1: Open the terminal and enter command “ping 30.0.0.5” to ping to Kali OS. If there is a connection, the output of the command will look something similar to the screenshot shown below.



Administrator: C:\Windows\System32\cmd.exe

```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>ping 30.0.0.5

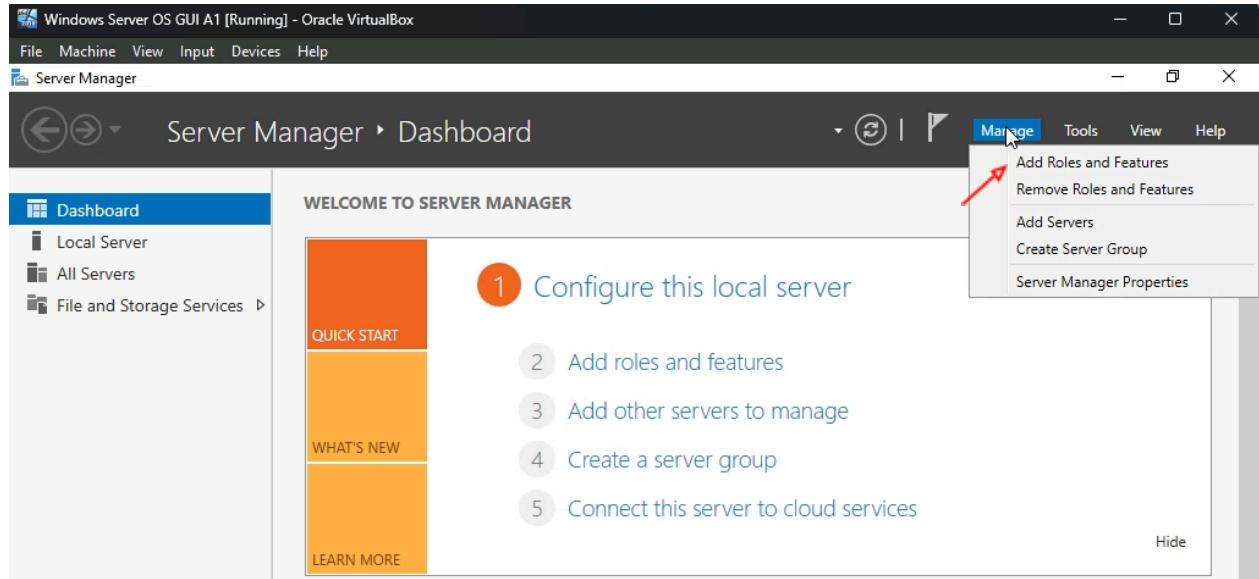
Pinging 30.0.0.5 with 32 bytes of data:
Reply from 30.0.0.5: bytes=32 time<1ms TTL=128

Ping statistics for 30.0.0.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

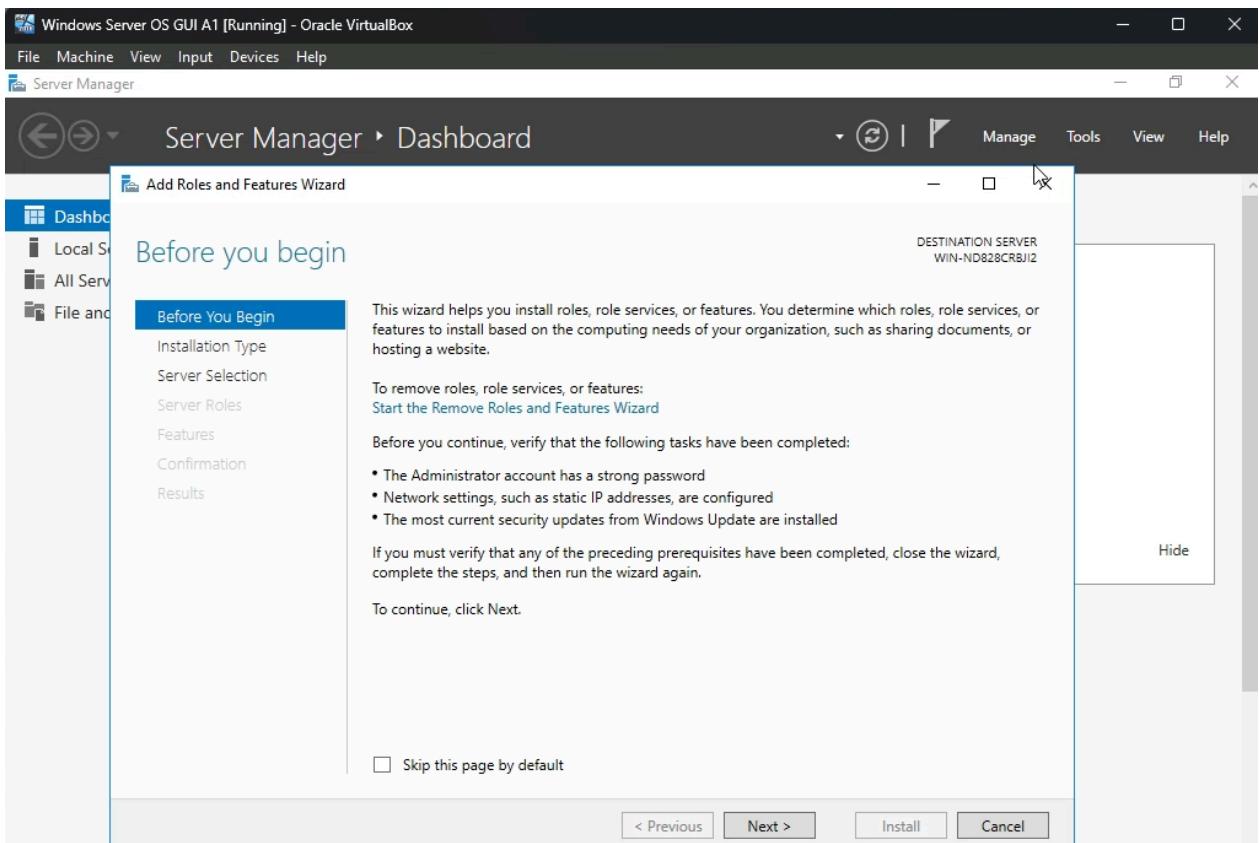
C:\Windows\system32>
```

Setup Primary domain for Windows Server GUI

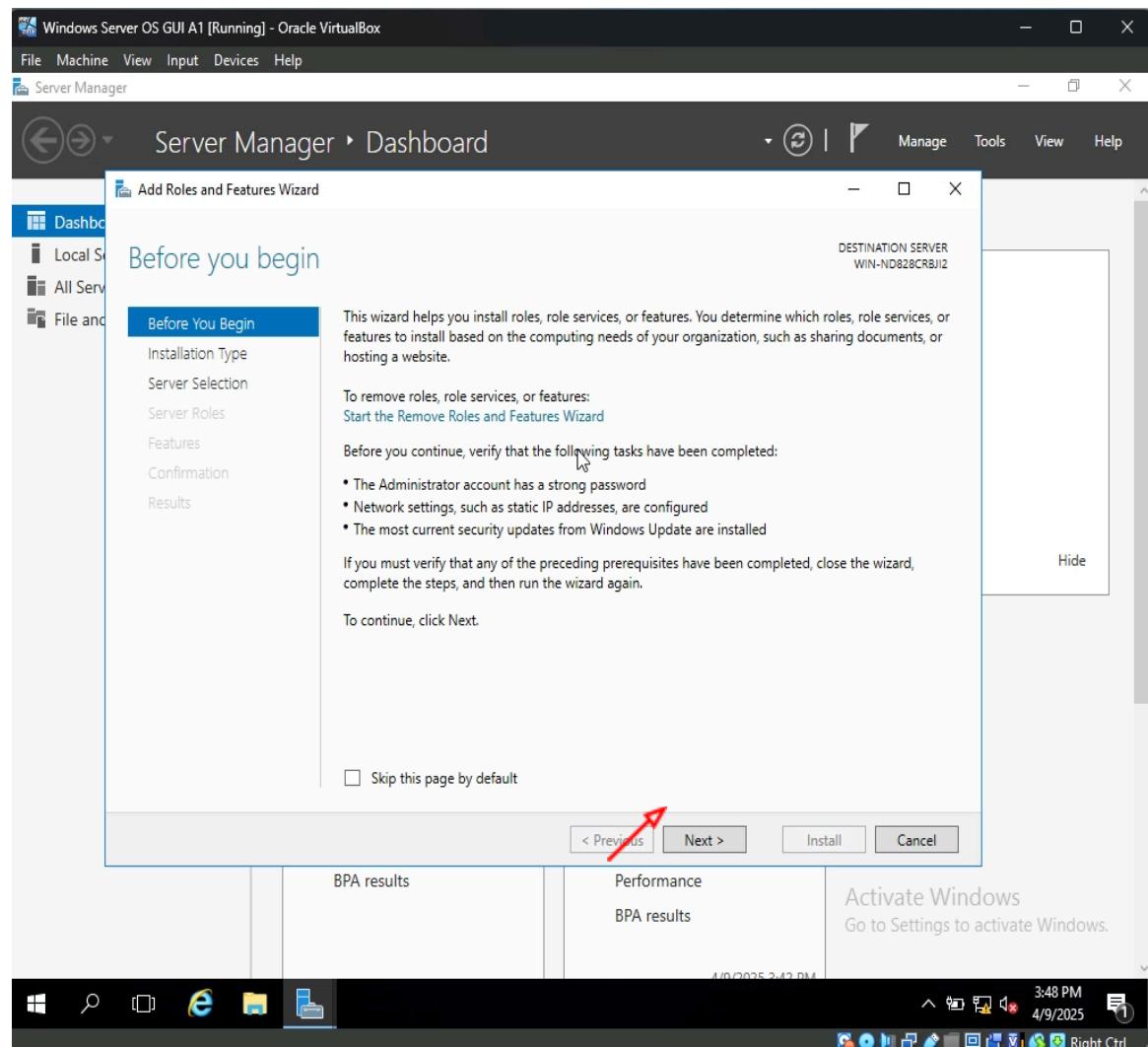
Step 1: Open the Server Manager Program and press Manage that located to the top right corner of the program.



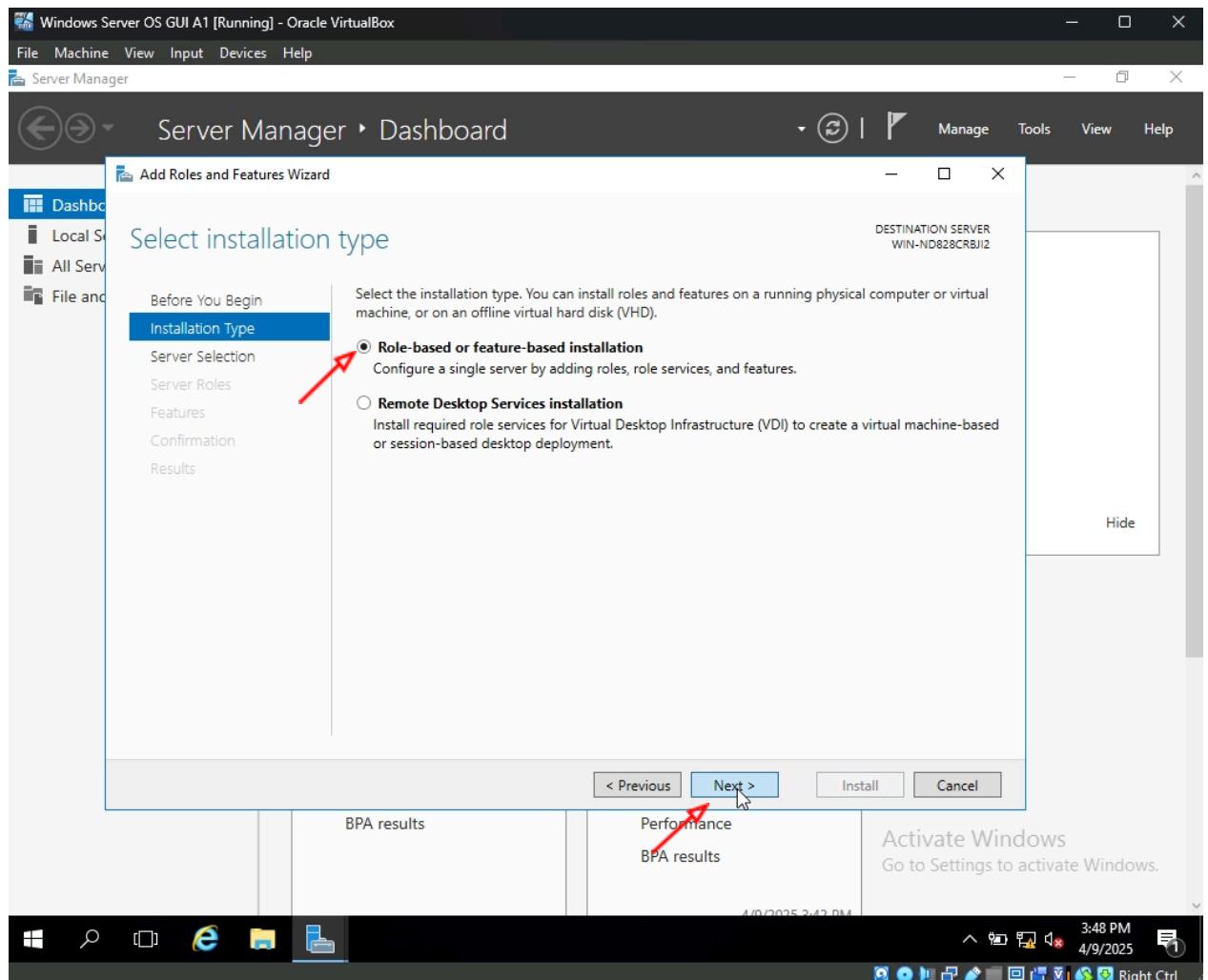
Next, click the “Add Roles and features” option to start the setup process of adding a user role and its related features to the server.



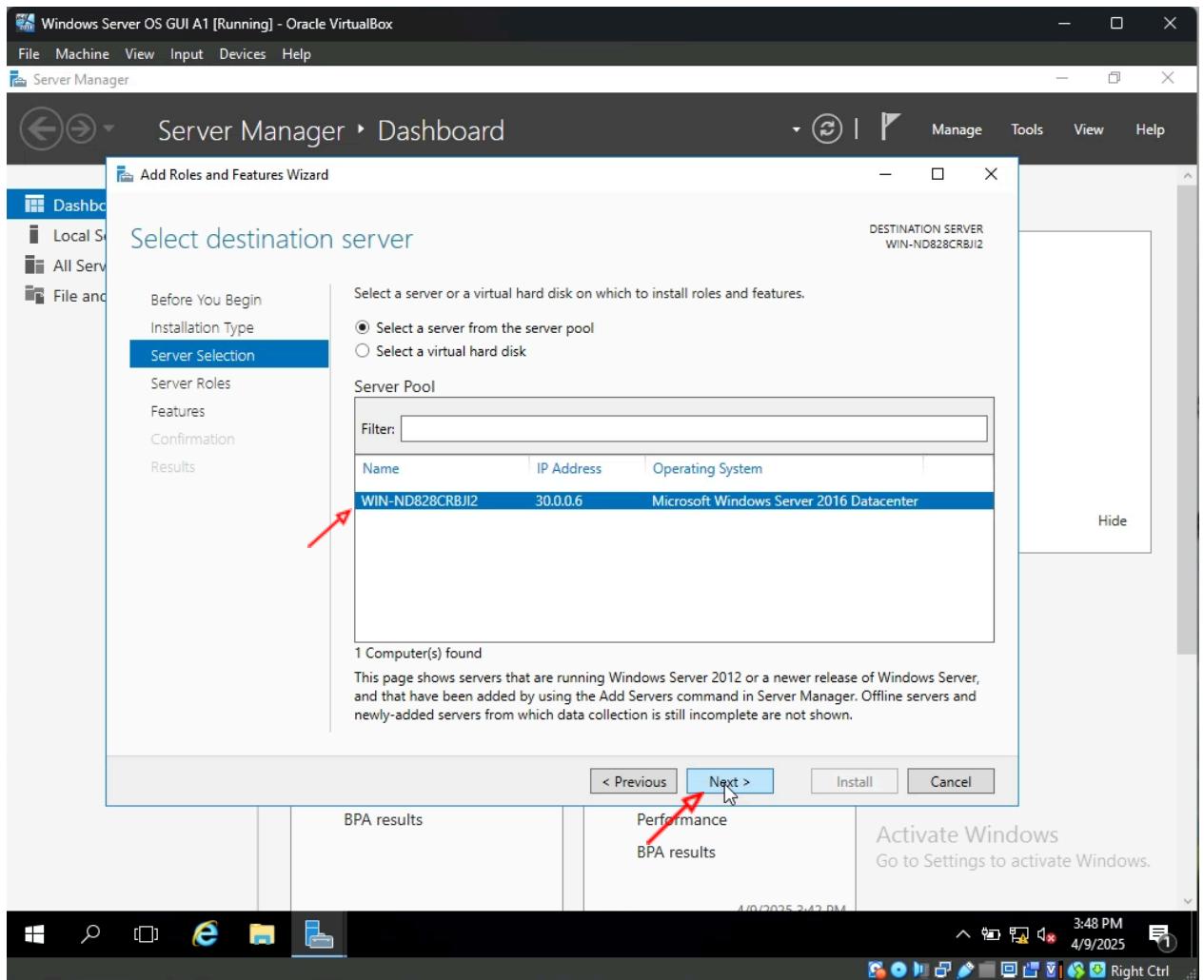
Step 2: This setup wizard will guide you through adding Roles and features to your server. Click Next.



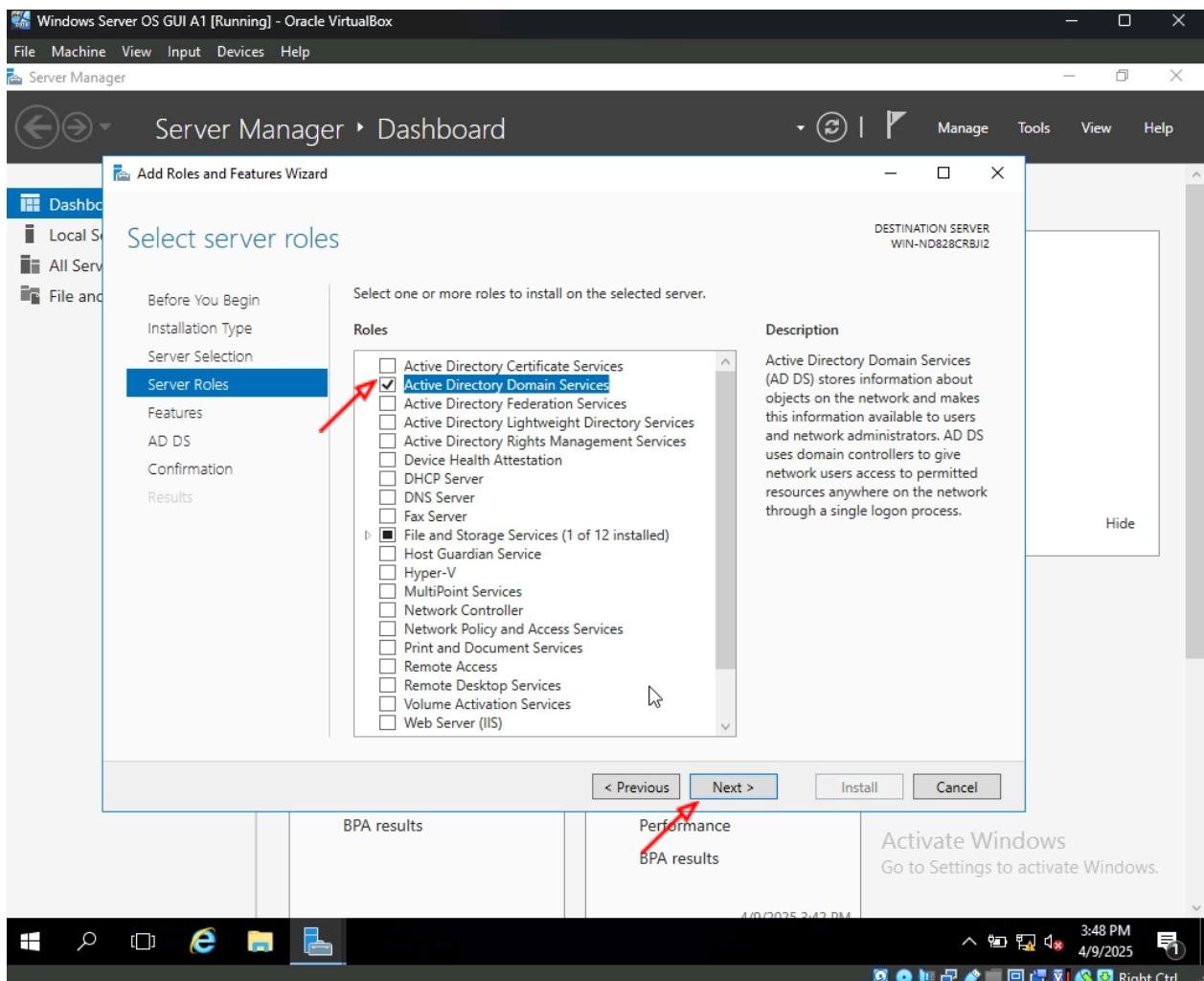
Step 3: For your installation type select the first radio button from the top and click next.



