

AAA Lab

Evan Choi | Cisco CCNP | 6/10/22

**Purpose**

The purpose of this lab is to configure AAA on Linux and a windows server to ensure your network is secure by making sure only the right users are given access, and that they are only given access to network resources set by the network administrator.

**Background Information on lab concepts**

AAA is a protocol that controls access to computer resources, enforces policies, audits usage, and provides essential information to network administrators. It keeps your network secure by making sure that only the right users are authenticated, and that those users can only access the network resources they have permissions to, and that what they do is logged so network administrators can see what they did. AAA stands for Authentication, Authorization, and Accounting. Authentication identifies users by using a login and password before the user ever gains access to the network. Authorization happens after authentication. It looks at what the authenticated user has access to do and performs authorization for specific privileges by defining attribute-value or AV pairs. Accounting happens last. It lets network administrators see what users they do once are authenticated and authorized, and also provides the network administrator with a way of collecting security information that can be used for auditing, billing, and reporting.

AAA consists of two main security protocols: TACACS+ and RADIUS. These two protocols can be used similarly and also simultaneously, but they also have many differences.

TACACS+ or Terminal Access Controller Access Control System, is a remote authentication protocol that allows a remote access server to communicate with an authentication server to verify user access into a network. TACACS+ has a client called the Network Access Server (NAS) or the Network Access Device (NAD) and all traffic between the NAS and the TACACS+ process is encrypted by TACACS+. Whenever there is an authentication login attempt on a NAS, it is verified by a remote TACACS+ process. This process happens on a software or program running on a security server, providing AAA services using TACACS+. TACACS.net is an example of a software that does this. The program will process authentication, authorization, and accounting requests from a NAS. When processing authentication requests, the process may request additional information, like a password from the NAS. To send these requests TACACS+ authentication uses three packet types. Start and Continue packets are always sent by the user, and reply packets are always sent by the TACACS+ process. The process starts with TACACS+ setting up a TCP connection to the TACACS+ host and sending a Start packet. The TACACS+ host then responds by sending a Reply packet, which will either give or deny access, report an error, or challenge the user. TACACS+ may challenge the user by requesting them to provide username, password, passcode, or other information. Once the requested information is entered, TACACS+ sends a Continue packet through the existing connection. The TACACS+ host will then send a Reply packet. Users are only allowed three login retries, and once this authentication process is completed, the connection will close. If a user wants privilege authentication, which determines if a user is allowed to use commands on a specific privilege level, the process is very similar, however the user is limited to only one authentication attempt. If the TACACS+ host sends an empty reply, access is denied. To configure authorization, you can use commands like the aaa authorization exec command, which will determine the user privilege level when users are authenticated. To configure accounting, you must use software such as JunosE, where you can define accounting method lists and associate consoles and lines with method lists.

RADIUS or Remote Authentication Dial-In User Service is a client-server protocol and software which enables remote access servers to communicate with a central server to authenticate dial-in users and authorize what access they have to the requested system. To establish communication between a network access server (NAS) and a RADIUS server, UDP or User Datagram Protocol is used. However, RADIUS is generally connectionless, as issues relating to server availability, timeouts, and retransmission are handled by the RADIUS-enabled devices instead of the transmission protocol. RADIUS is a client/server protocol, and the RADIUS client is usually a NAS while the RADIUS server is usually a daemon process running on a Windows NT machine or UNIX. The client sends user information to RADIUS servers and functions based on the response. RADIUS servers do multiple things, such as receive user connection requests, authenticate the user, and return the configuration information the client needs to deliver service to the user. It can also act as a proxy client to other RADIUS servers or authentication server. The interaction between a dial-in user and the RADIUS client and server happens in 6 steps. First, the user initiates PPP authentication to the NAS. PPP authentication consists of the Password Authentication Protocol (PAP) and Challenge-Handshake Authentication Protocol (CHAP). Then the NAS either prompts for a username and password (PAP) or challenge (CHAP). The user then replies with the requested information, and the RADIUS client sends the username and encrypted password to the RADIUS server. The RADIUS server will then respond with Accept, Reject, or Challenge, and finally, the RADIUS client will then act upon services and services parameters bundled with Accept or Reject. When the RADIUS server receives the Access-Request from the NAS, it will search a database for the username list, and if that username doesn’t exist in the database, it will either load a default profile, or it will immediately send an Access-Reject message. The Accept-Reject message may come with a message that indicates the reason for the refusal. In RADIUS, authentication and authorization are grouped together, so if the entered username and password are found and correct, the RADIUS server will return an Access-Accept response, which will also include a list of attribute-value pairs which describe what parameters should be used during that session. The parameters usually include service type (shell or framed), protocol type, the IP address which the user should be assigned (static or dynamic), the access list to apply, or a static route to install in the NAS routing table. The information that has been configured in the RADIUS server will determine what to install on the NAS. Accounting features in RADIUS are used independently of RADIUS authentication or authorization. They allow data to be sent at the start and end of sessions, which indicate how many resources like time, bytes, packets, etc were used during a session.

TACACS+ and RADIUS are similar in their processes. Both processes are started by a Network Access Device (NAD). The NAD contacts the TACACS+ or RADIUS server and transmits the request for authentication to the server. The NAD then obtains the username prompt and transmits it to the TACACS+ or RADIUS server, and then does the same for the password. Then, the TACACS+/RADIUS server replies with an access-accept message if the credentials are valid, otherwise it sends an access-reject message to the client.