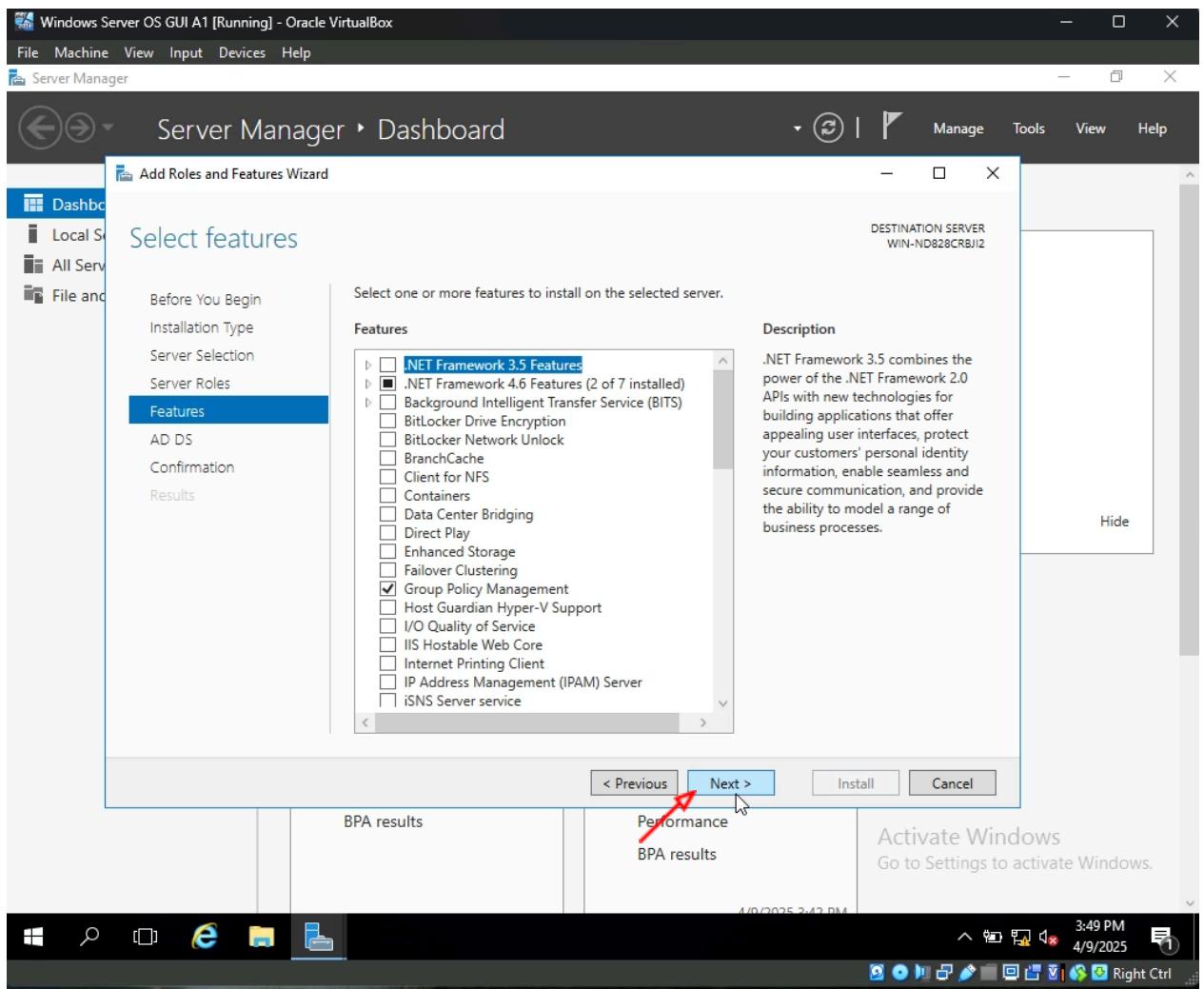
Step 4: Select your server destination drive where the installation will be downloaded to. Click Next.



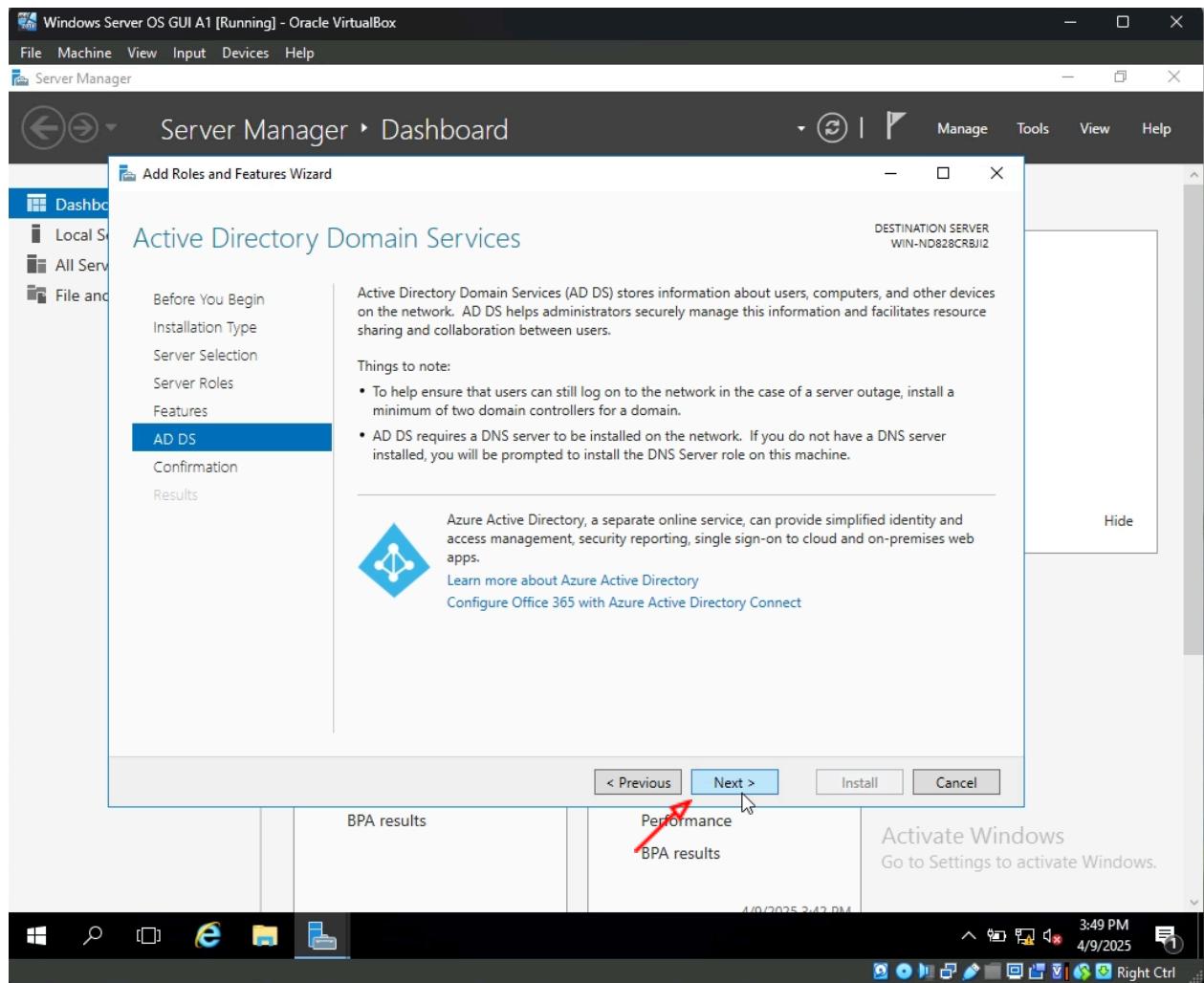
Step 5: For your server roles screen, ensure that the Active Directory Domain Services checkbox is checked and press Next.



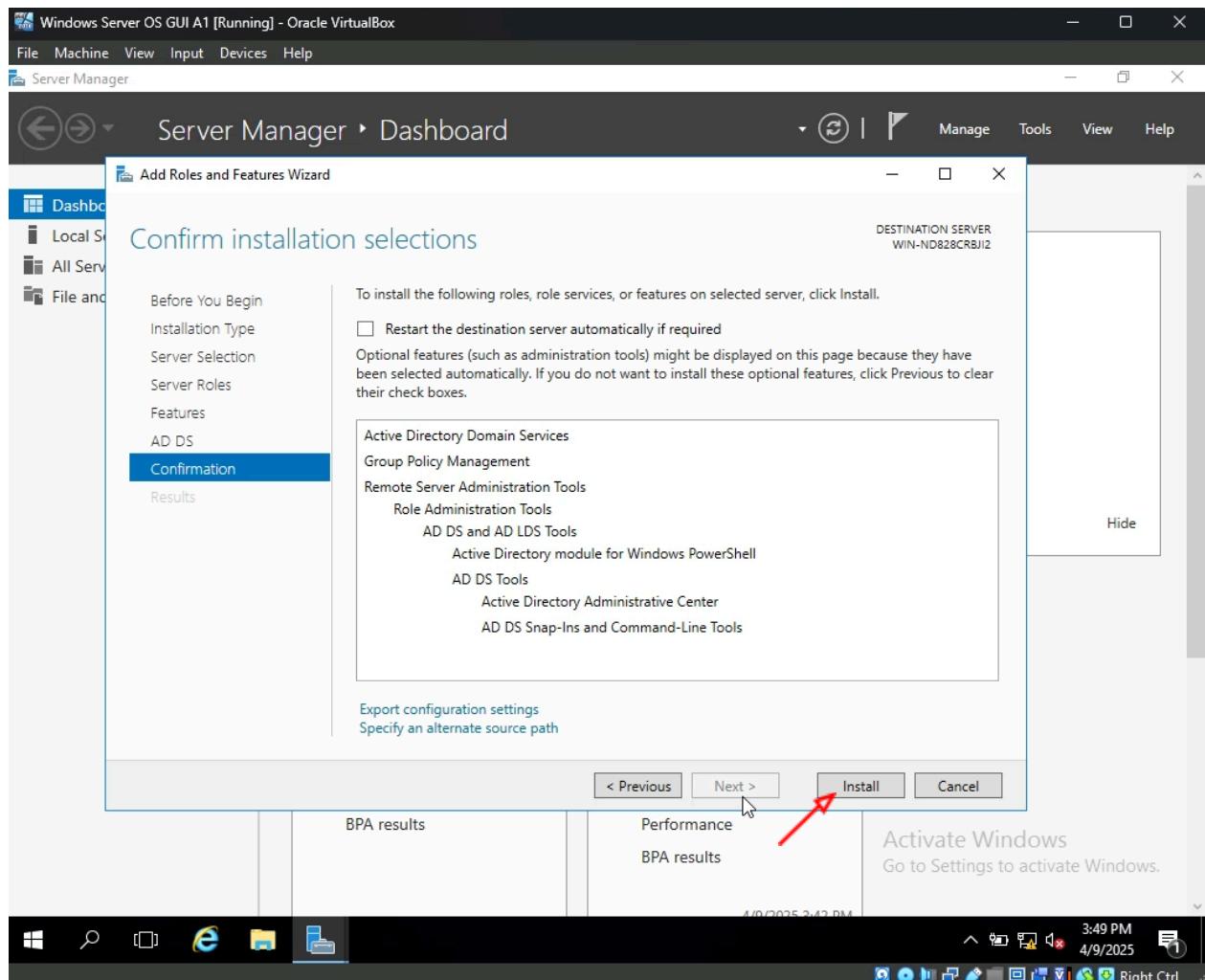
Step 6: For your server feature screen, you don't have to really select anything so just go ahead and press Next.



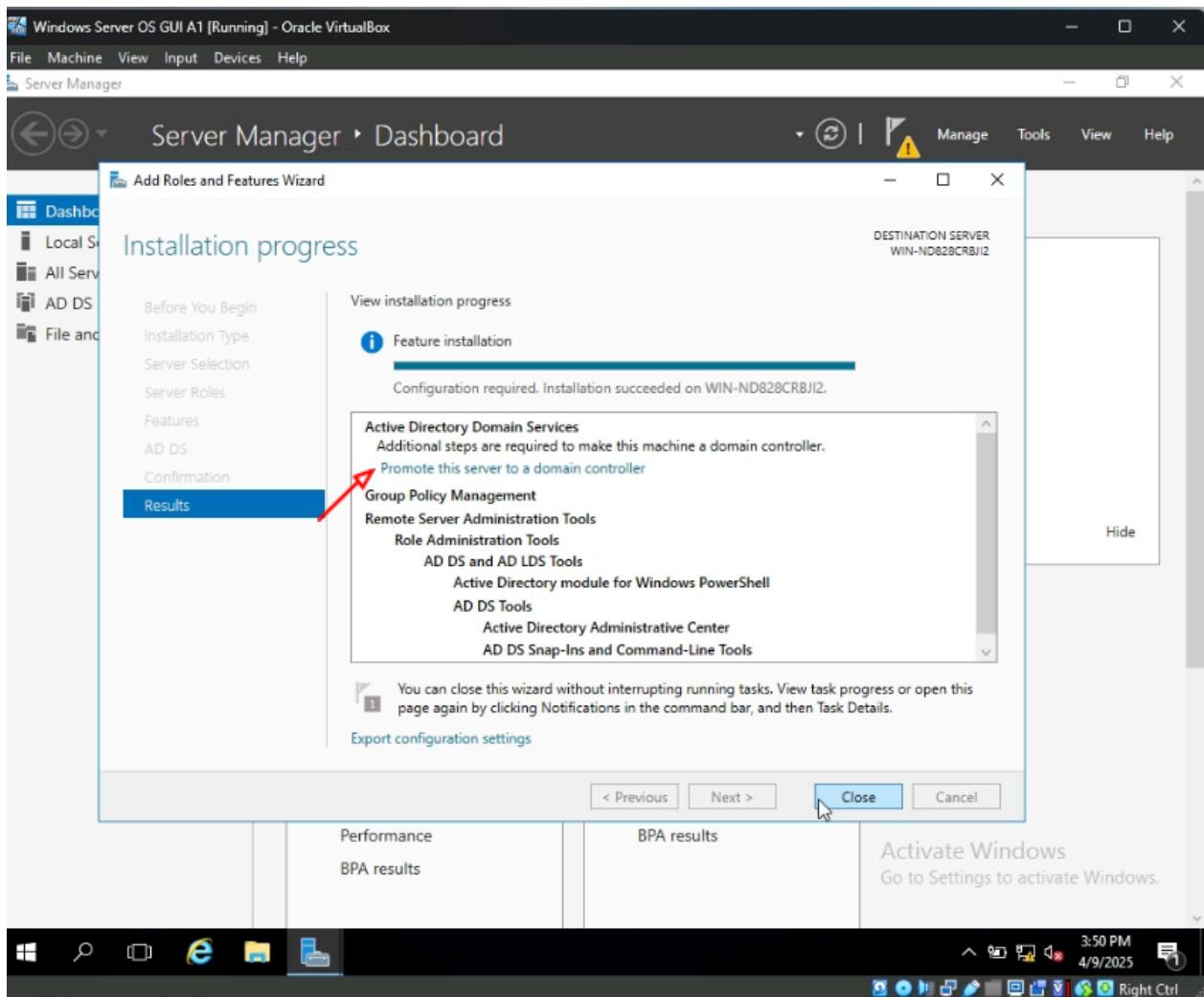
Step 7: The following screen gives you information about Active Directory Domain Services and anything important you should take note of. Click Next.



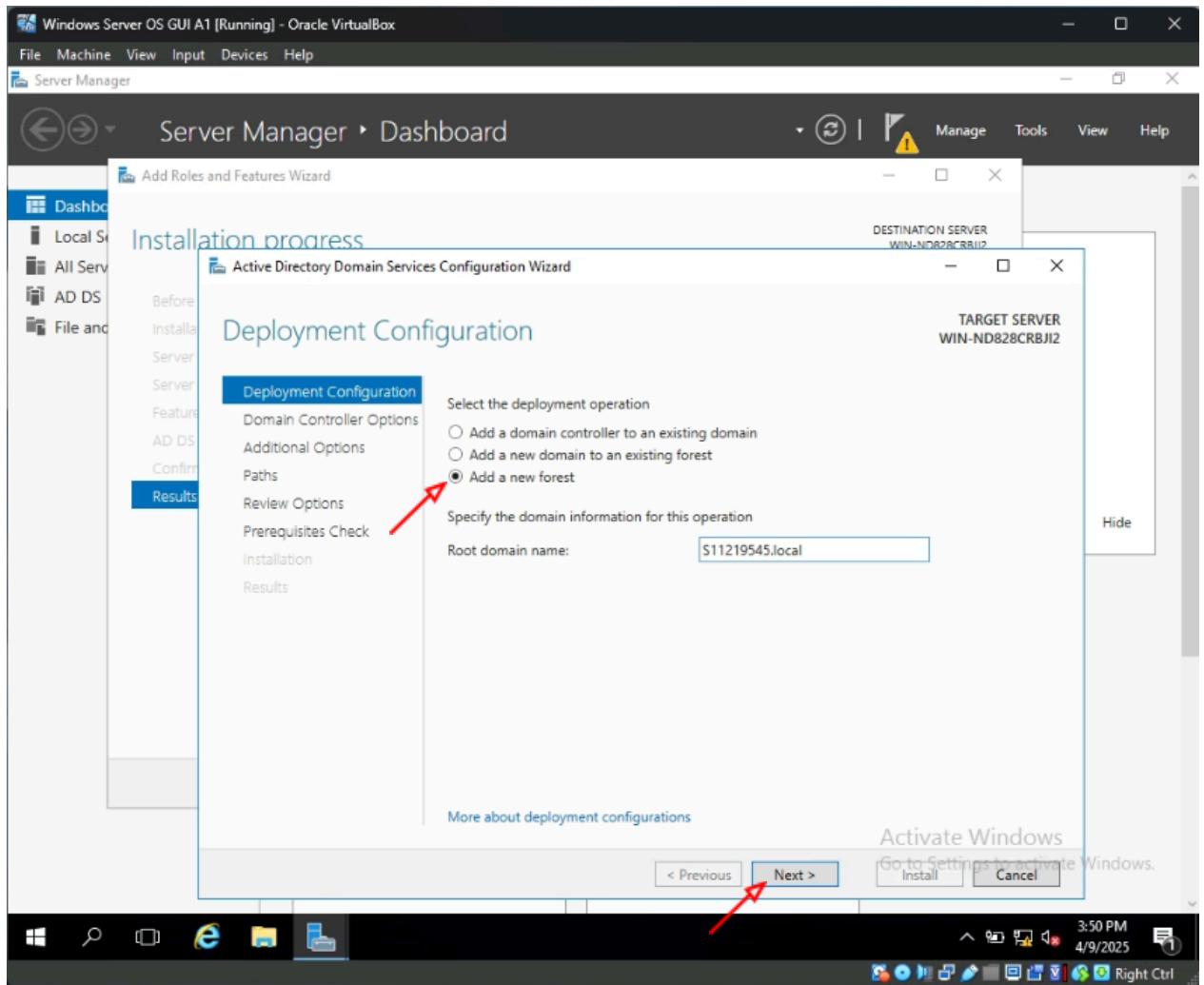
Step 8: The next screen just gives a brief rundown of what selections you have made and what features are about to be installed. Press install to begin the installation process.



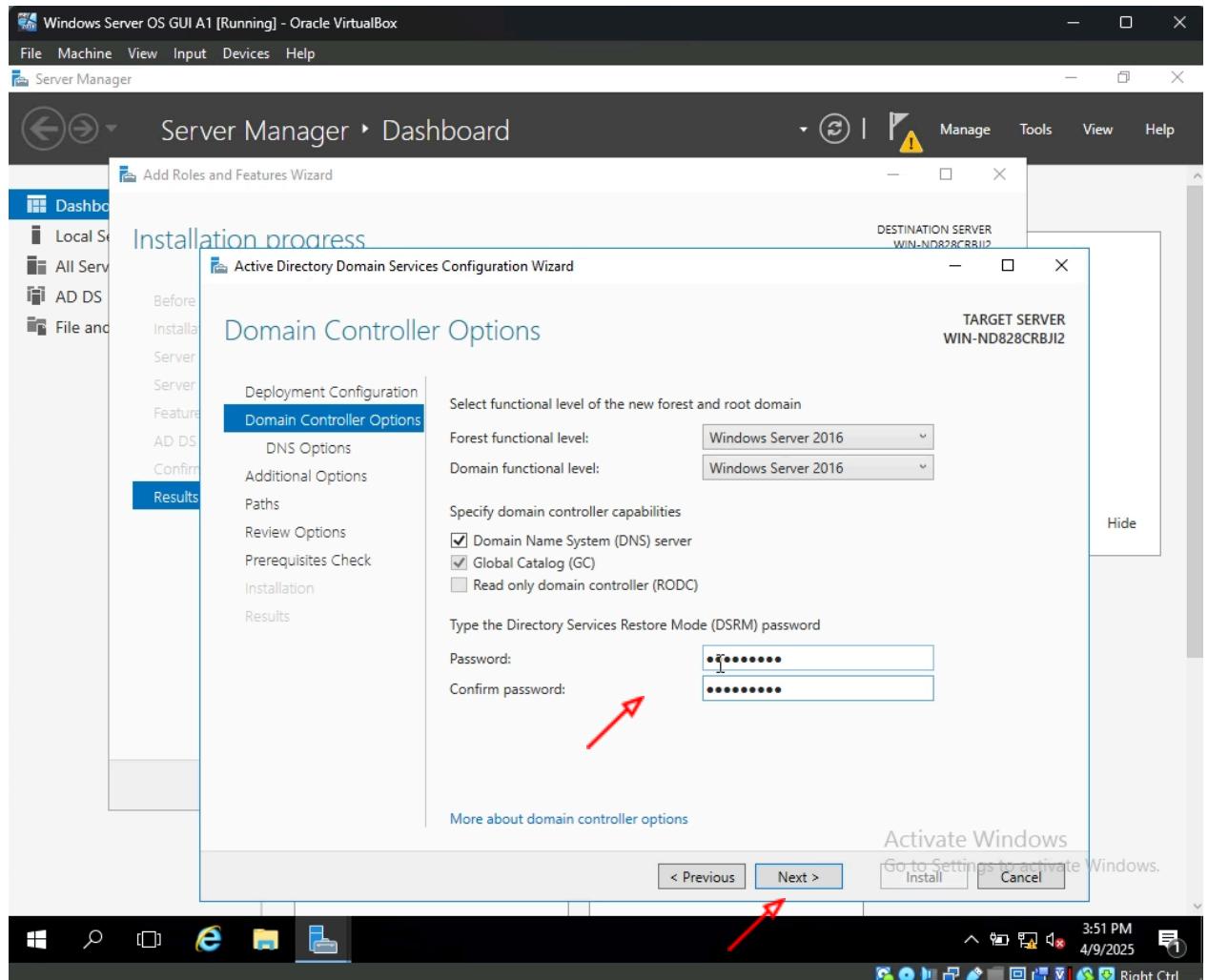
Step 9: After the installation has completed, the installation wizard will ask you to promote the following server to a domain controller. Click on the blue link to be directed to the Deployment Configuration screen.



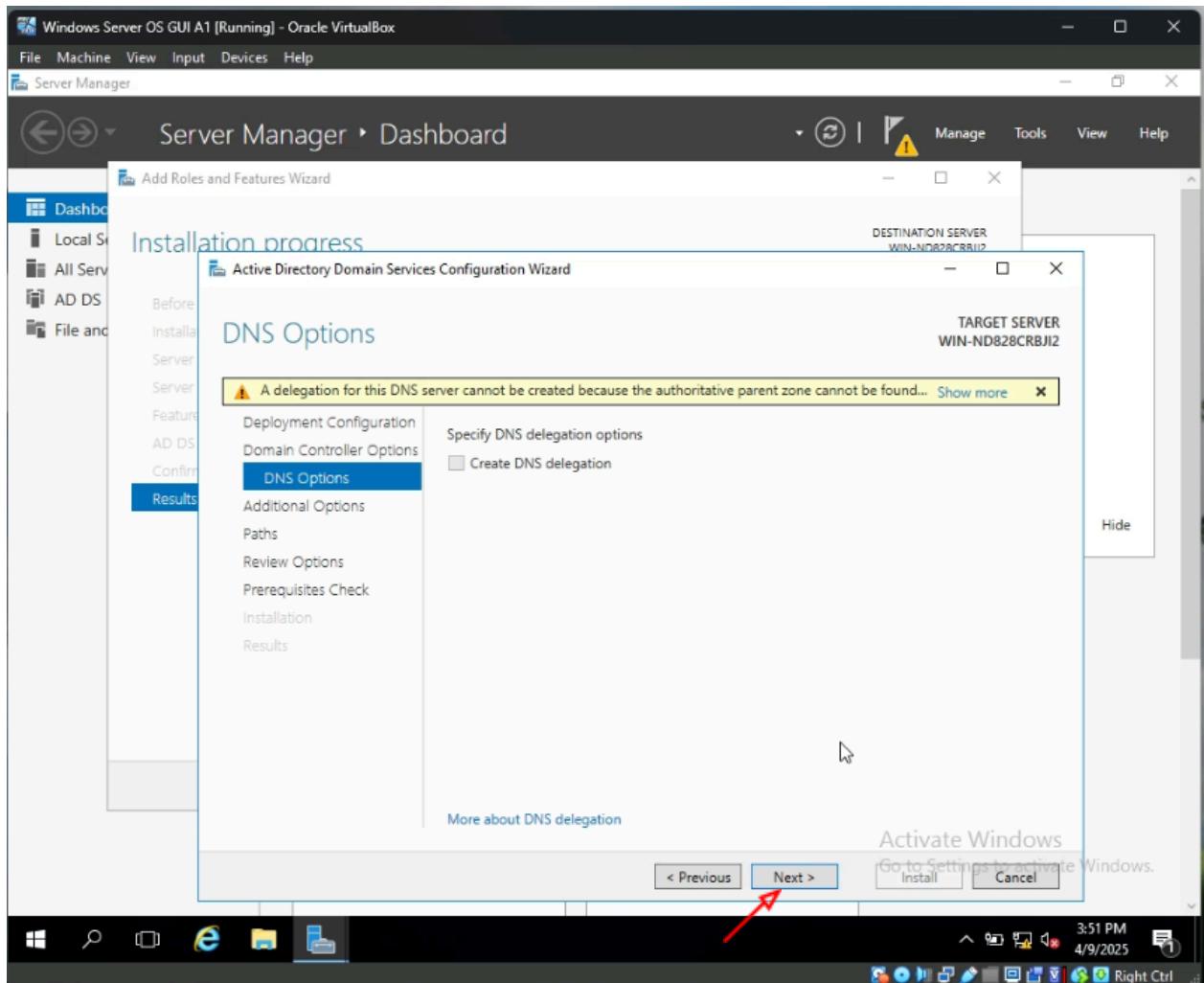
Step 10: Select the add a new forest radio button and add your root domain name. Root domain" name could be anything but it must end with".local", just like in the screen shown below. Click Next.



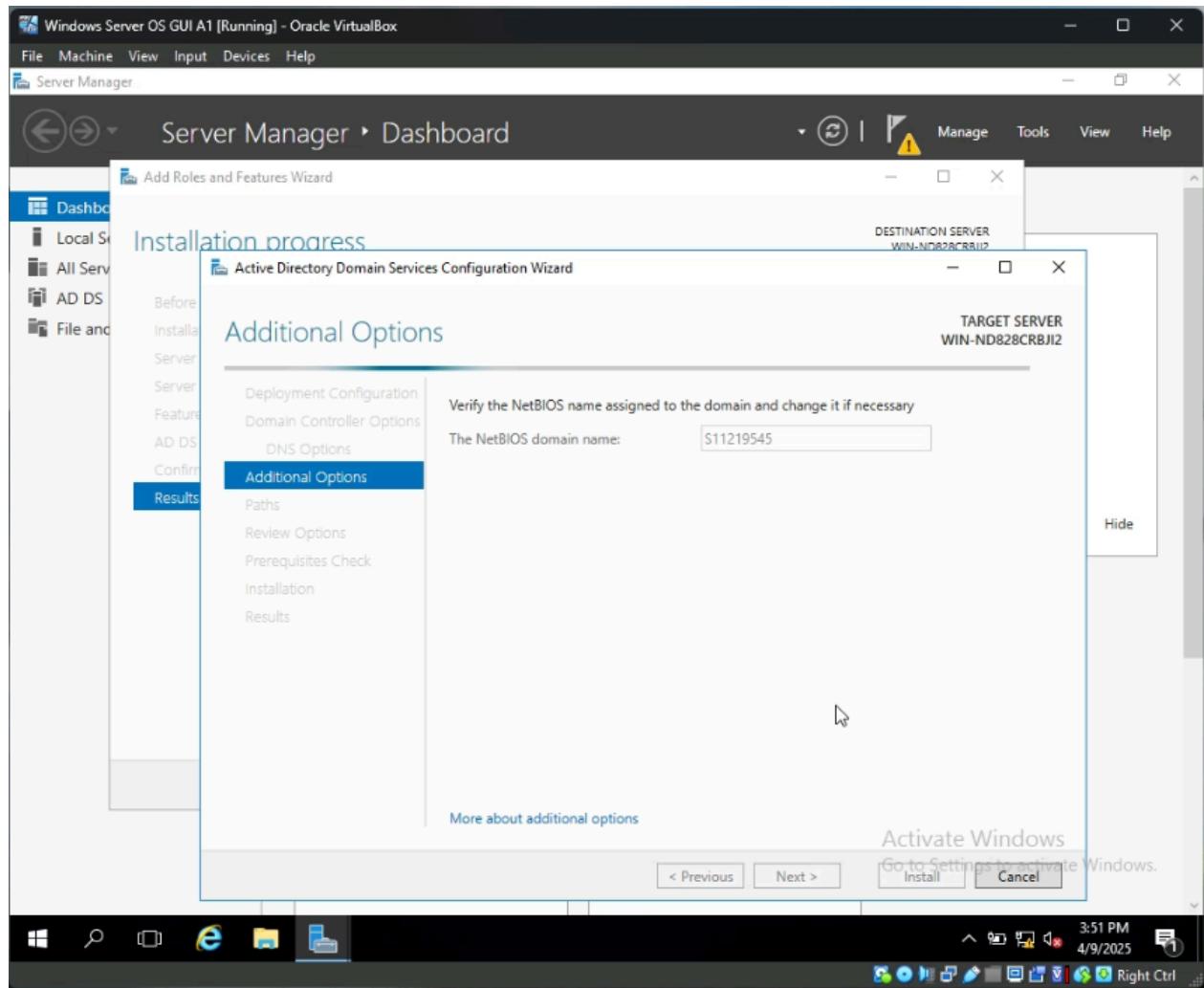
Step 11: Create a password for the domain controller. Once created, press Next.



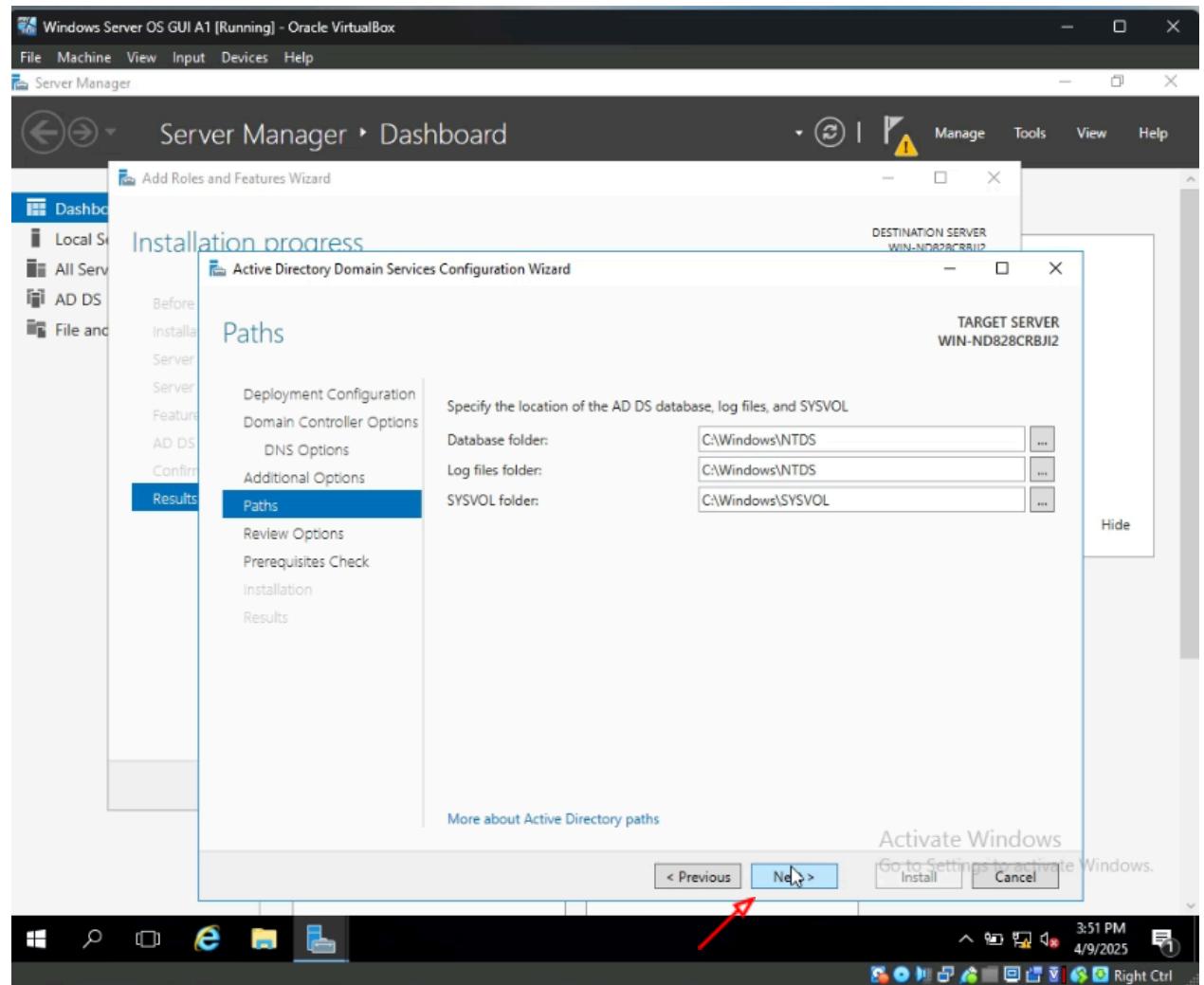
Step 12: For your DNS option nothing needs to be changed or enabled so go ahead and press Next.



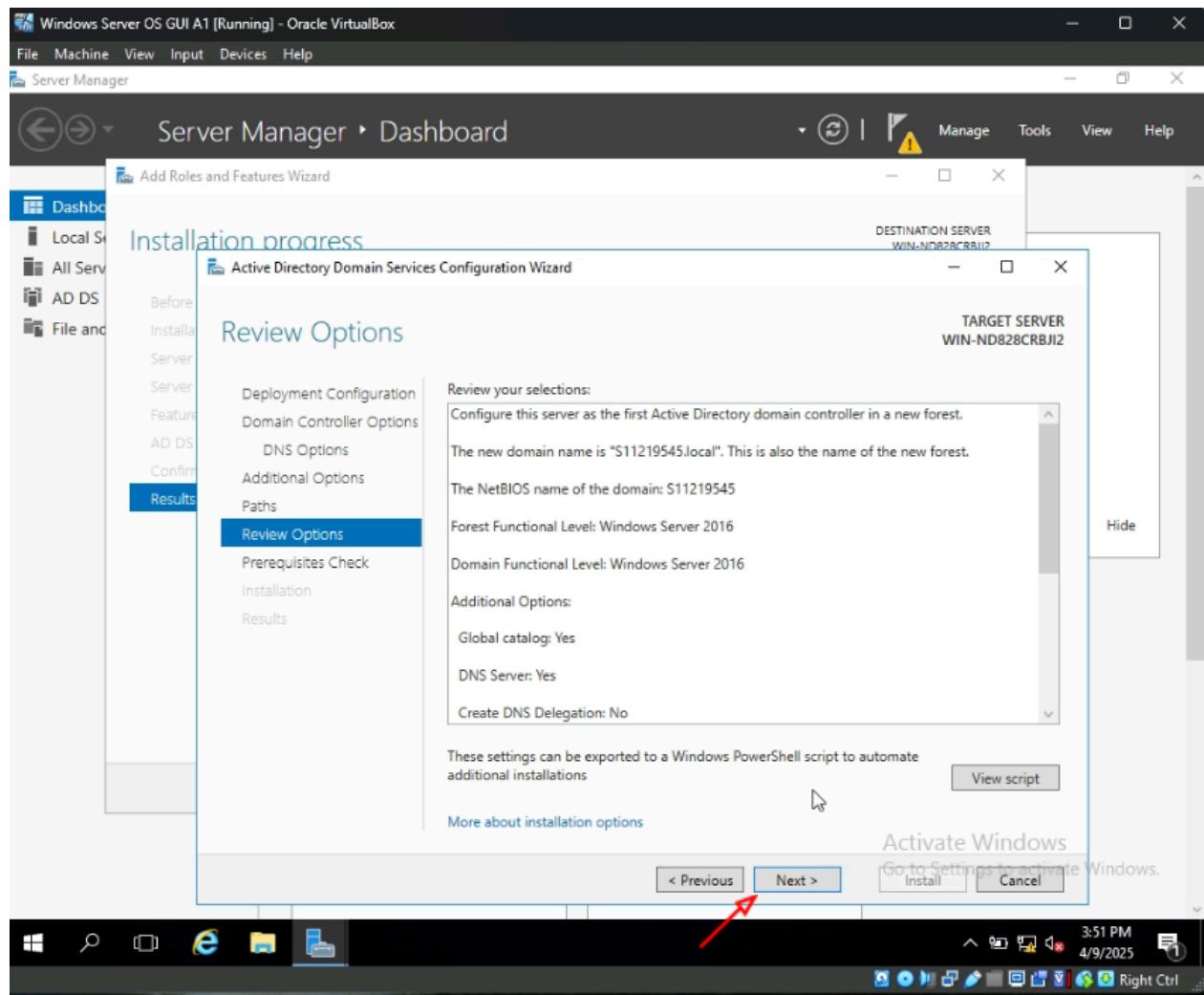
Step 13: Your NetBIOS name will be configured automatically so no input from your side is needed hence, press Next.



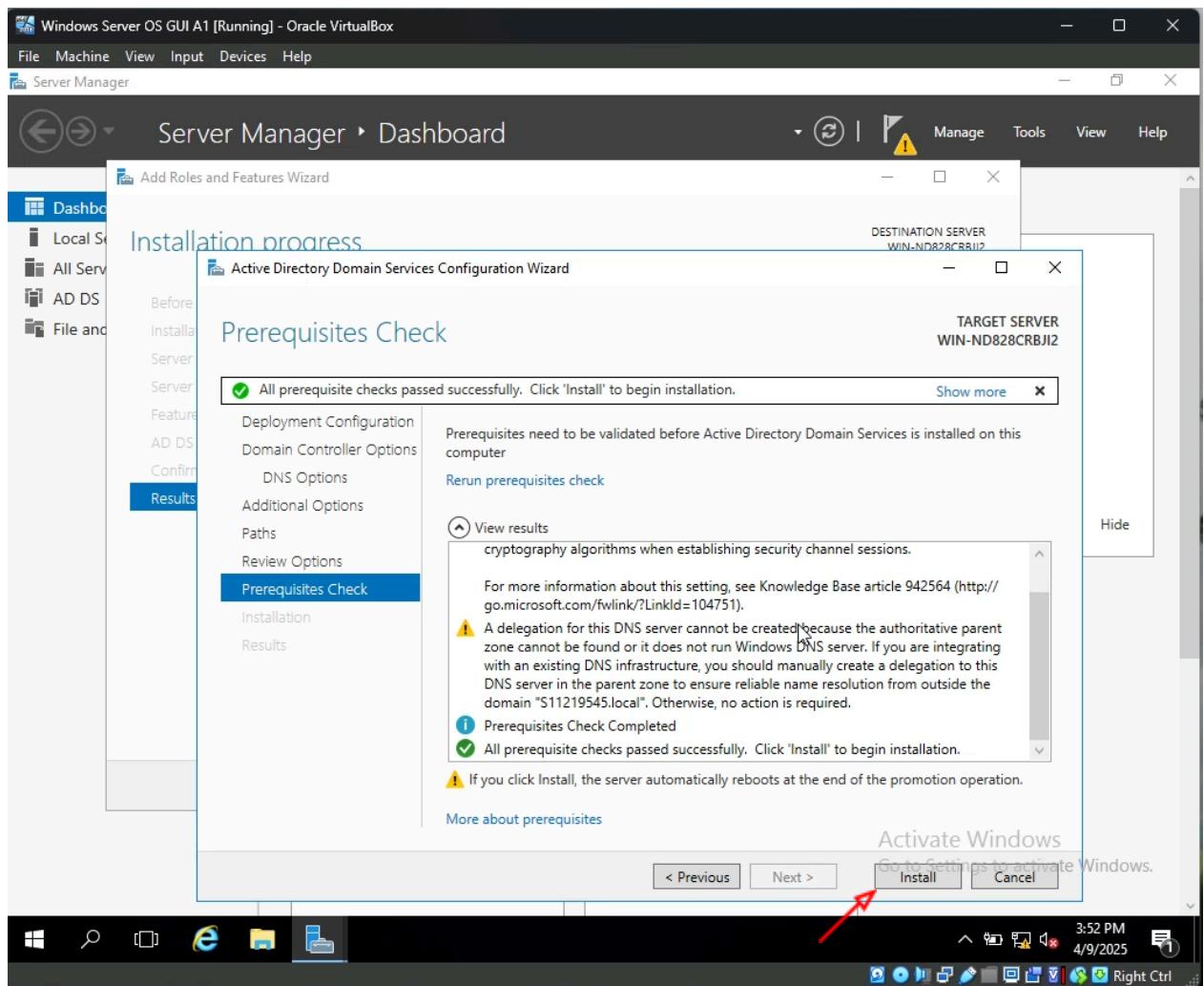
Step 14: For your path, you can use the default ones provided by the setup wizard for simplicity, unless you want to change it. Go ahead and press next.



Step 15: The next screen shows you the options you have chosen and you can go through it to ensure you are installing the right items. Press Next.

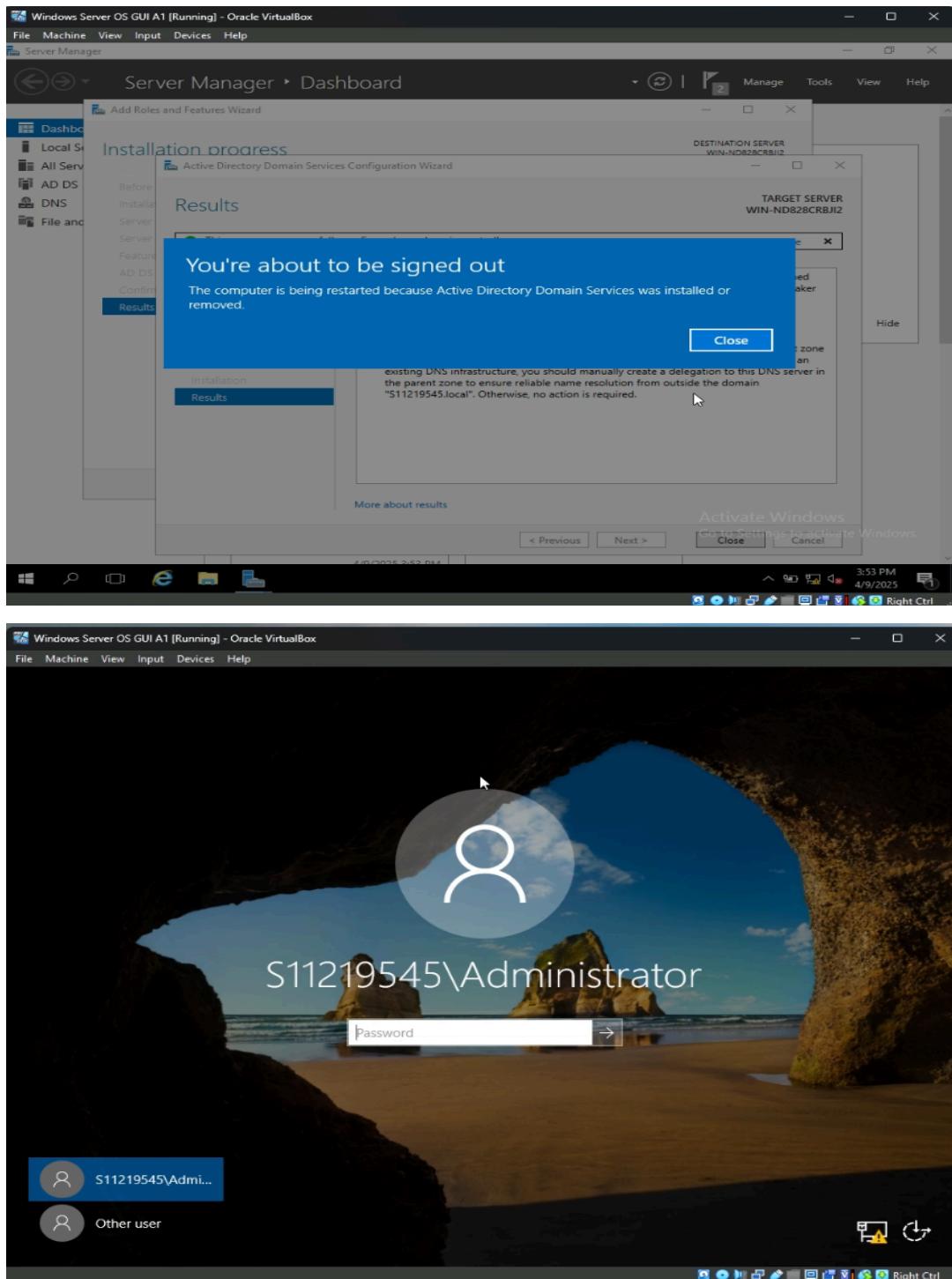


Step 16: The following screen is the last checkpoint to verify the items and to check whether the server is compatible. After this screen the final installation will occur. If everything is in order, press Next.