TACACS+ and RADIUS also have many differences. TACACS+ is created by Cisco while RADIUS is created by IETF. TACACS+ also uses full packet encryption, making it more secure, while RADIUS uses only password encryption. TACACS+ also uses TCP, and TCP Port 49, while RADIUS uses UDP and UDP Port 1812 and 1645 for Authentication, and UDP Port 1813 1646 for Accounting. In addition, TACACS+ uses bidirectional CHAP while RADIUS uses unidirectional CHAP, but it also needs more resources. TACACS+ is also generally used for administration and has full command logging, while RADIUS is generally used for network access and has no command logging. Also, while TACACS+ has separate authentication, authorization, and accounting, RADIUS has combined Authentication and Authorization, with separate accounting. TACACS+ additionally has limited accounting, but supports 15 privilege modes, while RADIUS has extensive accounting but only supports one privilege mode.

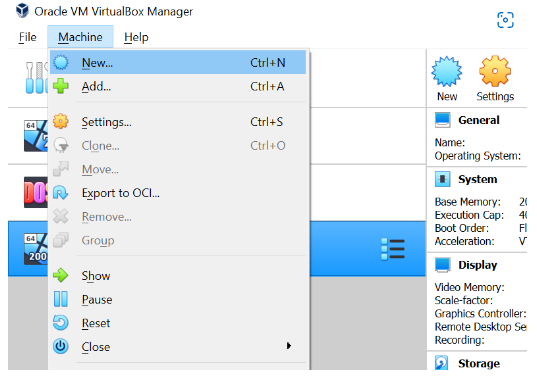
There are also many advantages and disadvantages of using TACACS+ instead of RADIUS or RADIUS instead of TACACS+. TACACS+ more reliable than RADIUS since it uses TCP, and it also provides more control over the authorization of commands while in RADIUS, there is no support for external authorization for commands. TACACS+ is also more secure in the fact that all the AAA packets are encrypted, while in RADIUS, passwords are only encrypted. RADIUS is advantageous in how it is open standard and can be used with other vendor’s devices, while TACACS+ is Cisco proprietary, meaning it can only be used by Cisco devices. RADIUS also has more extensive accounting support than TACACS+.

**Lab Summary**

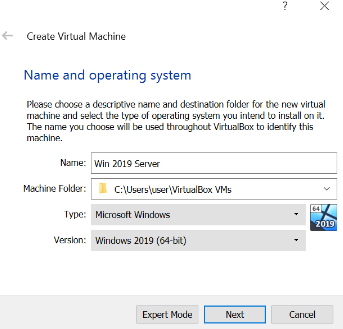
**How to configure TACACS+**

Setting up your VM

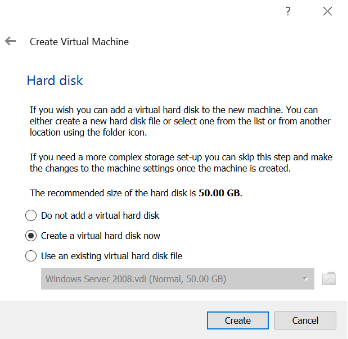
1. Install VirtualBox and open it. Click the **Machine** button at the top and click **New.** This will create a new machine.



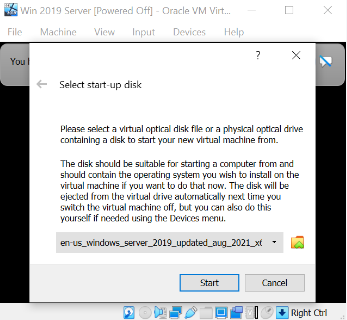
1. Name the Virtual Machine and select a version. We will be using **Windows 2019 (64-bit).**



1. Create a Disk Drive and allocate it space. 50 GB should be fine.

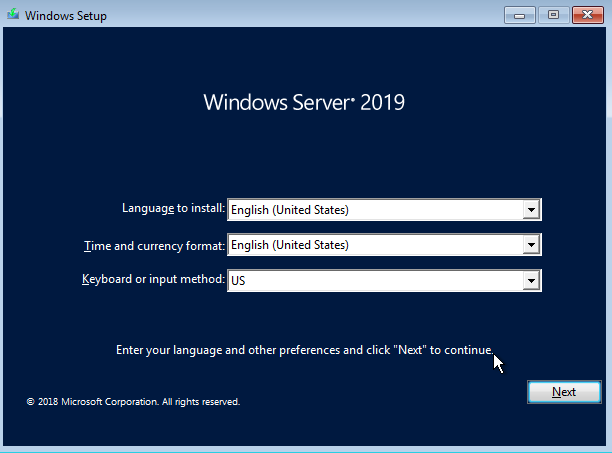


1. Launch the VM once the disk is created. When prompted, click the dropdown and find the **Window Server ISO file location.** Click **start.**