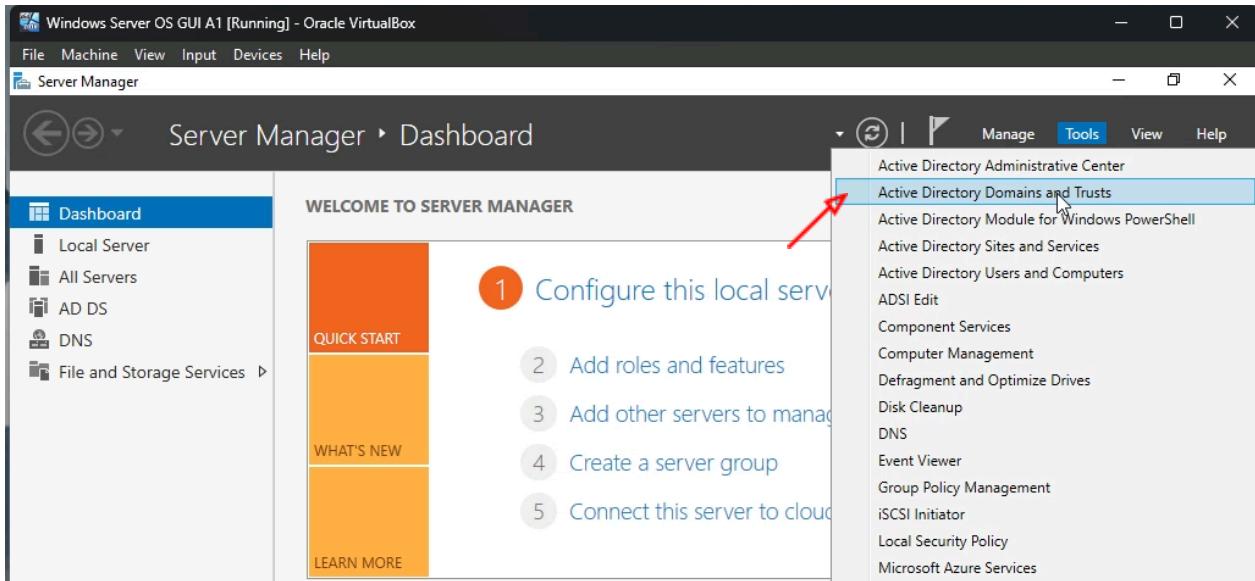
You will be prompted to restart at the end of the installation, go ahead and perform the reboot.

After the reboot, you should be able to see the following screen.



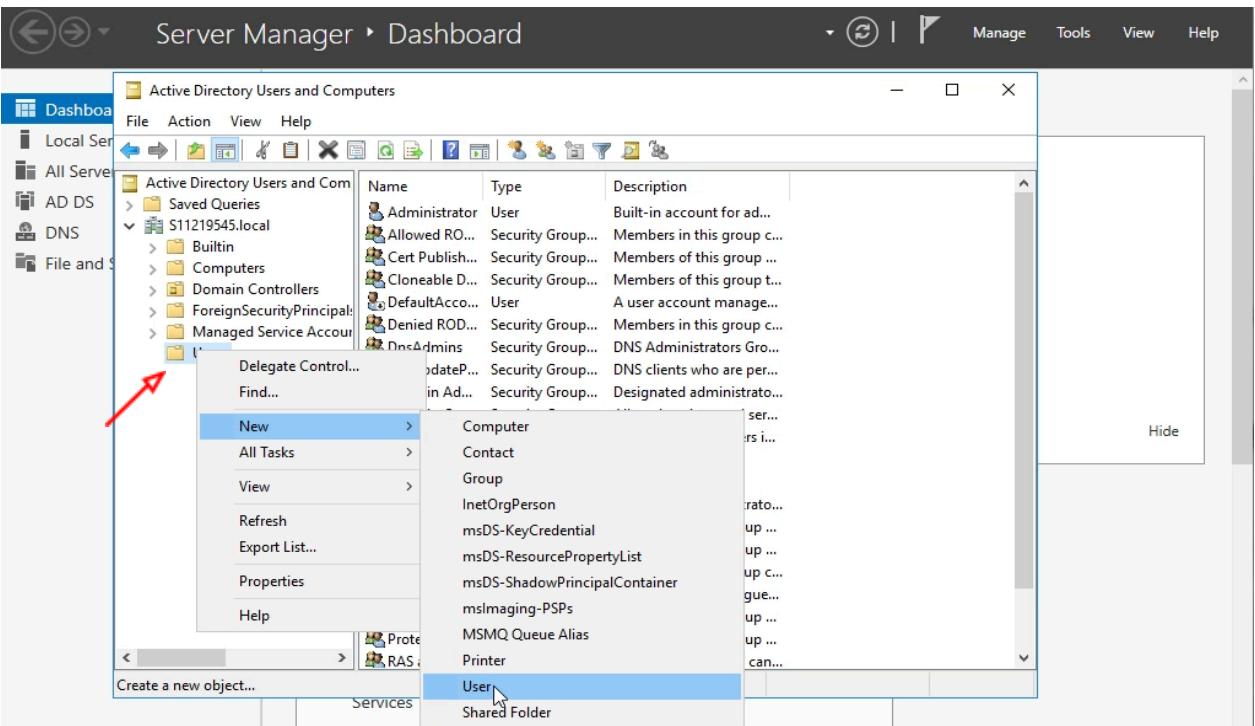
Setting Up Users In Windows GUI Server OS

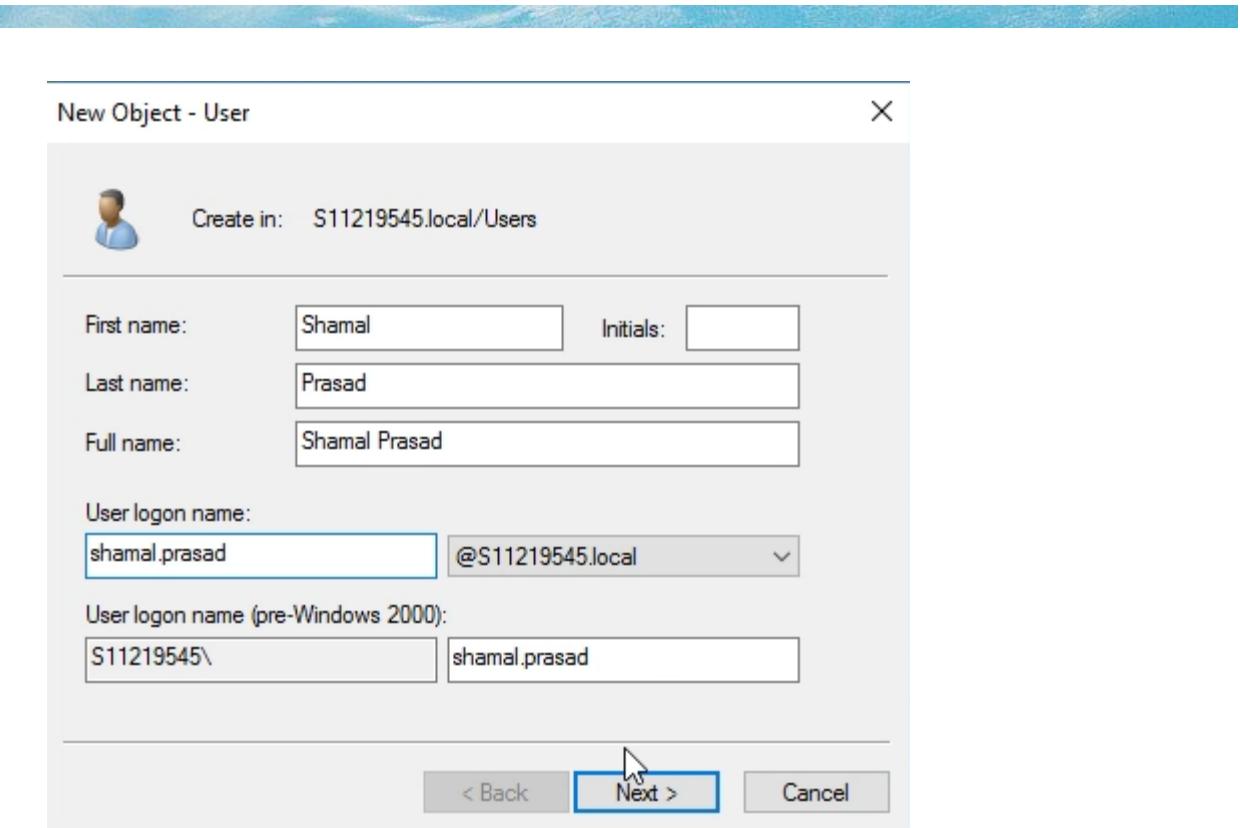
Step 1: Open the Server Manager Program and at the top right corner, press the Tools button to open a drop down list where you will be presented with the various tools the Server Manager offers. Select Active Directory Domain and Trust.



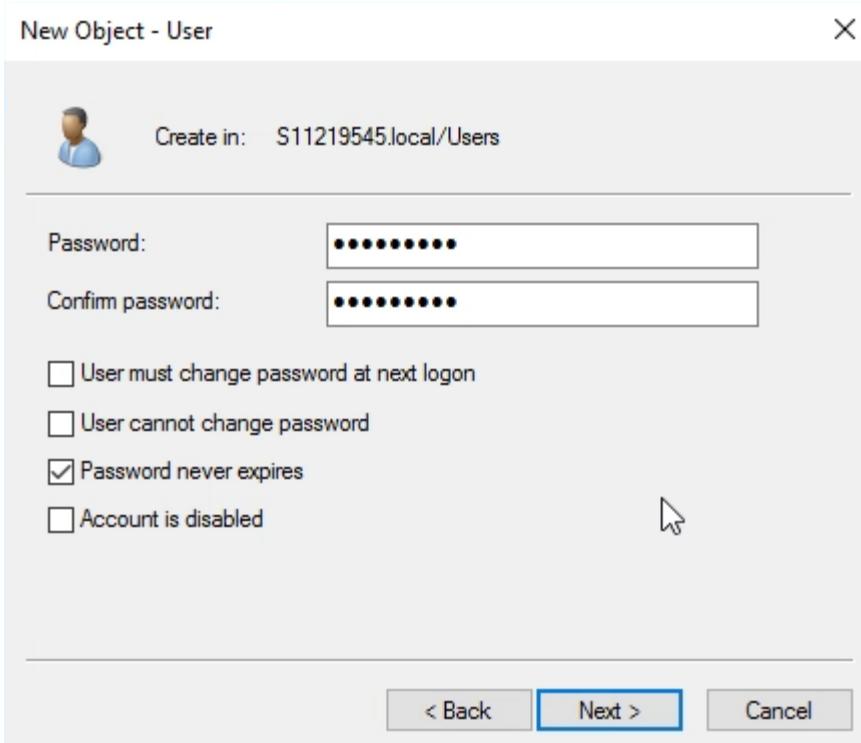
Step 2: Click on the domain tree name to expand the dropdown list, then right click on users to open another menu and in this menu look for new which will expand the menu list even further, in this new list look for users and click on it. The screen would look something like below.

Step 3: Fill in user details for the account you are trying to create. Click Next.

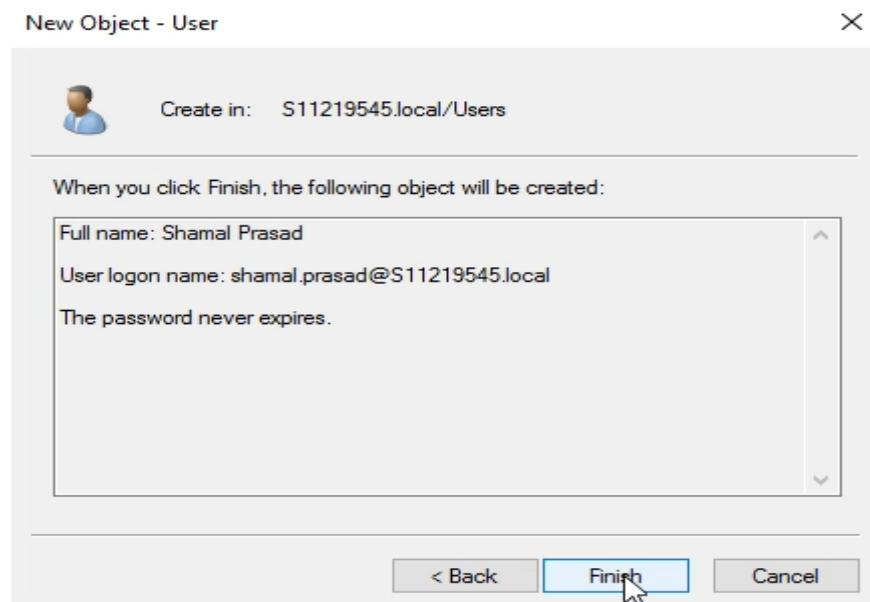




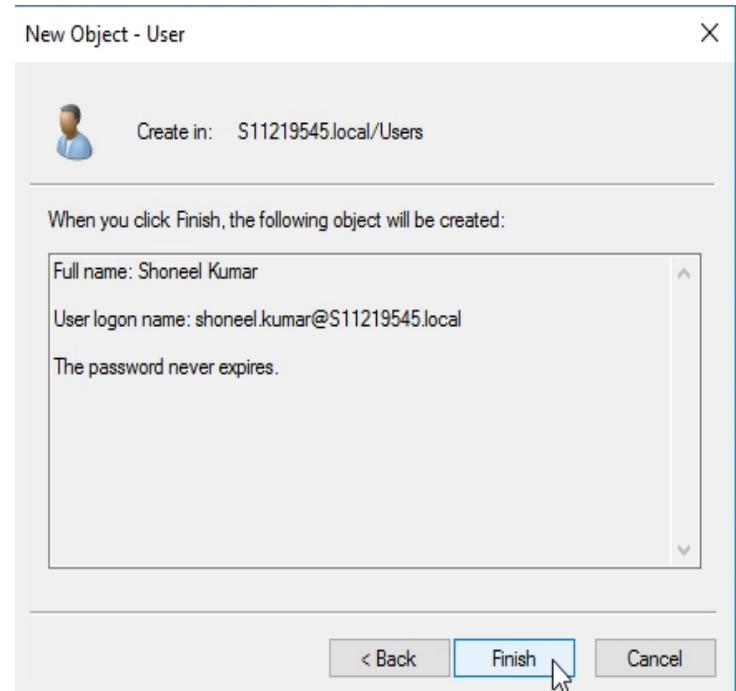
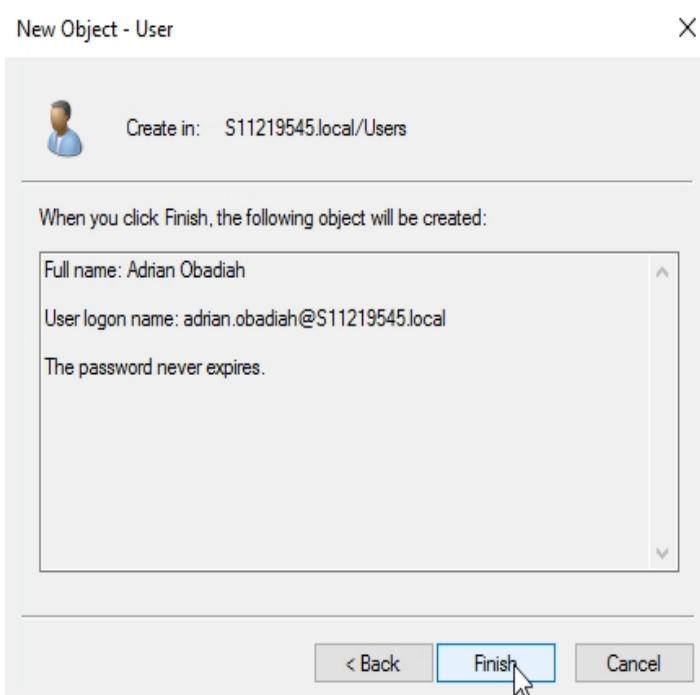
Step 4: Enter the password for your account and change the password policy to password never expires for simplicity sake. Click Next.



Step 5: Review the new user's details to check if everything is in order. If everything is in order then click on Finish to create a user account.

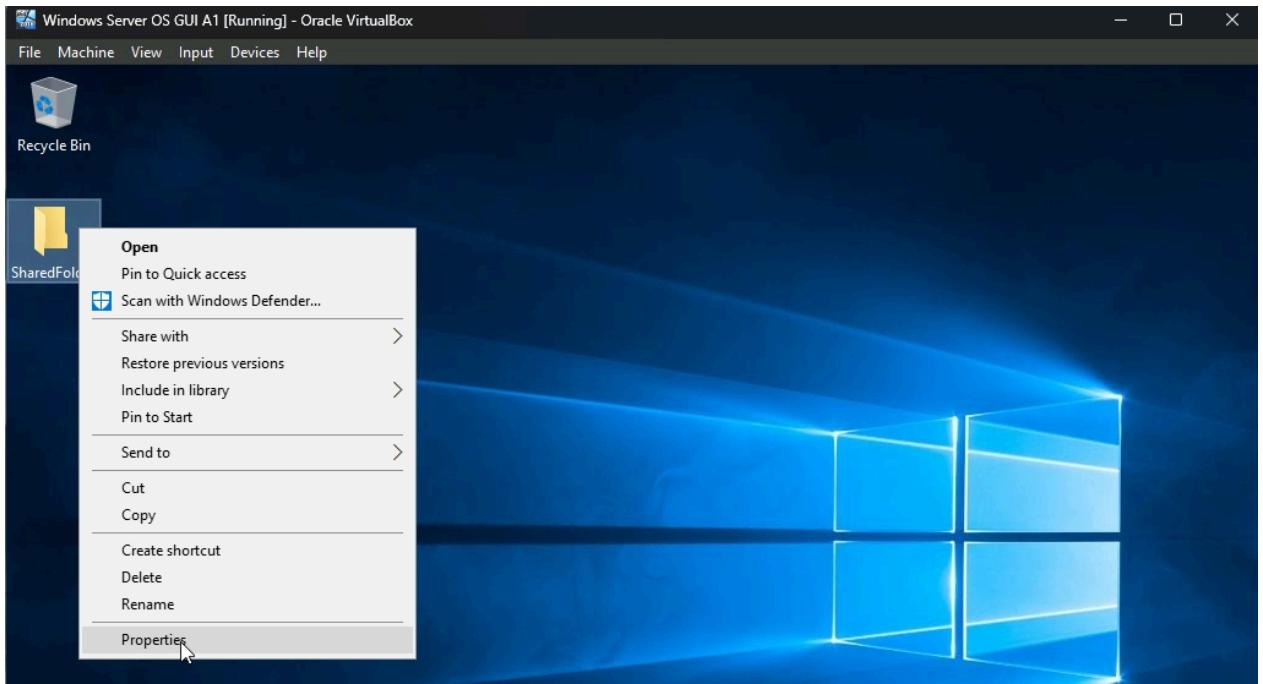


Step 6: Repeat the same process to create additional users.