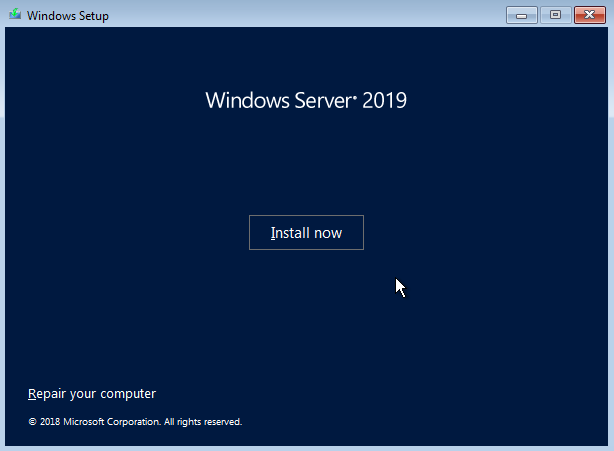


Installing a Windows Server

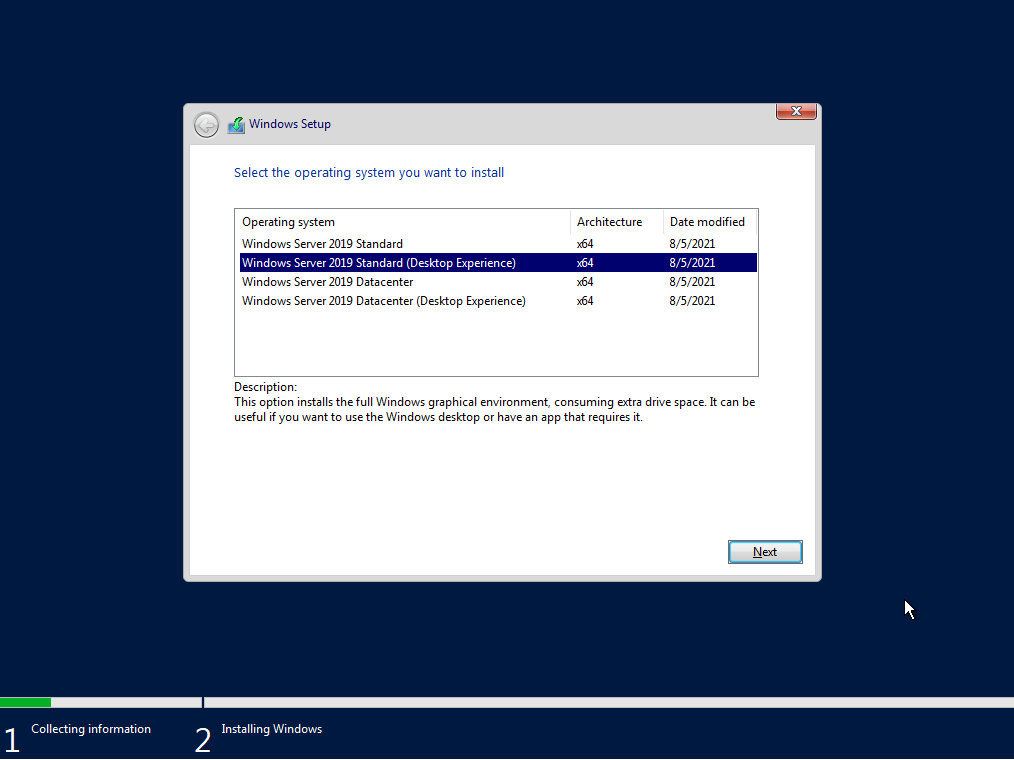
1. Set your language preferences.



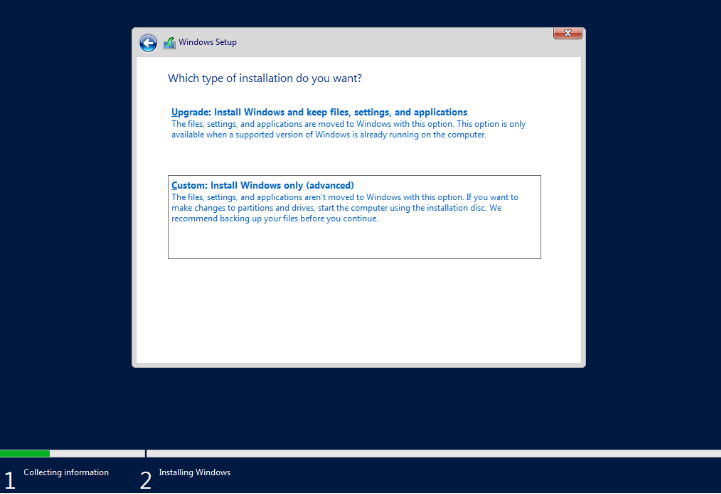
1. Click **Install Now.**



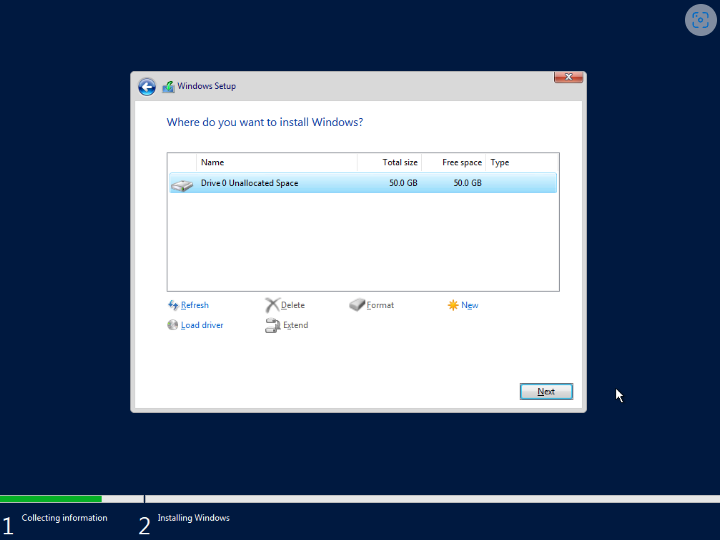
1. Select the operating system you want to install. I used **Windows Server 2019 Standard (Desktop Experience).** Click **next.**