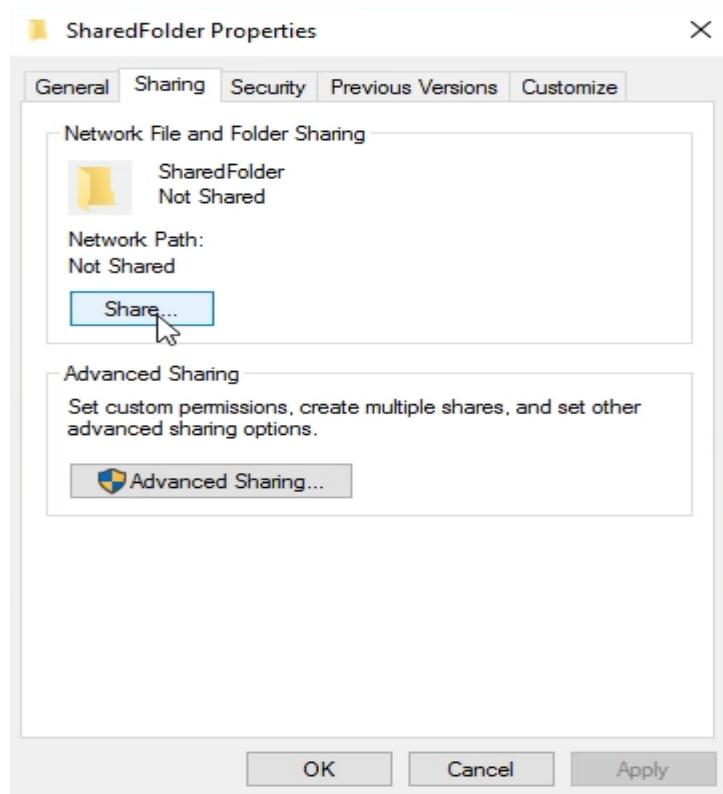


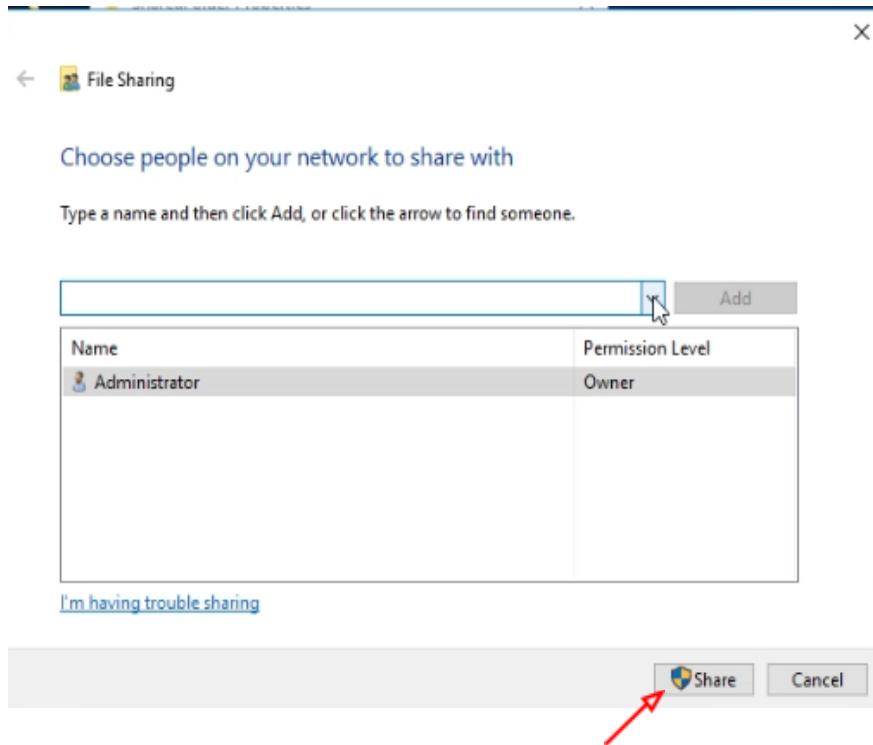
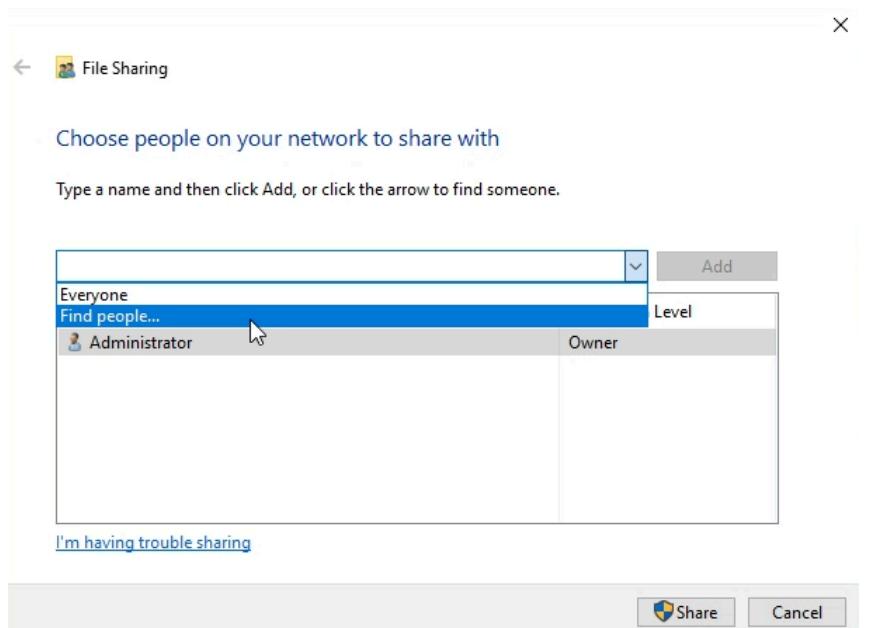
Creating a Shared Folder in GUI Server OS

Step 1: Create a folder in the location that you wish to. Right click on the folder to reveal a menu.

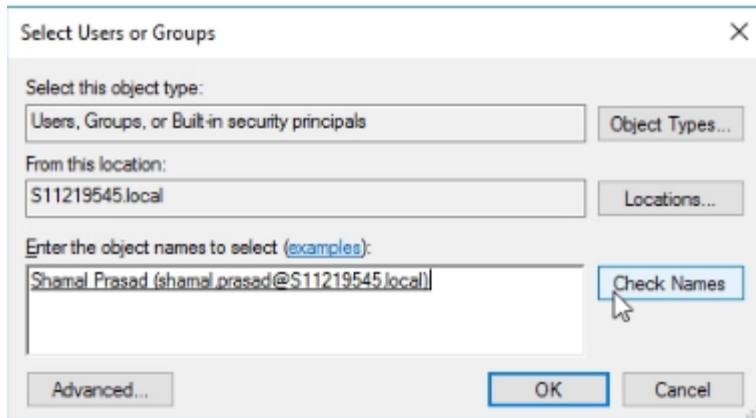


Step 2: Click on the properties of the folder that you wish to share and go to the sharing tab that is located at the top of the properties window.



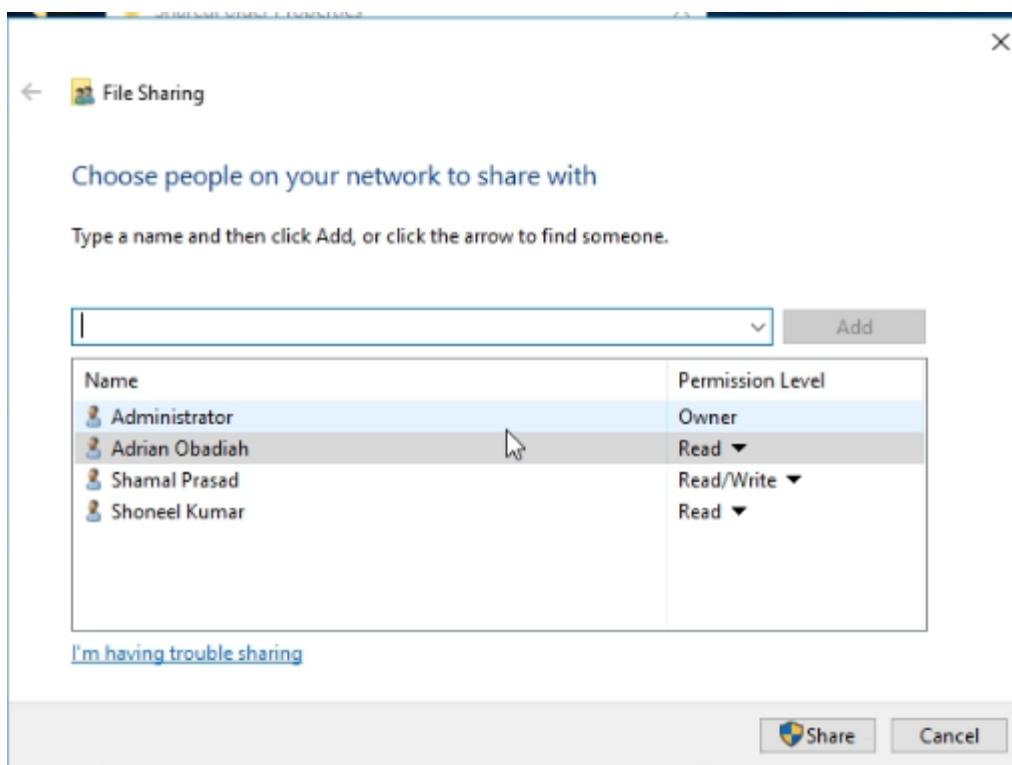
Step 3: Click on “Share”**Step 4:** Expand the dropdown list and select “Find People” to give access to the users that have created previously.

Step 5: Type in the first name of the user of the account and press check name to check if any user accounts exist with the user's name having the one you just typed in. If the desired user account appears, press OK.



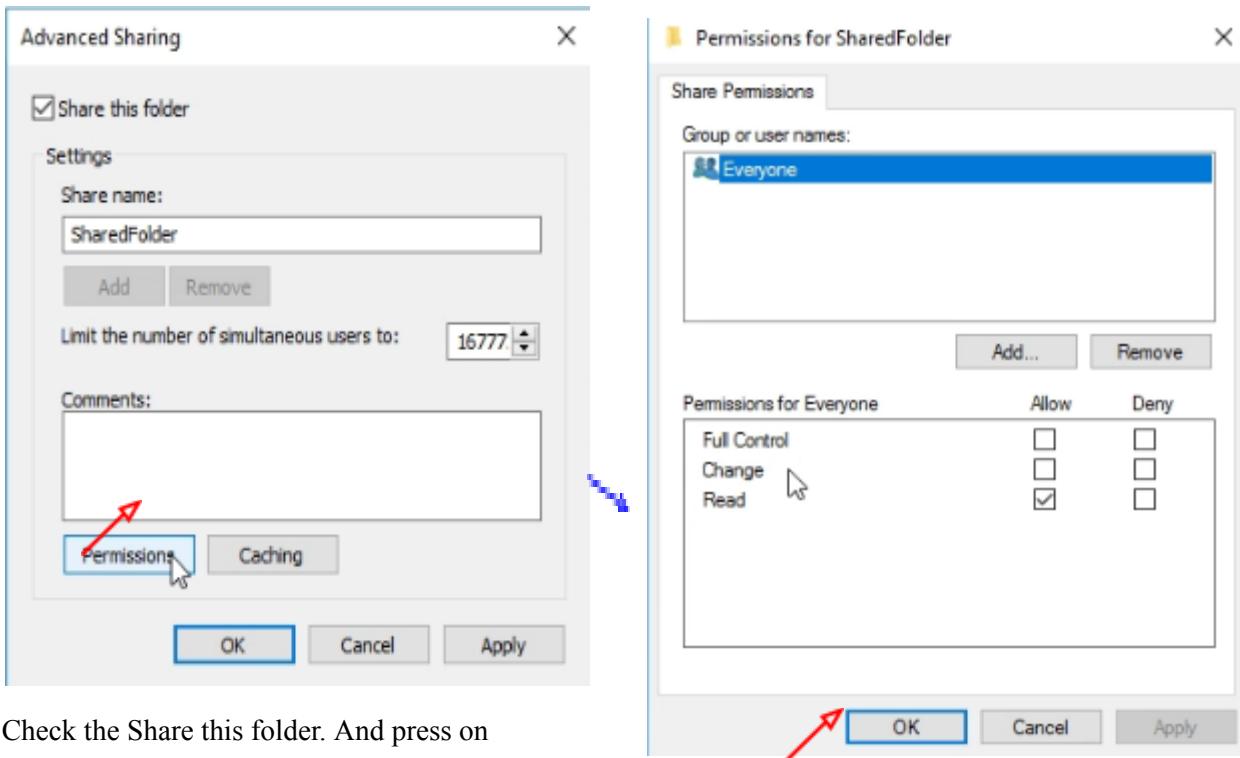
(Do the same for any other users you want to add).

Step 6: Once you have added the users who you want to give access to, can now enforce some rules as to how much permission the user can have in accessing and writing to the shared folder.



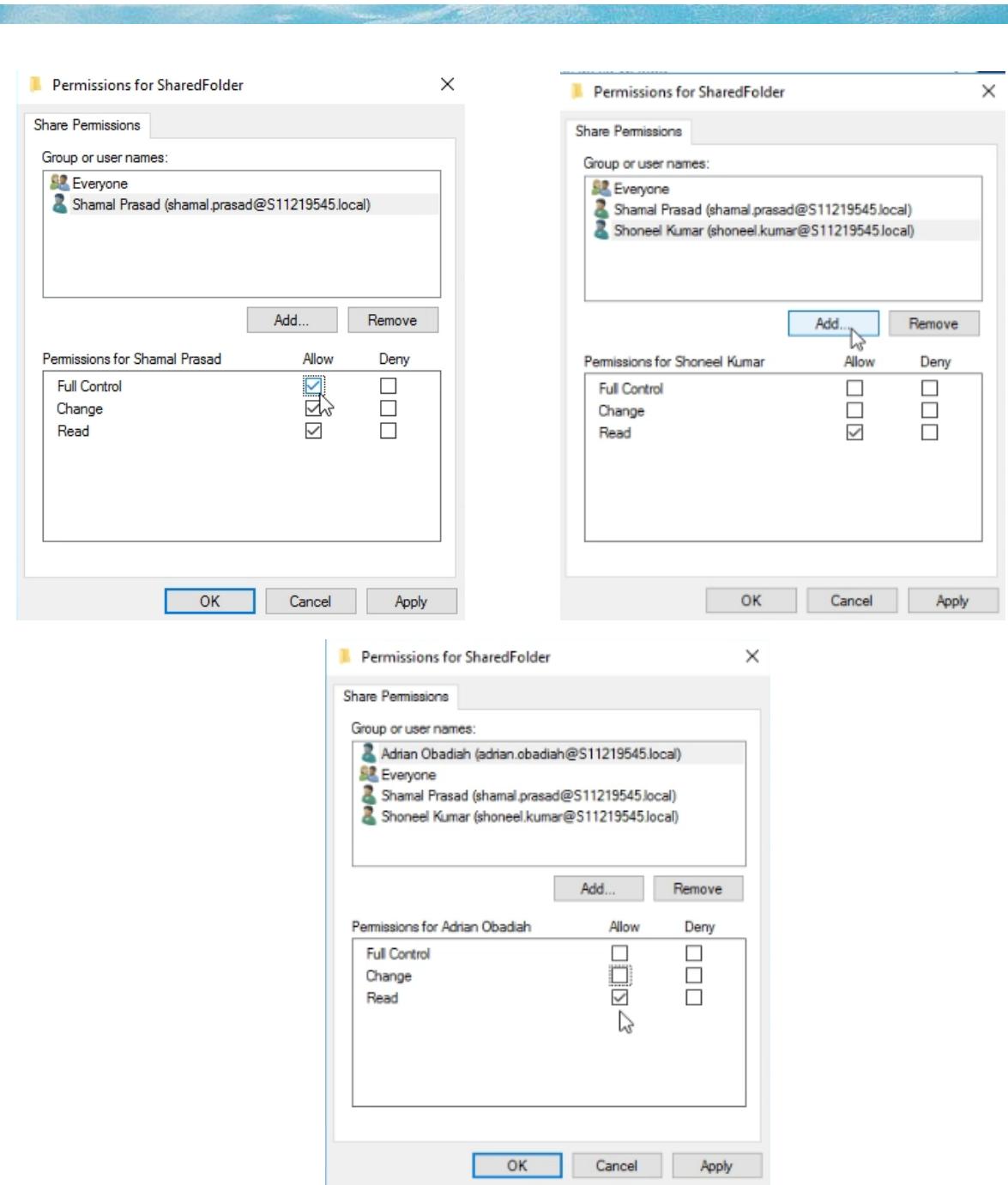
User shamal can read from the folder and create new files and store to it but the remaining users can only view the content of the folder but not put any files into it. You can now press share save the information.

Step 7: Going back to the shared folder properties window, now you must select the “Advanced Sharing...” option and click on it.



Check the Share this folder. And press on Permissions to change the access control to the folder .

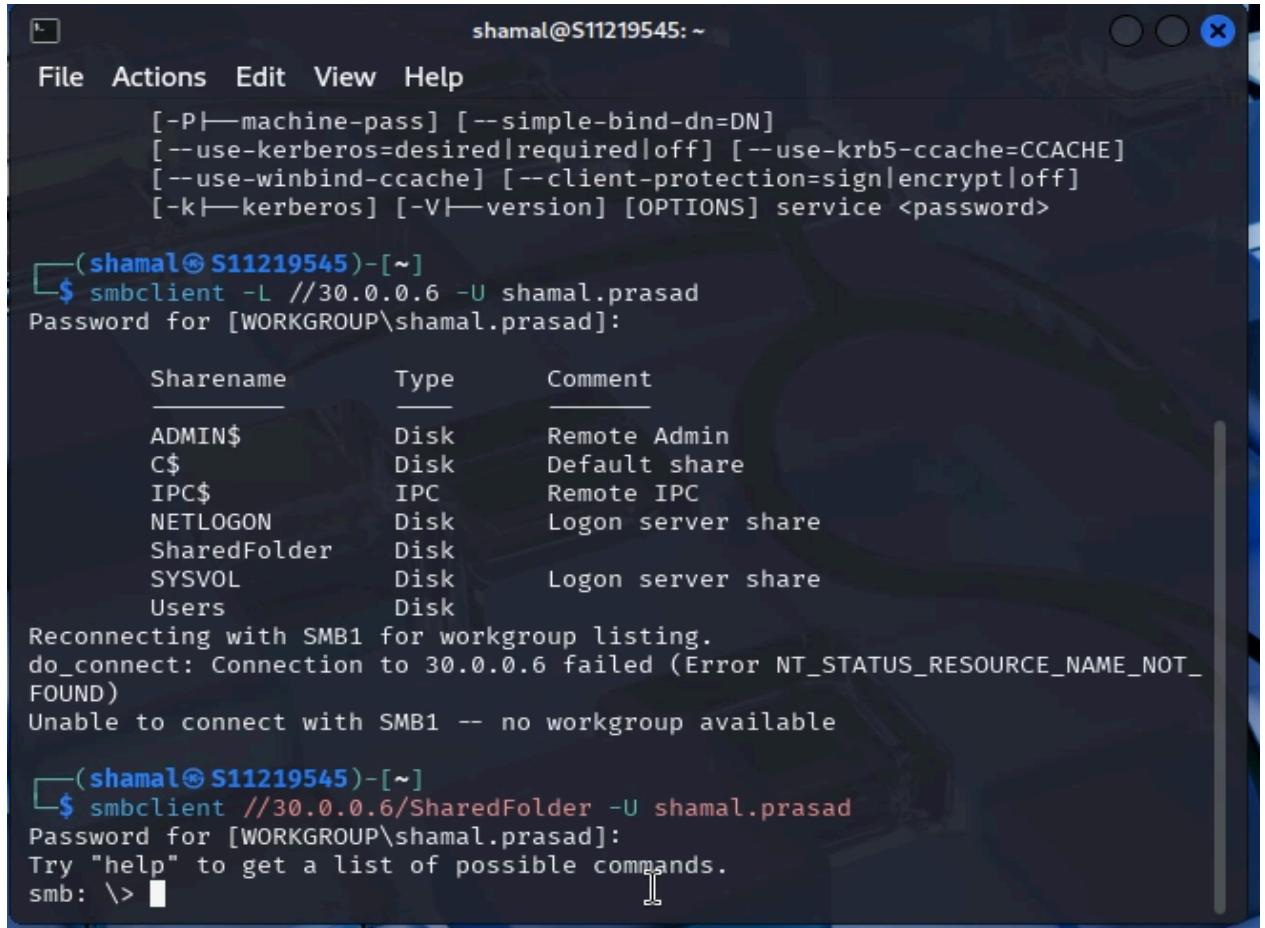
Press the Add button to add users to the folder the same way as you did in the sharing option previously. You can also control the access level of each user of the folder as shown below.



After adding the user and adjusting their access permission to the folder. Press Apply and then Press OK.

Accessing the Shared Folder of Server OS GUI from Kali OS

Step 1: Open the terminal in Kali OS and type in the following command to initiate a communication session with the GUI Server OS. The command is “smbclient //30.0.0.6/SharedFolder -U shamal.prasad”. After executing that command you will need to enter your password for the specific user account.



```

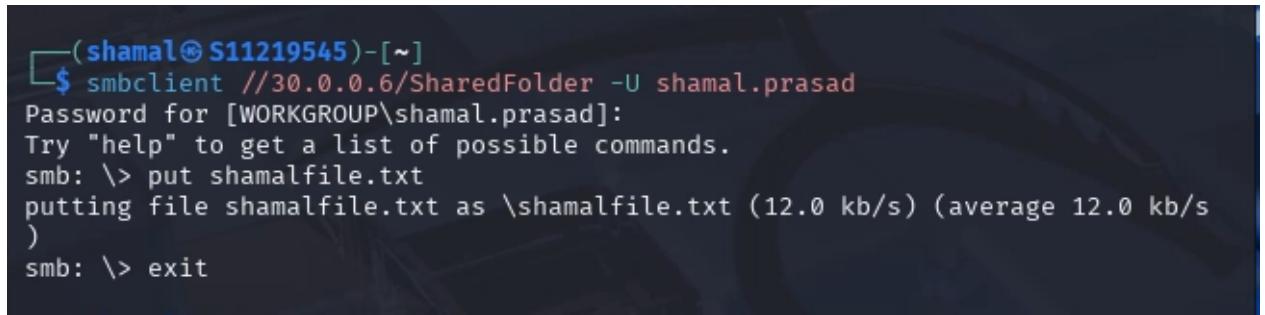
shamal@S11219545: ~
File Actions Edit View Help
[-P]--machine-pass] [--simple-bind-dn=DN]
[--use-kerberos=desired|required|off] [--use-krb5-ccache=CCACHE]
[--use-winbind-ccache] [--client-protection=sign|encrypt|off]
[-k]--kerberos] [-V]--version] [OPTIONS] service <password>

(shamal@S11219545)-[~]
$ smbclient -L //30.0.0.6 -U shamal.prasad
Password for [WORKGROUP\shamal.prasad]:
      Sharename          Type      Comment
      ADMIN$            Disk      Remote Admin
      C$                Disk      Default share
      IPC$              IPC       Remote IPC
      NETLOGON          Disk      Logon server share
      SharedFolder       Disk      Logon server share
      SYSVOL            Disk      Logon server share
      Users              Disk
Reconnecting with SMB1 for workgroup listing.
do_connect: Connection to 30.0.0.6 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)
Unable to connect with SMB1 -- no workgroup available

(shamal@S11219545)-[~]
$ smbclient //30.0.0.6/SharedFolder -U shamal.prasad
Password for [WORKGROUP\shamal.prasad]:
Try "help" to get a list of possible commands.
smb: \> █

```

Step 2: User Shamal Prasad has access to read and write to the shared folder.