1. Select **Custom: Install Windows only (advanced).**

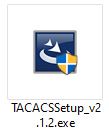


1. Select the available partition and click **Next.** Your windows server should be installed.

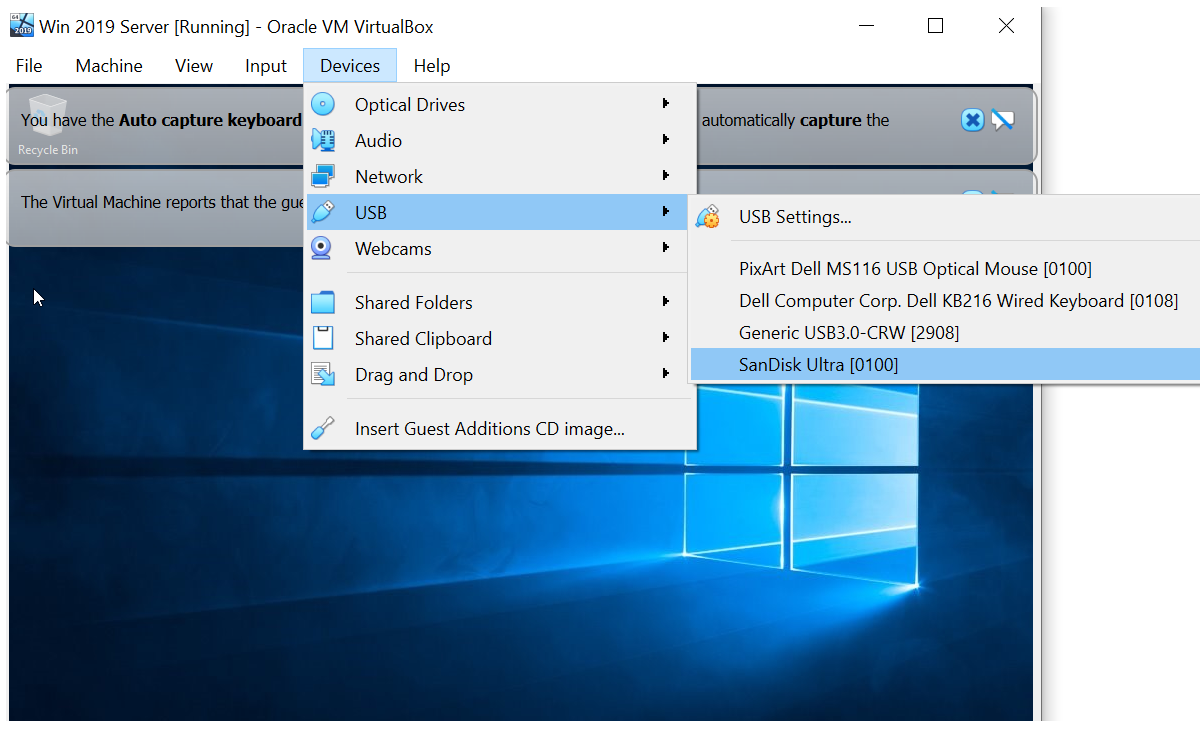


Install TACACS+

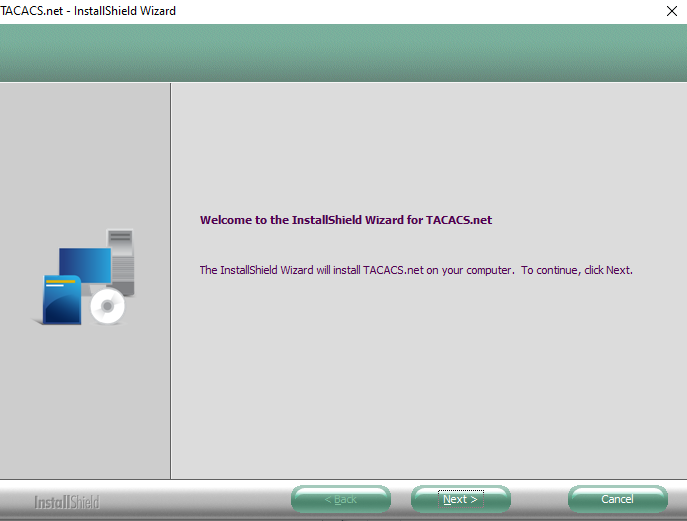
1. Download **TACACS+** from **TACACS.net.**



1. Transfer the file from your computer to the Virtual Machine using a USB drive. You can do this by clicking **Devices > USB > [your USB drive]**



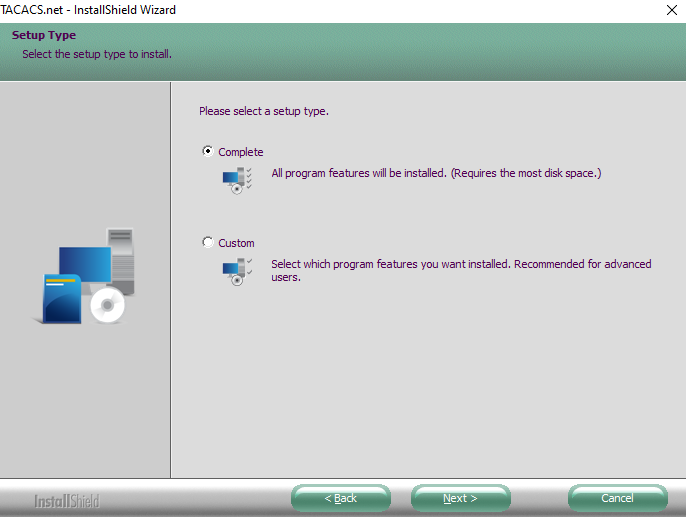
1. Run the **TACACSSetup\_v2.1.2.exe** file.
2. You’ll be prompted with this screen. Click **Next**



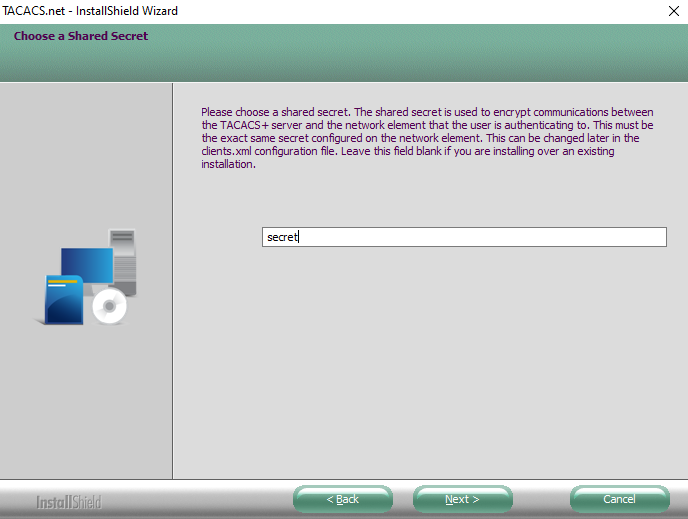
1. Read the End-User License Agreement. Click **I accept the terms of the license agreement.** Then click **Next.**



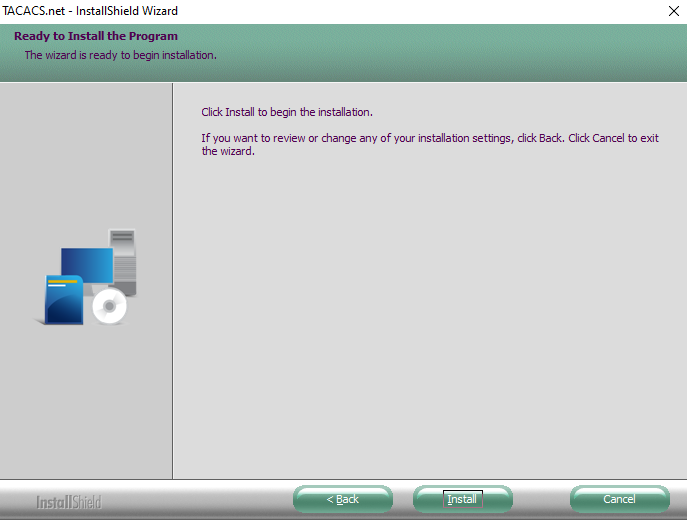
1. Select **Complete.** This should be selected by default. Click **Next.**



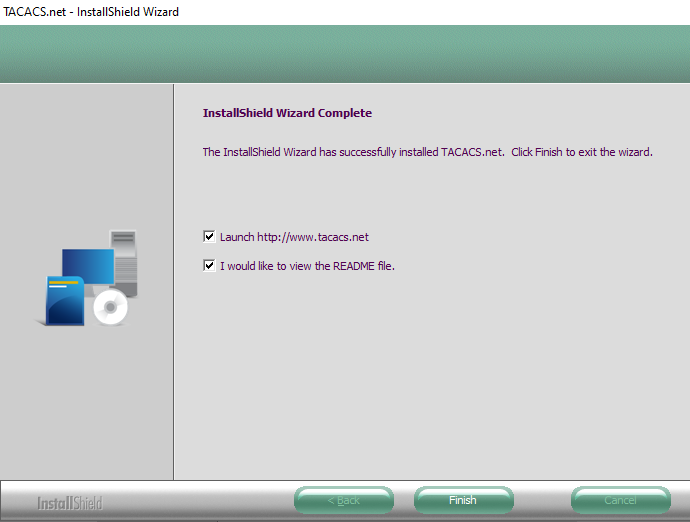
1. You will be prompted to enter a **key.** You’ll need this later so remember it. Click **Next**