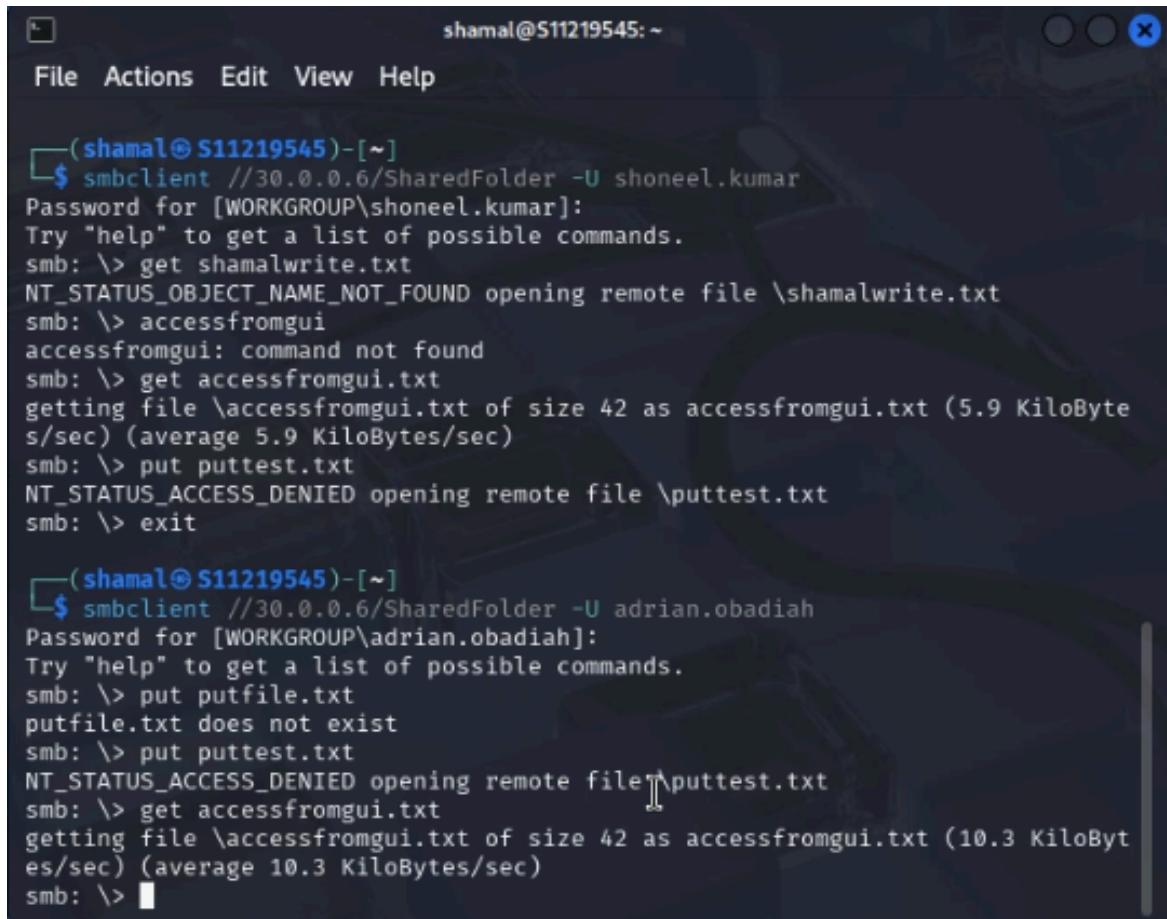
```

(shamal@S11219545)-[~]
$ smbclient //30.0.0.6/SharedFolder -U shamal.prasad
Password for [WORKGROUP\shamal.prasad]:
Try "help" to get a list of possible commands.
smb: \> put shamalfie.txt
putting file shamalfie.txt as \shamalfie.txt (12.0 kb/s) (average 12.0 kb/s)
)
smb: \> exit

```

In this snippet the user can put the file into the folder.

For a user like Shoneel Kumar or Adrian Obadiah who only have read only access, they can get files from the folder but not put the files in the folder. This can be seen in the snippet below as when the users try to put a file into the folder, an error pops up and prevents them from writing to the folder.



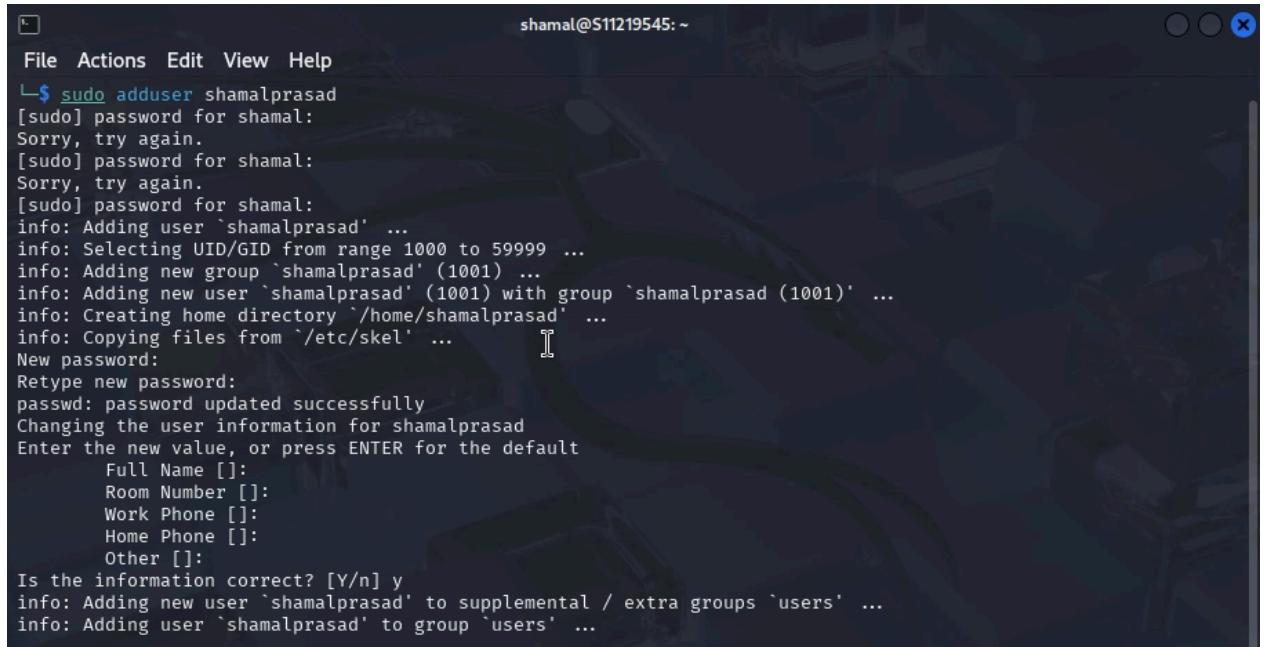
The screenshot shows a terminal window titled "shamal@S11219545: ~". The window contains two sessions of the smbclient command-line interface:

```
(shamal@S11219545)-[~]
$ smbclient //30.0.0.6/SharedFolder -U shoneel.kumar
Password for [WORKGROUP\shoneel.kumar]:
Try "help" to get a list of possible commands.
smb: \> get shamalwrite.txt
NT_STATUS_OBJECT_NAME_NOT_FOUND opening remote file \shamalwrite.txt
smb: \> accessfromgui
accessfromgui: command not found
smb: \> get accessfromgui.txt
getting file \accessfromgui.txt of size 42 as accessfromgui.txt (5.9 KiloByte
s/sec) (average 5.9 KiloBytes/sec)
smb: \> put puttest.txt
NT_STATUS_ACCESS_DENIED opening remote file \puttest.txt
smb: \> exit

(shamal@S11219545)-[~]
$ smbclient //30.0.0.6/SharedFolder -U adrian.obadiah
Password for [WORKGROUP\adrian.obadiah]:
Try "help" to get a list of possible commands.
smb: \> put putfile.txt
putfile.txt does not exist
smb: \> put puttest.txt
NT_STATUS_ACCESS_DENIED opening remote file \puttest.txt
smb: \> get accessfromgui.txt
getting file \accessfromgui.txt of size 42 as accessfromgui.txt (10.3 KiloByte
es/sec) (average 10.3 KiloBytes/sec)
smb: \> █
```

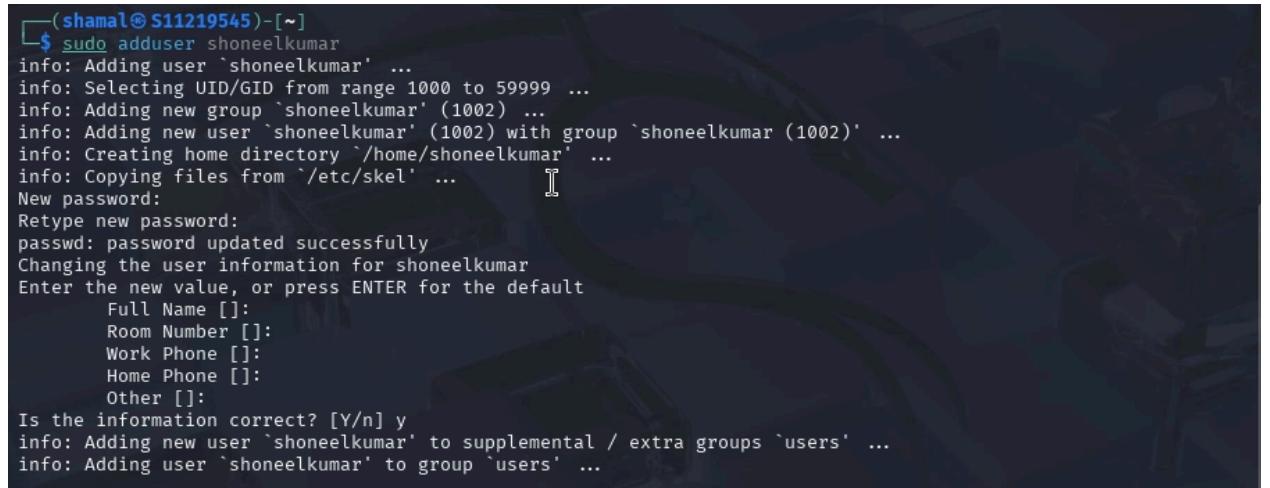
Creating Users, Shared Folder In Kali OS and Accessing from GUI Server OS

Step 1: To create a user on a Kali Os, you need to run the following command “sudo adduser username”. After that you will have to create a password for the new user and fill in the details about the user, but entering the details about the user is not important so it can be left blank and updated later. The main things are the username and password.



```
shamal@S11219545:~$ sudo adduser shamalprasad
[sudo] password for shamal:
Sorry, try again.
[sudo] password for shamal:
Sorry, try again.
[sudo] password for shamal:
info: Adding user `shamalprasad' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `shamalprasad' (1001) ...
info: Adding new user `shamalprasad' (1001) with group `shamalprasad (1001)' ...
info: Creating home directory `/home/shamalprasad' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for shamalprasad
Enter the new value, or press ENTER for the default
      Full Name []:
      Room Number []:
      Work Phone []:
      Home Phone []:
      Other []:
Is the information correct? [Y/n] y
info: Adding new user `shamalprasad' to supplemental / extra groups `users' ...
info: Adding user `shamalprasad' to group `users' ...
```

You can now run the command again for creating the next user



```
(shamal@S11219545)-[~]
$ sudo adduser shoneelkumar
info: Adding user `shoneelkumar' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `shoneelkumar' (1002) ...
info: Adding new user `shoneelkumar' (1002) with group `shoneelkumar (1002)' ...
info: Creating home directory `/home/shoneelkumar' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for shoneelkumar
Enter the new value, or press ENTER for the default
      Full Name []:
      Room Number []:
      Work Phone []:
      Home Phone []:
      Other []:
Is the information correct? [Y/n] y
info: Adding new user `shoneelkumar' to supplemental / extra groups `users' ...
info: Adding user `shoneelkumar' to group `users' ...
```

```
(shamal@S11219545) -[~]
$ sudo adduser adrianobadiah
info: Adding user `adrianobadiah' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `adrianobadiah' (1003) ...
info: Adding new user `adrianobadiah' (1003) with group `adrianobadiah (1003)' ...
info: Creating home directory `/home/adrianobadiah' ...
info: Copying files from `/etc/skel' ...
New password: [REDACTED]
Retype new password: [REDACTED]
Sorry, passwords do not match.
passwd: Authentication token manipulation error
passwd: password unchanged
Try again? [y/N] y
New password: [REDACTED]
Retype new password: [REDACTED]
passwd: password updated successfully
Changing the user information for adrianobadiah
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] [REDACTED]
```

Step 2: Moving on, we need to now create a folder that can be shared between each newly created user. To create a folder run “sudo mkdir -p /srv/samba/SharedFolder”, “sudo chown root:root /srv/samba/SharedFolder” and “sudo chmod 2770 /srv/samba/SharedFolder” as shown in the snippet below.

```
(shamal@S11219545) -[~]
$ sudo mkdir -p /srv/samba/SharedFolder

(shamal@S11219545) -[~]
$ sudo chown root:root /srv/samba/SharedFolder

(shamal@S11219545) -[~]
$ sudo chmod 2770 /srv/samba/SharedFolder
```

Step 3: For the user you have just created, you must also create a Samba account for them so they can use its functions. Type in the command “sudo smbpasswd -a username” and you will be prompted to enter a password for the user account. Your Samba account password and your actual account password do not need to be the same but simplicity can make it same.

```
(shamal@S11219545) [~]
$ sudo smbpasswd -a shamalprasad
New SMB password:
Retype new SMB password:
Added user shamalprasad.

(shamal@S11219545) [~]
$ sudo smbpasswd -a shoneelkumar
New SMB password:
Retype new SMB password:
Added user shoneelkumar.

(shamal@S11219545) [~]
$ sudo smbpasswd -a adrianobadiyah
New SMB password:
Retype new SMB password:
Added user adrianobadiyah.
```

Step 4: After creating Samba accounts, we can now configure the folder by sharing it and defining the permission for each user. Run the command “sudo nano /etc/samba/smb.conf” to open a file where you will be adding the folder sharing rules. When you are in the file can define the rule similar to the one in the snippet below.

```
[SharedDocs]
path = /srv/samba/SharedFolder
browseable = yes
read only = no
guest ok = no
valid users = shamalprasad, shoneelkumar, adrianobadiyah
write list = shamalprasad
read list = shoneelkumar, adrianobadiyah
```

You can then save the changes made to the file and exit out of the file.

There is also an ACL option to define permission and it appears as shown below.

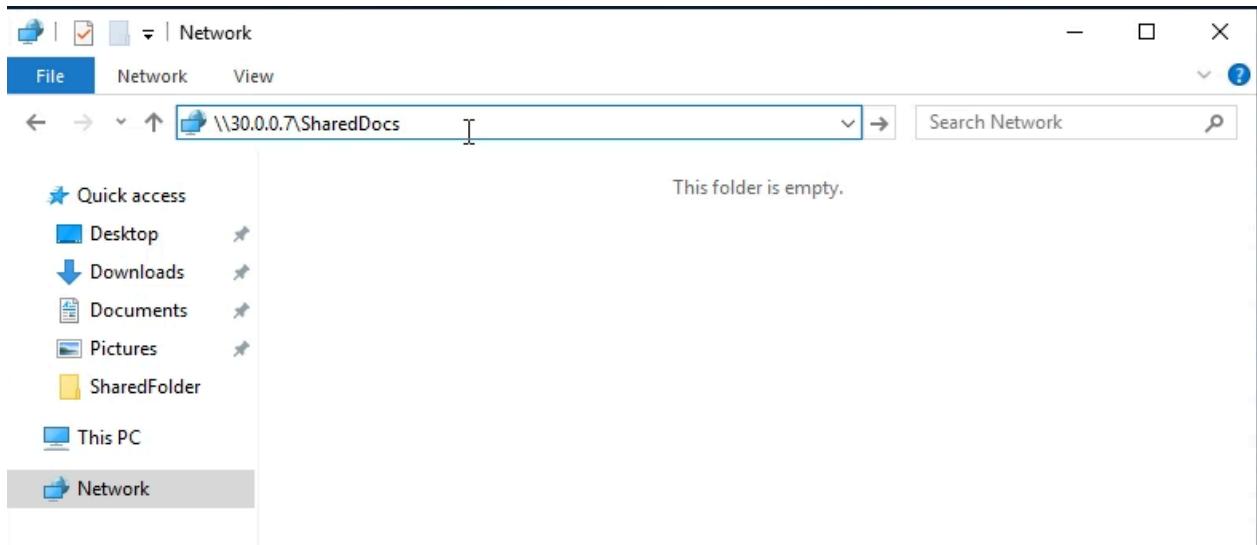
```
(shamal@S11219545) [~]
$ sudo setfacl -m u:shoneelkumar:rx /srv/samba/SharedFolder

(shamal@S11219545) [~]
$ sudo setfacl -m u:adrianobadiyah:rx /srv/samba/SharedFolder

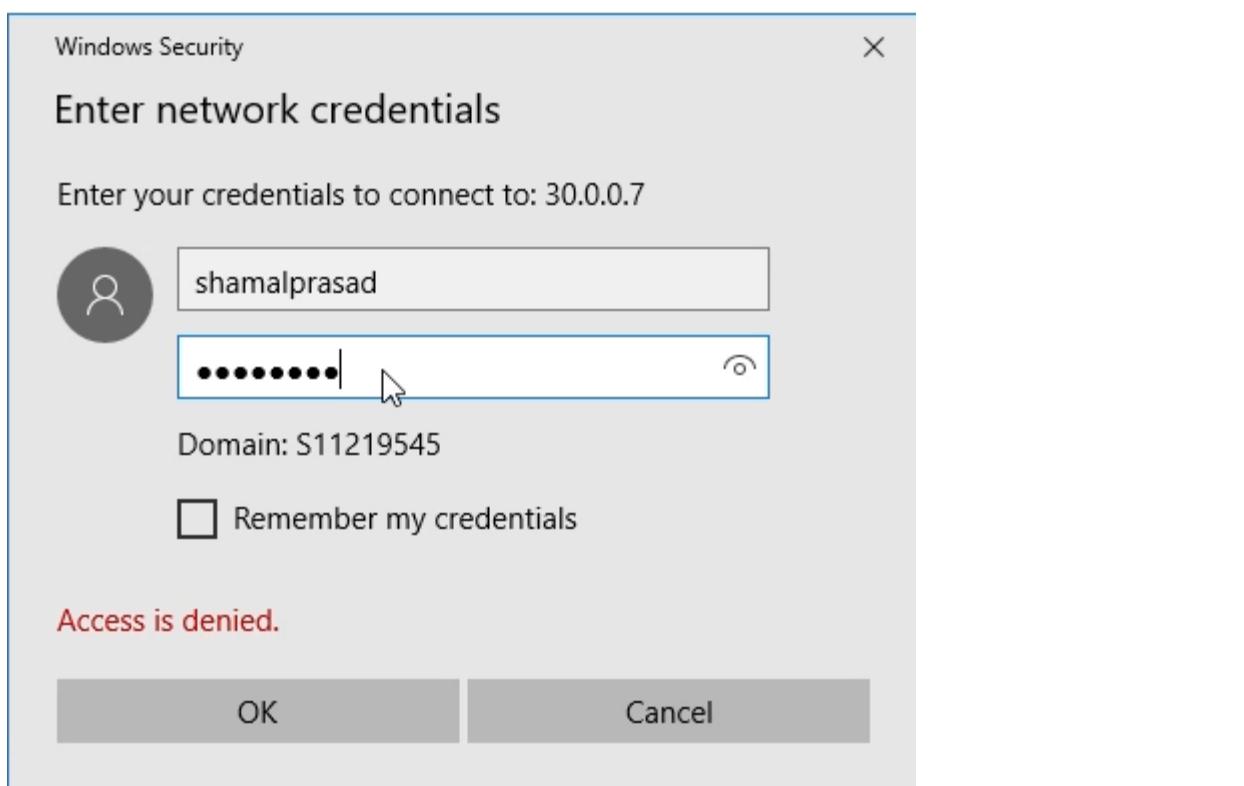
(shamal@S11219545) [~]
$ sudo setfacl -m u:shamalprasad:rwx /srv/samba/SharedFolder
```

After that you will need to restart Samba by running the command “sudo systemctl restart smbd”.

Step 3: Accessing the shared folder from the Server OS. In the file explorer, look for the network tab and in that tab, type out the server IP address and then the shared folder name in the pathway address bar.