1. Click the **Install** button to install TACACS.net

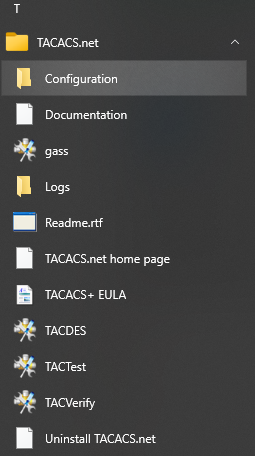


1. Click **Finish.** You’ve installed TACACS.net services..



Configure TACACS+

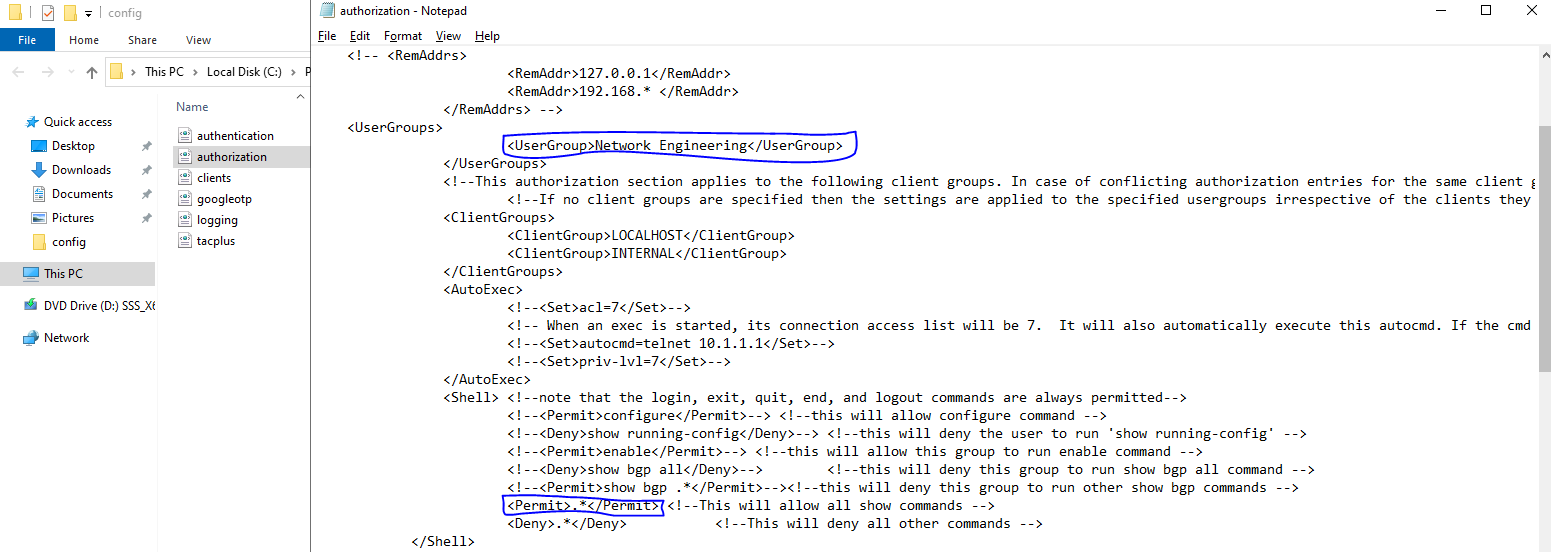
1. To open the TACACS configuration folder, click the **Windows Key** then till you find a folder called **TACACS.net.** Click the down arrow and open the **Configuration** file.



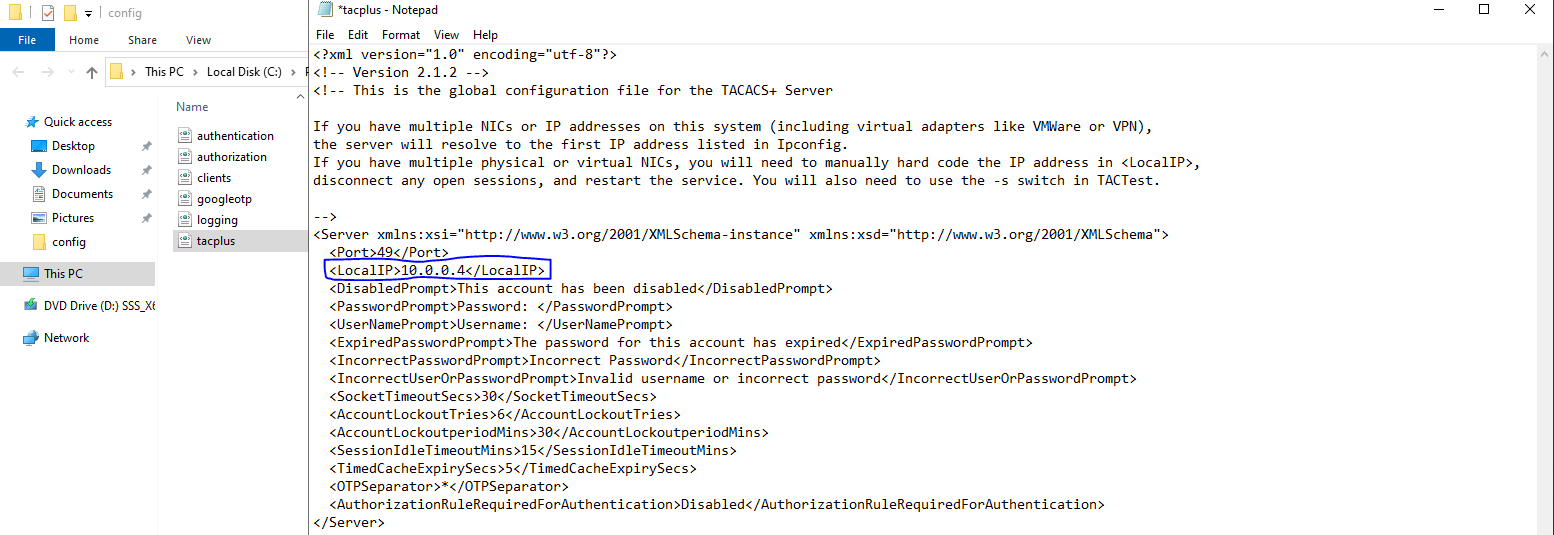
1. We need to edit 3 different files: authentication, authorization, and tacplus. We will start with **authentication.** Open the **authentication.xml** file. To set your username, in the space between **<Name>** and **</Name>,** type your desired username. To set your login password, in the quotations after **ClearText=** type your desired login password. To set your enable password, in the quotations after **ClearText=** type your desired enable password.