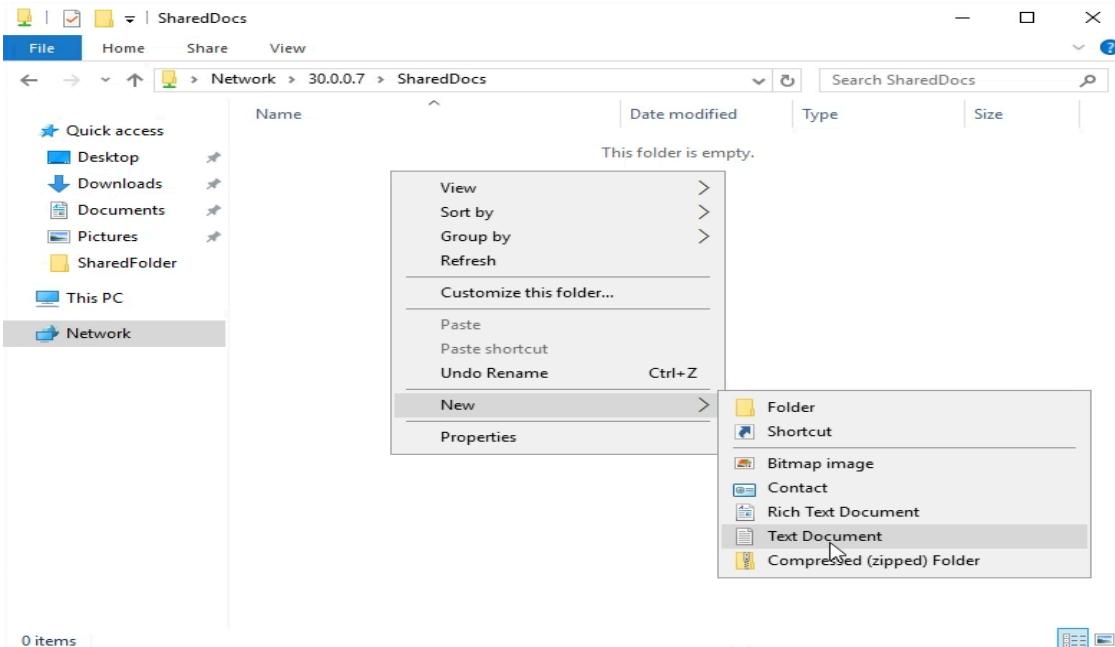


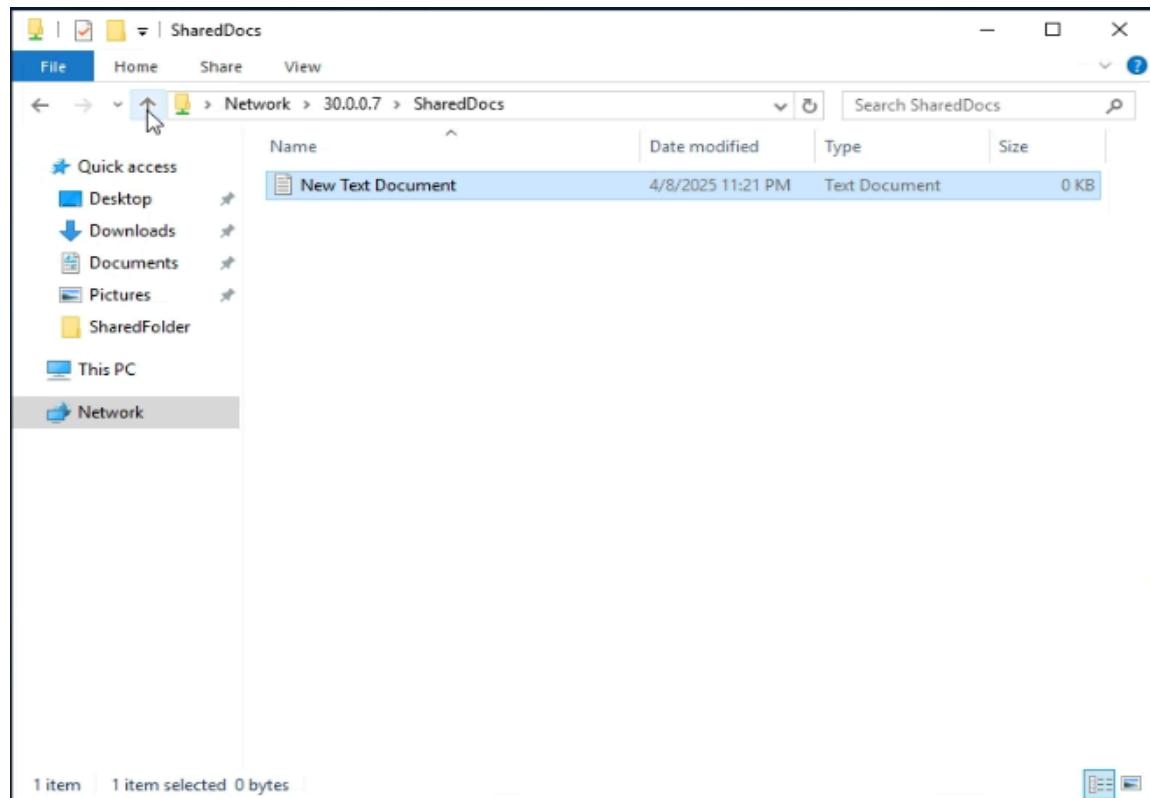
Once you press enter, there will be a prompt asking you to login before accessing the contents of the folder.



Step 4: Different accounts having varying levels of permission. The shamalprasad account has both read and write access.

Shamalprasad can create a new file and add it to the shared folder as well as view the file in the folder.

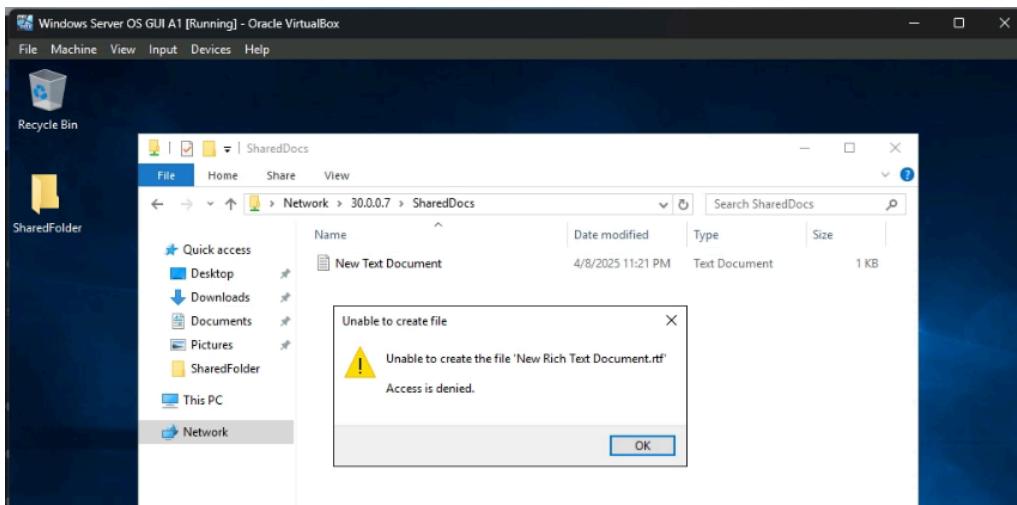
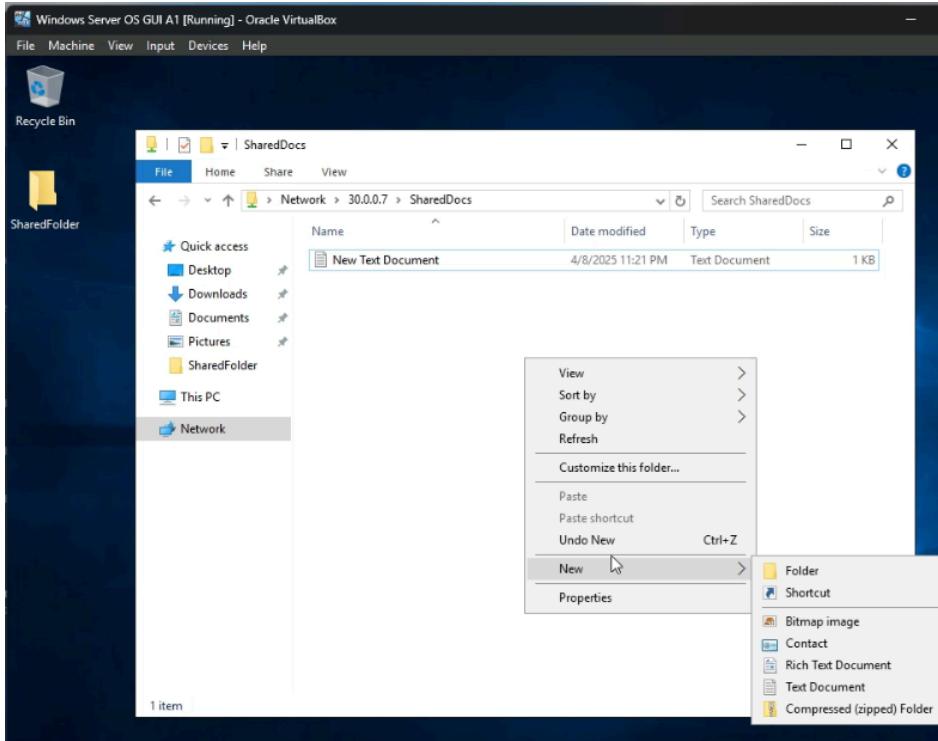




However, users like shoneelkumar and adrianobadiah do not have write access but only read access and can view the content.

The snippets above show that user shoneelkumar can view the contents of the file in the shared folder.

But when trying to write to the folder, the access is denied.



The same results are followed if we try performing read and write actions using adrianobadiah's account since both users adrianobadiah and shoneelkumar have the same privilege .

Creating Users, Shared Folder In CLI Server OS and Accessing from Kali OS

Step 1: To create users in CLI Server OS, we need to run the following command in the cmd, “net user username password /add”. The username and password are all in one command.

```
C:\Users\Administrator>net user shamal Pass.1234 /add
The command completed successfully.

C:\Users\Administrator>net user shoneel Pass.1234 /add
The command completed successfully.

C:\Users\Administrator>net user adrian Pass.1234 /add
The command completed successfully.
```

Step 2: After creating the users, we now need to create the folder, set its permission and share it. Run the command “mkdir Shared” to make a directory. After that, set the NFTS permission for each user and use net share command to allow other OS’s to see the folder as a shared folder. The commands for all 3 functions are shown below.

```
Administrator: C:\Windows\system32\cmd.exe
04/09/2025 07:39 PM <DIR> SharedDocs
04/09/2025 08:53 PM <DIR> SharedFolder
04/09/2025 09:09 AM <DIR> Videos
    0 File(s)      0 bytes
  15 Dir(s) 45,794,934,784 bytes free

C:\Users\Administrator>mkdir Shared

C:\Users\Administrator>icacls "C:\Users\Administrator\Shared" /grant shamal:(OI)(CI)F
processed file: C:\Users\Administrator\Shared
Successfully processed 1 files; Failed processing 0 files

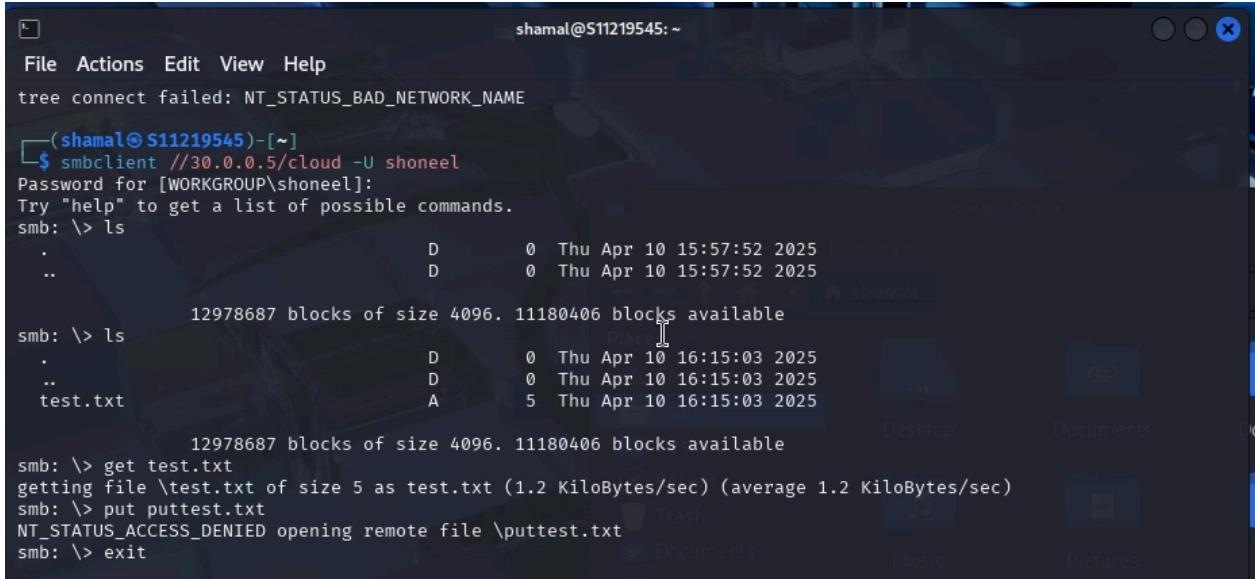
C:\Users\Administrator>icacls "C:\Users\Administrator\Shared" /grant shoneel:(OI)(CI)R
processed file: C:\Users\Administrator\Shared
Successfully processed 1 files; Failed processing 0 files

C:\Users\Administrator>icacls "C:\Users\Administrator\Shared" /grant adrian:(OI)(CI)R
processed file: C:\Users\Administrator\Shared
Successfully processed 1 files; Failed processing 0 files

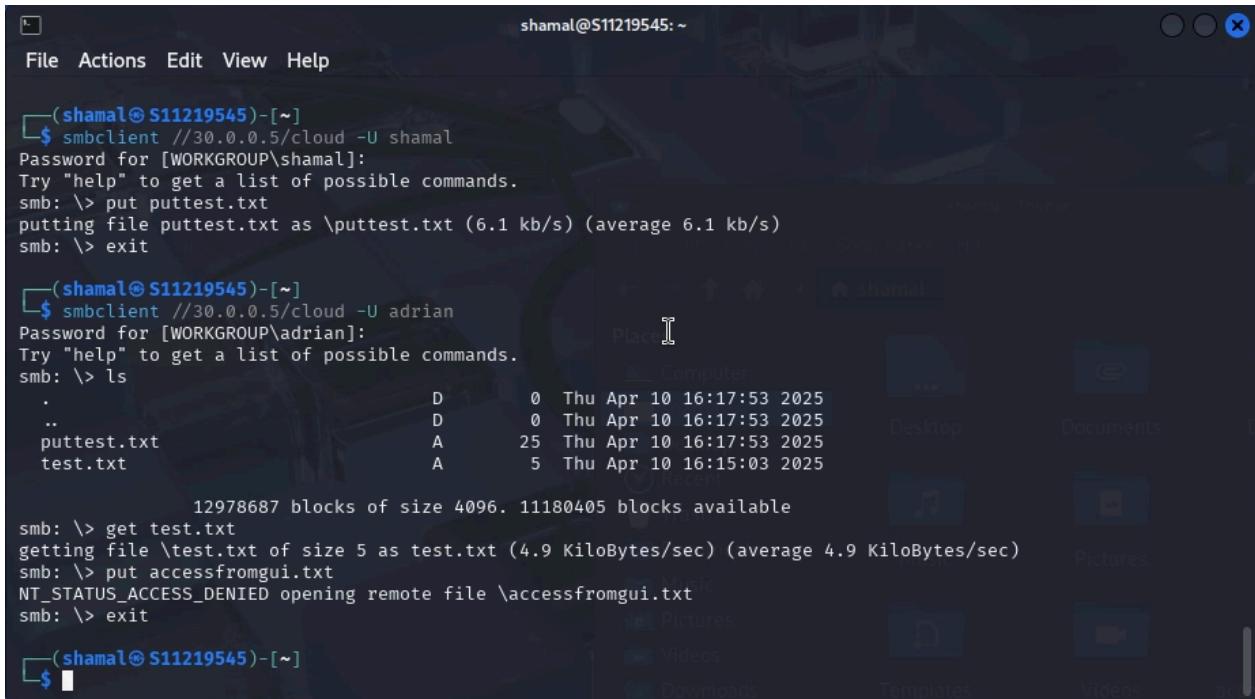
C:\Users\Administrator>net share cloud="C:\Users\Administrator\Shared" /GRANT:shamal,FULL /GRANT:shoneel,READ /GRANT:adrian,READ
cloud was shared successfully.

C:\Users\Administrator>cd shared
C:\Users\Administrator\Shared>echo hi > test.txt
C:\Users\Administrator\Shared>dir
```

Step 3: After creating the shared folder and sharing it to the users while assigning restriction of access to the folder. You can now use smbclient in Kali OS to access the shared folder. In our case user shamal has full read and write access while the remaining two users only have read access and cannot upload files to the shared folder.



```
shamal@S11219545: ~
File Actions Edit View Help
tree connect failed: NT_STATUS_BAD_NETWORK_NAME
(shamal@S11219545)-[~]
$ smbclient //30.0.0.5/cloud -U shoneel
Password for [WORKGROUP\shoneel]:
Try "help" to get a list of possible commands.
smb: \> ls
.
..
D 0 Thu Apr 10 15:57:52 2025
D 0 Thu Apr 10 15:57:52 2025
12978687 blocks of size 4096. 11180406 blocks available
smb: \> ls
.
..
D 0 Thu Apr 10 16:15:03 2025
D 0 Thu Apr 10 16:15:03 2025
test.txt A 5 Thu Apr 10 16:15:03 2025
12978687 blocks of size 4096. 11180406 blocks available
smb: \> get test.txt
getting file \test.txt of size 5 as test.txt (1.2 Kilobytes/sec) (average 1.2 Kilobytes/sec)
smb: \> put puttest.txt
NT_STATUS_ACCESS_DENIED opening remote file \puttest.txt
smb: \> exit
```



```
shamal@S11219545: ~
File Actions Edit View Help
(shamal@S11219545)-[~]
$ smbclient //30.0.0.5/cloud -U shamal
Password for [WORKGROUP\shamal]:
Try "help" to get a list of possible commands.
smb: \> put puttest.txt
putting file puttest.txt as \puttest.txt (6.1 kb/s) (average 6.1 kb/s)
smb: \> exit

(shamal@S11219545)-[~]
$ smbclient //30.0.0.5/cloud -U adrian
Password for [WORKGROUP\adrian]:
Try "help" to get a list of possible commands.
smb: \> ls
.
..
D 0 Thu Apr 10 16:17:53 2025
D 0 Thu Apr 10 16:17:53 2025
puttest.txt A 25 Thu Apr 10 16:17:53 2025
test.txt A 5 Thu Apr 10 16:15:03 2025
12978687 blocks of size 4096. 11180405 blocks available
smb: \> get test.txt
getting file \test.txt of size 5 as test.txt (4.9 Kilobytes/sec) (average 4.9 Kilobytes/sec)
smb: \> put accessfromgui.txt
NT_STATUS_ACCESS_DENIED opening remote file \accessfromgui.txt
smb: \> exit

(shamal@S11219545)-[~]
$
```

Based on the snippets shown above, user shamal can put files into the shared folder but the other 2 users cannot as they receive an error stating that the access has been denied. However, users adrian and shoneel can both get the files from the shared folder and view it on their end.

Part 2: Kali LInux Versus Windows Server OS.

Before we indulge into any comparisons between the 2 operating systems (Kali Linux windows server OS) it's important to understand why they are built differently. The answer lies in the fact that both were built for different audiences and primary uses. With Kali targeting security professionals and ethical hackers and Windows server OS more towards the everyday users that are non-techy (Parker, 2024).

Security Features

Below are a few example comparisons between the two OSes.

Example 1: Windows firewall.

In Windows, by default, some settings are disabled for security reasons. For instance, when we tried to ping windows server from Kali Linux's side initially, it gave an error saying, host unreachable. Meaning by default, the settings for remote configurations were disabled. In this case, the ICMP protocol - which is responsible for letting other remote users in the network to determine if a server is reachable.

From CLI windows this is done by entering sconfig, choose option 4, then option 3.

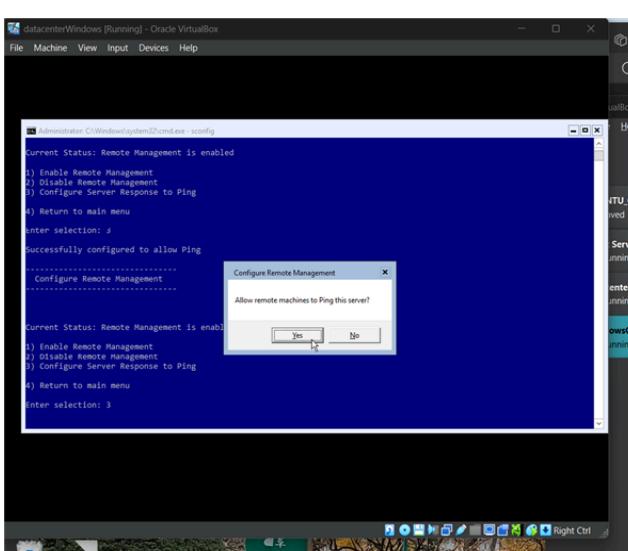


Figure 2.0 enables ICMP protocol in CLI OS.

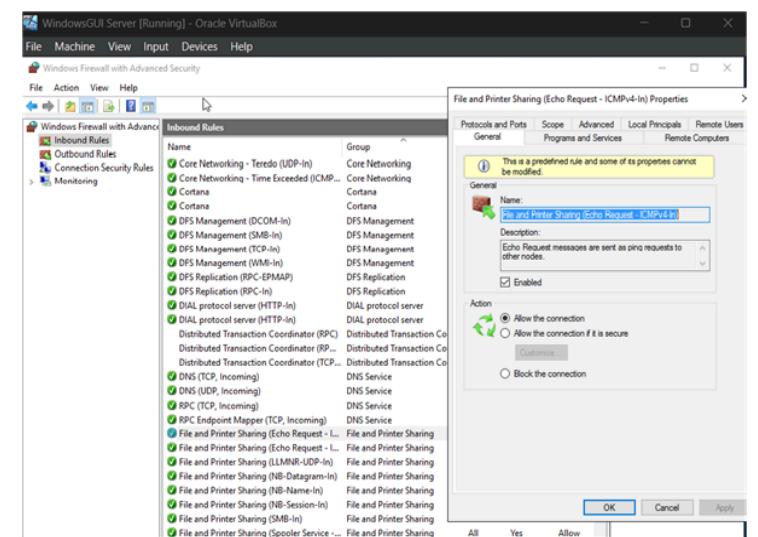


figure 2.1 enable ICMP in GUI OS

The same is done for the GUI windows server but through the firewall inbound settings, enabling the File and Printer Sharing from the firewall setting.

Kali Linux OS doesn't, by default, does not have that security feature disabled so you can directly ping and send echo request messages and get responses. So in this case windows is much more secure.

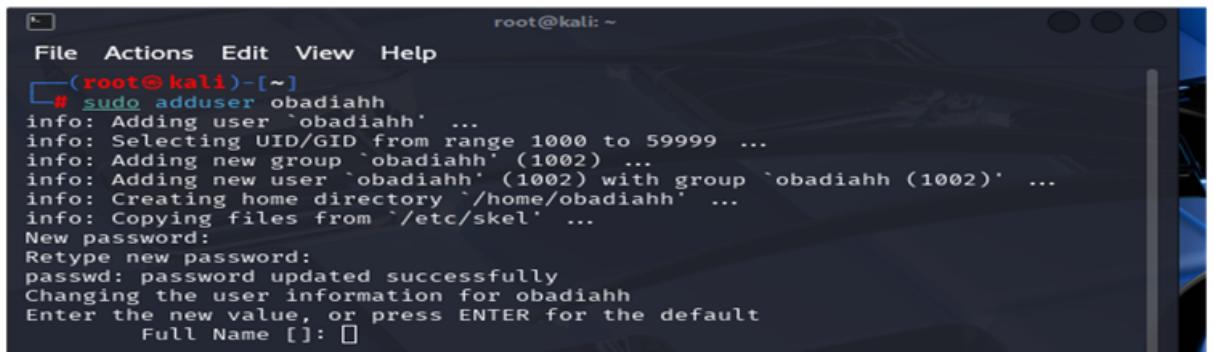
Example 2: Password.

Windows has strict criteria that need to be met before any password is set. This security feature ensures that the user's password is not too simple and can be easily hacked. For Kali on the other hand, passwords can be set without any strict requirements. For example, *loser3es* is acceptable in Kali Linux while in Windows this will give an error, as it must meet the password requirements (at least 8 characters, at least one Uppercase letter, and at least a number). Thus, we collectively agreed that for any ordinary user, it's more secure to use windows than kali.

User Management And Access Control

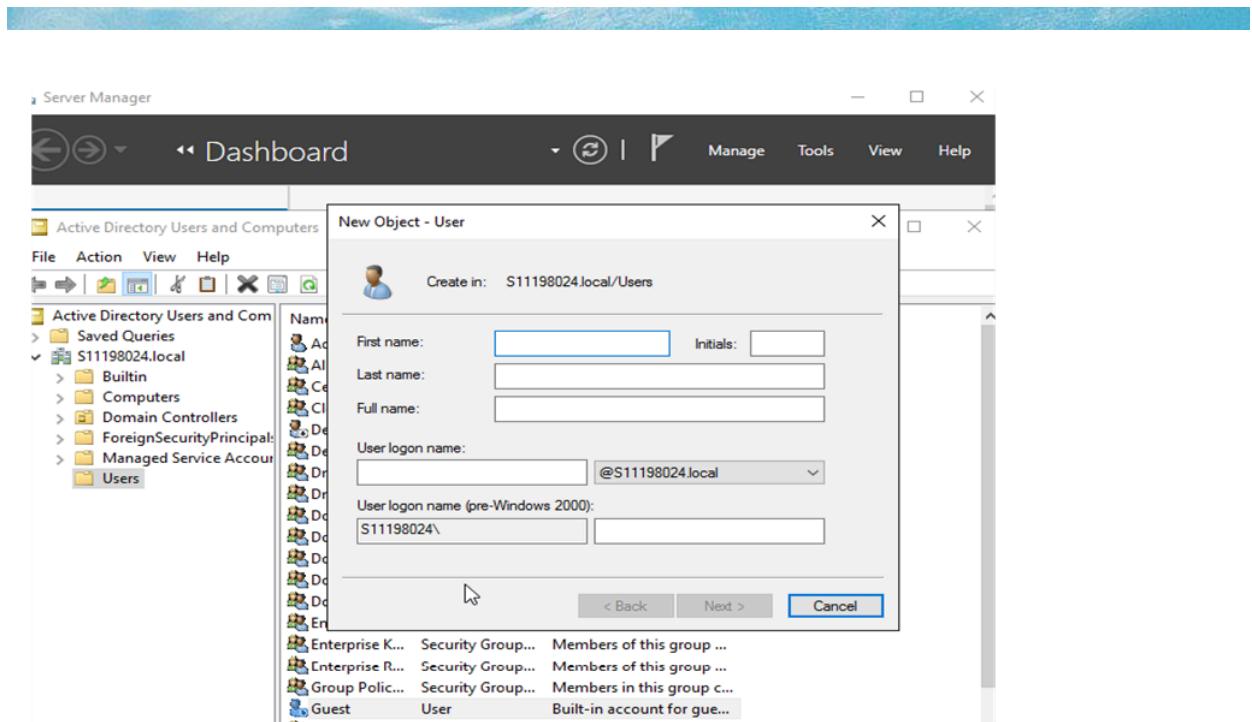
- Kali OS - Allows you to create users (by the root user only), and manage how much control and access they have to other resources of the system. The users created can then be assigned into a group with specific permissions to access and manipulate shared resources.

However, It would take some research from the user's end to be able to use and navigate the environment overall as most functions are done through the terminal which means knowing the commands which new users can find a bit daunting.



```
root@kali: ~
File Actions Edit View Help
[(root@kali)-[~]
# sudo adduser obadiah
info: Adding user `obadiah' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `obadiah' (1002) ...
info: Adding new user `obadiah' (1002) with group `obadiah (1002)' ...
info: Creating home directory `/home/obadiah' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for obadiah
Enter the new value, or press ENTER for the default
Full Name []: 
```

- Windows OS – windows use a graphical user interface, with tools like active directory and Group policy for user management. And to create users, only the admin has the privilege to do so and determine their type.



Design and User Interface

In terms of the design and the user interface, Kali Linux certainly draws attention to its impressive user interface which not only focuses on functionality but also emphasizes simplicity.

It's designed in a manner that supports typical cybersecurity tasks of cybersecurity professionals with an accessible and minimalistic layout through an integrated command center via the terminal. However, new users may find using the terminal a bit daunting to use hence, compared to Windows OS it is less user-friendly to use.

Windows Server OS on the other hand offers a very intuitive and very user-friendly design that's easy to navigate. By providing a visually consistent layout it allows users to perform tasks effortlessly. The consistent layout of the environment across different updates among windows products simplifies complex server management tasks that are done in the prior section. The environment includes the start menu, taskbar at the bottom and desktop icons.

Part B: Payroll System

Code: Below is the code for the payroll system.

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <pthread.h>
4 #include <string.h>
5 #include <stdbool.h>
6
7 //constant declaration
8 #define NUM_OF_EMPLOYEES 5
9 #define MAX_NORMAL_HOURS 40
10 #define OVERTIME_RATE 1.5
11 #define SUPERANNUATION_RATE 0.08
12 #define WEEKS_PER_YEAR 52
13 #define HOURS_PER_YEAR (MAX_NORMAL_HOURS * WEEKS_PER_YEAR)// max normal hours per week * weeks per year
14 #define MINIMUM_WORKING_HOURS 1
15 #define MAXIMUM_WORKING_HOURS 168
16
17 //employee structure to capture all its info
18 typedef struct {
19     char employee_id[10];
20     char name[50];
21     double annual_salary;
22     double hourly_rate;
23     int hours_worked;
24     double gross_pay;
25     double net_pay;
26     double superannuation_deduction;
27 } Employee;
28
29
30
31 int find_employee_index(char *id, Employee *employees, int num_employees) {
32     for (int i = 0; i < num_employees; i++) {
33         if (strcmp(employees[i].employee_id, id) == 0) {
34             return i;
35         }
36     }
37     return -1;
38 }
39
40 // function to calculate payroll
41 void *calculate_payroll(void *arg) {
42     Employee *emp = (Employee *)arg;
43
44     // calculate hourly rate of employee
45     emp->hourly_rate = emp->annual_salary / HOURS_PER_YEAR;
46
47
48     emp->hourly_rate = emp->annual_salary / HOURS_PER_YEAR;
49
50     // calculate the gross pay based on the number of hours worked in a week. below 40- normal pay, above 40 - normal + overtime pay
51     if (emp->hours_worked > MAX_NORMAL_HOURS) {
52         int overtime_hours = emp->hours_worked - MAX_NORMAL_HOURS;
53         emp->gross_pay = (MAX_NORMAL_HOURS * emp->hourly_rate) + (overtime_hours * emp->hourly_rate * OVERTIME_RATE);
54     } else {
55         emp->gross_pay = emp->hours_worked * emp->hourly_rate;
56     }
57
58     // Superannuation deduction and net pay calculation
59     double superannuation_deduction_calc = emp->gross_pay * SUPERANNUATION_RATE;
60     emp->net_pay = emp->gross_pay - superannuation_deduction_calc;
61     emp->superannuation_deduction = superannuation_deduction_calc;
62
63     return NULL;
64 }
65
66 int main() {
67     // creating a employee array that contains the initial information for each employee
68     Employee employees[NUM_OF_EMPLOYEES] = {};
69     {"S11219545", "Shamal Prasad", 10000, 0, 0, 0, 0, 0},
70     {"S11198024", "Adrian Obadiah", 12500, 0, 0, 0, 0, 0},
71     {"S11219651", "Shoneel Kumar", 15000, 0, 0, 0, 0, 0},
72     {"S11210021", "Mark White", 22000, 0, 0, 0, 0, 0},
73     {"S11217532", "Eve Black", 25500, 0, 0, 0, 0, 0}
74 };
75
76     int valid_entries_num = 0;
77     int track_entered_employee[NUM_OF_EMPLOYEES];
78
79     //this loop takes in employee id and hours worked until all employees are entered
80     while (valid_entries_num < NUM_OF_EMPLOYEES) {
81
82         char emp_id[10];
83         printf("Enter Employee ID: ");
84         scanf("%s", emp_id);
85
86         // function to check if employee exist in the employee array and returns employees index
87         int employee_index = find_employee_index(emp_id, employees, NUM_OF_EMPLOYEES);
88         if (employee_index == -1) {
89             printf("Employee ID Invalid. Try again.\n");
90             continue; // Restart loop if employee is not found
91         }
92
93
94
95
96
97
98
99
100 }
```

```

85     // Checks if employee info has been entered previously
86     int employee_index = find_employee_index(emp_id, employees, NUM_OF_EMPLOYEES);
87     if (employee_index == -1) {
88         printf("Employee ID Invalid. Try again.\n");
89         continue; // Restart loop if employee is not found
90     }
91
92     // Checks to see if employee info has been entered previously
93     if (track_entered_employee[employee_index] == 1) {
94         printf("Employee %s has already entered. Please enter a different employee.\n", emp_id);
95         continue; // Ask for input again
96     }
97
98     int hours;
99
100    while (true) {
101        printf("Enter hours worked for %s (%s): ", employees[employee_index].employee_id, employees[employee_index].name);
102        scanf("%d", &hours);
103
104        if (hours >= MINIMUM_WORKING_HOURS && hours <= MAXIMUM_WORKING_HOURS) {
105            employees[employee_index].hours_worked = hours;
106            break;
107        } else {
108            printf("Invalid hours! Please enter a value between 1 and 168.\n");
109        }
110
111        // mark this employee as entered
112        track_entered_employee[employee_index] = 1;
113        valid_entries_num++;
114    }
115    pthread_t threads[NUM_OF_EMPLOYEES];
116    // Create threads for each valid employee entry
117    for (int i = 0; i < NUM_OF_EMPLOYEES; i++) {
118        pthread_create(&threads[i], NULL, calculate_payroll, (void *)&employees[i]);
119    }
120
121    // Wait for all threads to complete
122    for (int i = 0; i < NUM_OF_EMPLOYEES; i++) {
123        pthread_join(threads[i], NULL);
124    }
125
126    // Displays results after all calculations are completed
127    printf("\nPayroll Details:\n");
128    printf("_____\n");
129    printf(" %-11s %-15s %-10s %-10s %-10s %-22s \n",
130          "Employee ID", "Employee Name", "Hourly Rate", "Gross Pay", "Net Pay", "Superannuation Deduction");
131    printf("_____\n");

```

```

102
103        if (hours >= MINIMUM_WORKING_HOURS && hours <= MAXIMUM_WORKING_HOURS) {
104            employees[employee_index].hours_worked = hours;
105            break;
106        } else {
107            printf("Invalid hours! Please enter a value between 1 and 168.\n");
108        }
109
110        // mark this employee as entered
111        track_entered_employee[employee_index] = 1;
112        valid_entries_num++;
113    }
114    pthread_t threads[NUM_OF_EMPLOYEES];
115    // Create threads for each valid employee entry
116    for (int i = 0; i < NUM_OF_EMPLOYEES; i++) {
117        pthread_create(&threads[i], NULL, calculate_payroll, (void *)&employees[i]);
118    }
119
120    // Wait for all threads to complete
121    for (int i = 0; i < NUM_OF_EMPLOYEES; i++) {
122        pthread_join(threads[i], NULL);
123    }
124
125    // Displays results after all calculations are completed
126    printf("\nPayroll Details:\n");
127    printf("_____\n");
128    printf(" %-11s %-15s %-10s %-10s %-10s %-22s \n",
129          "Employee ID", "Employee Name", "Hourly Rate", "Gross Pay", "Net Pay", "Superannuation Deduction");
130    printf("_____\n");
131    printf("_____\n");
132
133    for (int i = 0; i < NUM_OF_EMPLOYEES; i++) {
134        printf(" %-11s %-15s %10.2f %10.2f %10.2f %22.2f \n",
135              employees[i].employee_id,
136              employees[i].name,
137              employees[i].hourly_rate,
138              employees[i].gross_pay,
139              employees[i].net_pay,
140              employees[i].superannuation_deduction);
141    }
142    printf("_____\n");
143
144    return 0;
145 }
146
147

```

Result (Kali OS)

```
s11219545@s11219545:~/CS311/A1
File Actions Edit View Help
(s11219545@s11219545)-[~/CS311/A1]
$ gcc A1P2.c -pthread -o A1P2
(s11219545@s11219545)-[~/CS311/A1]
$ ./A1P2
Enter Employee ID: S11219545
Enter hours worked for S11219545 (Shamal Prasad): 25
Enter Employee ID: S11198024
Enter hours worked for S11198024 (Adrian Obadiah): 74
Enter Employee ID: S11219651
Enter hours worked for S11219651 (Shoneel Kumar): 39
Enter Employee ID: S11210021
Enter hours worked for S11210021 (Mark White): 41
Enter Employee ID: S11217532
Enter hours worked for S11217532 (Eve Black): 17

Payroll Details:
Employee ID Employee Name Hourly Rate Gross Pay Net Pay Superannuation Deduction
S11219545 Shamal Prasad 4.81 120.19 110.58 9.62
S11198024 Adrian Obadiah 6.01 546.88 503.12 43.75
S11219651 Shoneel Kumar 7.21 281.25 258.75 22.50
S11210021 Mark White 10.58 438.94 403.83 35.12
S11217532 Eve Black 12.26 208.41 191.74 16.67

(s11219545@s11219545)-[~/CS311/A1]
$
```

Result (Ubuntu OS)

```
ubuntu@ubuntu:~$ cd Documents/
ubuntu@ubuntu:~/Documents$ gcc A1P2.c -pthread -o A1P2
ubuntu@ubuntu:~/Documents$ ./A1P2
Enter Employee ID: S11219545
Enter hours worked for S11219545 (Shamal Prasad): 25
Enter Employee ID: S11198024
Enter hours worked for S11198024 (Adrian Obadiah): 74
Enter Employee ID: S11219651
Enter hours worked for S11219651 (Shoneel Kumar): 39
Enter Employee ID: S11210021
Enter hours worked for S11210021 (Mark White): 41
Enter Employee ID: S11217532
Enter hours worked for S11217532 (Eve Black): 17

Payroll Details:
Employee ID Employee Name Hourly Rate Gross Pay Net Pay Superannuation Deduction
S11219545 Shamal Prasad 4.81 120.19 110.58 9.62
S11198024 Adrian Obadiah 6.01 546.88 503.12 43.75
S11219651 Shoneel Kumar 7.21 281.25 258.75 22.50
S11210021 Mark White 10.58 438.94 403.83 35.12
S11217532 Eve Black 12.26 208.41 191.74 16.67

ubuntu@ubuntu:~/Documents$
```

POSIX threads stand for Portable Operating System Interface and is a standard threading interface that is defined by the IEEE and provides an API for creating and managing threads in a Unix-based operating system (Wikipedia, 2025). A pthread or POSIX threads allows programs to execute their processes simultaneously.

The pthread_t is first used to identify a thread or in our case an array of threads. After which the pthread_create() function is used to create a POSIX thread that accepts the thread address, memory information, function that it will be executing and any arguments the pthread_create() function needs to execute the thread. After the threads have completed executing, the pthread_join() function is then used to block a thread that has finished executing until all the created threads have finished its execution, and then the program starts execution after the termination of all the threads.

Synchronization is the process of coordinating the execution of the program's multiple processes in such a way that no 2 processes can access the same resource at a single given time (Geek For Geeks, 2025). This prevents race conditions where many processes are trying to access the same shared resource and modify it. If the processes are not synchronized, then one process may use up to date information and other processes may use out of date information that can lead to errors in processing of the program. An example of this could be when a person is trying to withdraw from a bank account and another person is trying to deposit into the bank account, if the two processes are not synchronized, then the final balance will be different at the different end points. The person depositing will have a higher balance than the person withdrawing.

Part C: Critical Thinking & Analysis

Question 1

i) The process scheduling algorithm I would implement to meet these requirements would be a Preemptive Priority Based Multilevel Queue Scheduling where each type of process will be divided into 3 levels based on their priority. Interactive processes will be given highest priority, Real-time processes second priority while batch processes have last priority. The reason for choosing the above mentioned scheduling algorithm is because it has the ability to divide the processes into several hierarchy queues and each queue possesses a different priority, and process type. The scheduling algorithm also ensures that processes with higher priority get scheduled first, which can then finish in a timely manner, and lower priority gets scheduled whenever the CPU is idle.

ii) The issues could arise if the scheduler does not handle process priorities effectively are the following:

A) Starvation of Low Priority Processes - if low level processes are never scheduled and allocated the needed resources, then they will never complete. Example a backup process in your windows machine. It is a low priority event, but if the CPU only schedules high and medium priority events then the backup process will never get the opportunity to execute. A simple solution will be Aging where a process in waiting has their priority increased for every certain amount of time it is waiting.

B) Missed Deadline - This is when a high priority process misses its chance to be scheduled because a low level process is being processed for too long and does not give the opportunity and resources for a higher priority level process to be scheduled by the CPU (Convoy Effect). Examples of this can be seen when a background program is taking up the CPU utilization and a video game application that relies on real time processing, does not have enough CPU and GPU resources to execute properly and leads to lags or dips in performance. The preemptive nature of the scheduling process allows context switching from lower priority events to higher priority.

Question 2

- i) The synchronization technique I would employ would be a Mutual Exclusion Lock, also known as Mutex. Mutex is used in concurrent programming to place a lock on a shared resource to avoid simultaneous use of a common resource (Geek For Geeks, 2024). This would prevent race conditions and ensure data consistency. In the scenario stated in the question, Thread 1 adds to the inventory while Thread 2 wants to remove from the inventory. The variable that tracks how many items are in the inventory must be accessed by both threads. If both threads access the variable at the same time , both threads will change the shared variable's value. While thread 1 is adding to inventory the number of items will increase and at the same time thread 2 will remove items from inventory, they will both change the variable and at the end of the transaction the value of the variable will be different, which leads to data inconsistency. Mutex lock will ensure only one thread accesses the shared variable at a time and this will ensure the consistency of the data.
- ii) The potential issues could arise if the synchronization mechanism is not implemented are the following:
- a) Deadlock - Mutex is used to lock resources to certain threads and processes, but improper locking could lead to deadlock. Deadlock is where 2 or more threads that have some resources allocated, but are waiting for resources that other threads are holding on to. The threads cannot proceed until each thread releases the resource but a thread can not release the resource until it acquires the necessary resource to finish execution which leads to a deadlock. This issue can be rectified by employing deadlock prevention and avoidance mechanisms, such as Banker's Algorithm that ensures that the system stays in a safe state when allocating resources to processes (Geek For Geeks, 2025).
 - b) Race conditions - this is where threads fight to acquire shared resources to continue processing. Improper lock on shared resources can lead to data inconsistency where each thread manipulates the shared resources differently and other threads may use out of date information (Moussa, 2023). To solve this problem, Mutex locks need to be used to prevent race conditions and promote data integrity.

Reference

1. University of the South Pacific. (2025). *CS311 operating systems - Assignment 1 Semester 1, 2025*. School of Information Technology, Engineering, Mathematics, and Physics.
2. Geek For Geeks. (2024, July 22). *Mutual Exclusion in Synchronization*. Geek For Geeks. <https://www.geeksforgeeks.org/mutual-exclusion-in-synchronization/>
3. Geek For Geeks. (2025, January 10). *Introduction of Process Synchronization*. Geek For Geeks. <https://www.geeksforgeeks.org/introduction-of-process-synchronization/>
4. Geek For Geeks. (2025, January 24). *Banker's Algorithm in Operating System*. Geek For Geeks. <https://www.geeksforgeeks.org/bankers-algorithm-in-operating-system-2/>
5. Moussa, I. (2023, May 26). *Deadlock and Race Condition in iOS: Understanding and Best Practices*. Medium. <https://medium.com/@islammoussa.eg/deadlock-and-race-condition-in-ios-understanding-and-best-practices-6fc1aa6e6e50#:~:text=Best%20Practices%20for%20Avoiding%20Deadlocks%20and%20Race%20Conditions&text=Always%20try%20to%20minimize%20the,safety%20when%20acce>
6. Mulani, N. (2022, December 6). *Understand Process Synchronization with Example?* PrepBytes Blog. <https://www.prepbytes.com/blog/operating-system/process-synchronization-in-os-definition-working-and-example/#:~:text=Process%20synchronization%20in%20OS%20is,without%20interfering%20with%20each%20other>
7. Parker, J. (2024, 08 03). *What is the purpose of Kali*. James Parker. Retrieved april 04, 2025, from <https://www.jamesparker.dev/what-is-the-purpose-of-kali-linux/>
8. Wikipedia. (2025, February 19). *Pthreads*. Wikipedia. <https://en.wikipedia.org/wiki/Pthreads>