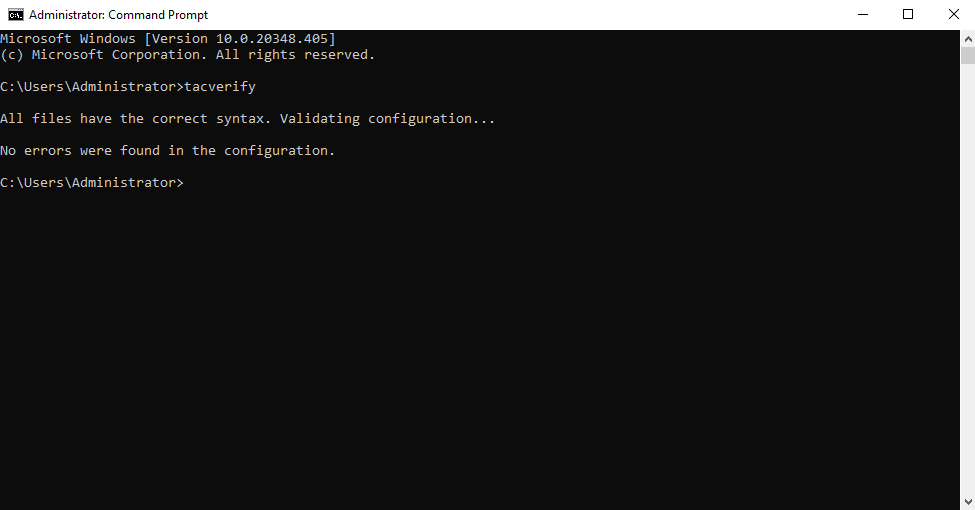
1. Now, open the **authorization.xml** file. Scroll until you see the UserGroup called **“Network Engineer.”** It should look like this **<UserGroup>Network Enigeering</UserGroup>.** Find the **<Shell>** tag, and change <Permit>.\*show.\*</Permit> to **<Permit>.\*</Permit>.**



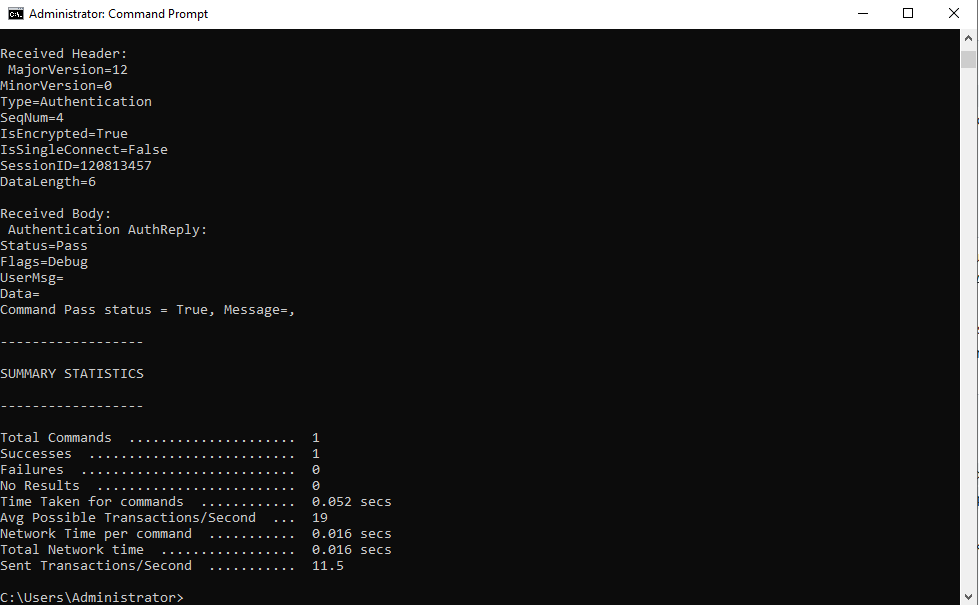
1. Lastly, open the **tacplus.xml** file. In between the **<LocalIP>** and **</LocalIP>** tag, add the IP of your TACACS+ Server.



1. To verify that there are no syntax errors in your files, enter the **tacverify** command in command prompt.



1. To test that your TACACS+ service is working, enter the **tactest** command. An example of this command follows: **tactest –s <IP> -k <Server key> -u <Username> -p <Password>.**



Configuring TACACS+ on your Cisco Router

1. To define a new AAA model with tacacs+, enter the following commands in your router:

**aaa new-model**

**aaa authentication login default group tacacs+**

**aaa authentication enable default group tacacs+**

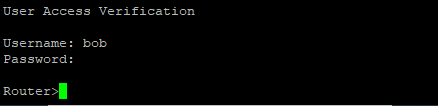
1. To create a new tacacs server enter the following commands. The **<Server key>** is the key you entered in step 16:

**tacacs server <Name>**

**address ipv4 <IP>**

**key <Server Key>**

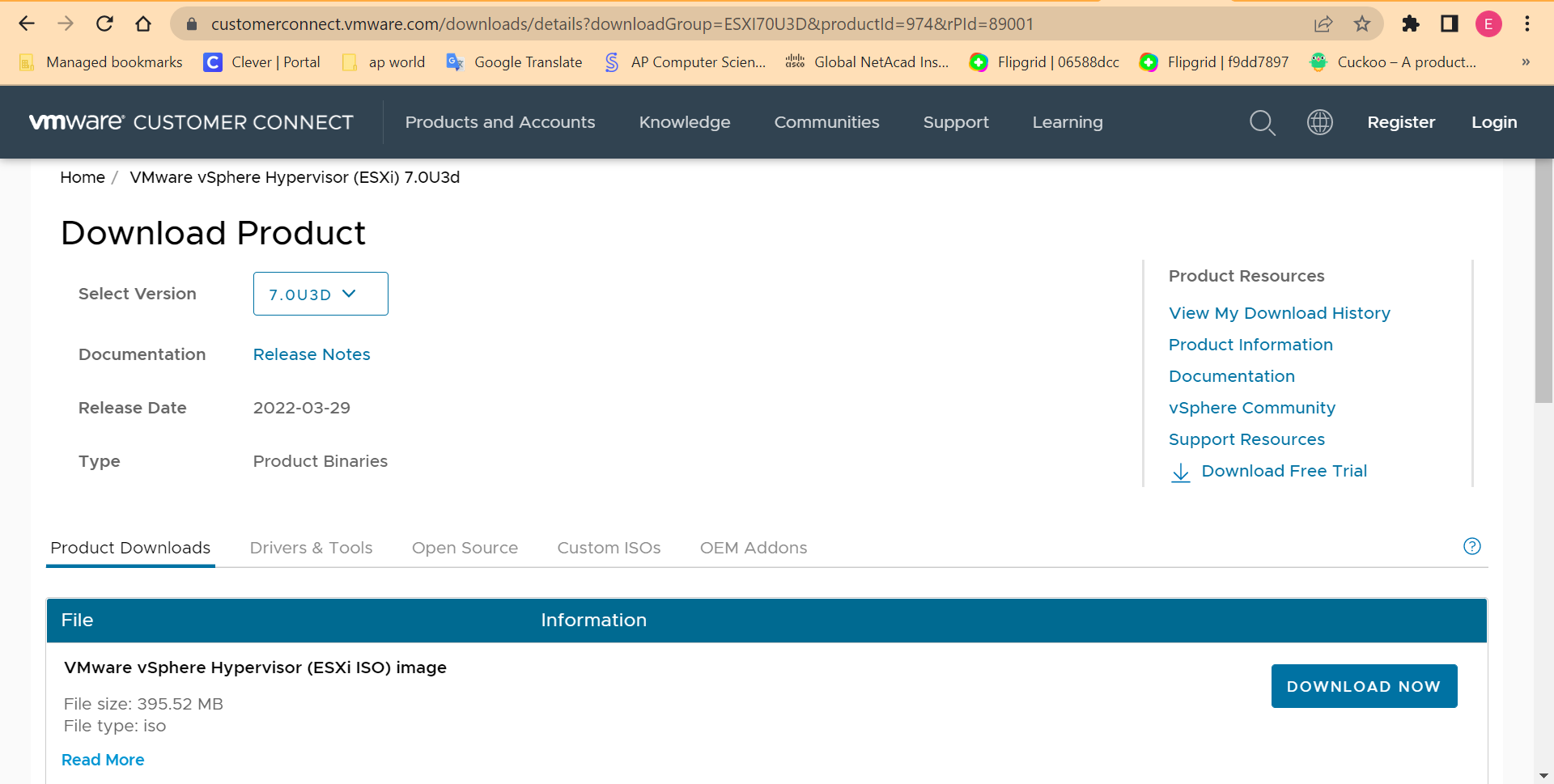
1. Now test if it works. You should be in routing protocol configuration mode, so type **exit** three times. This should bring you to the login prompt. Enter the username you created in step 20 for the username and enter the password you create in step 20 for the password. Click **enter.** You should now be in User Execution Mode.



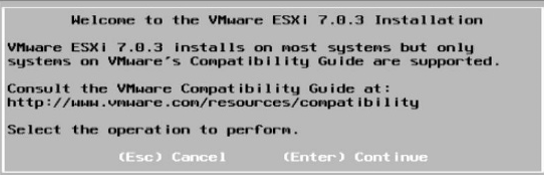
**How to configure RADIUS**

Installing VMware ESXi

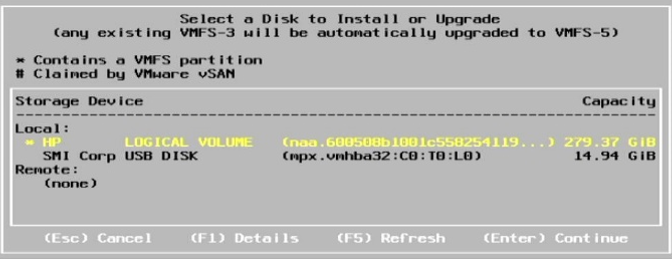
1. Download the **ISO file** from the vmware website: <https://customerconnect.vmware.com/downloads/details?downloadGroup=ESXI70U3D&productId=974&rPId=89001>



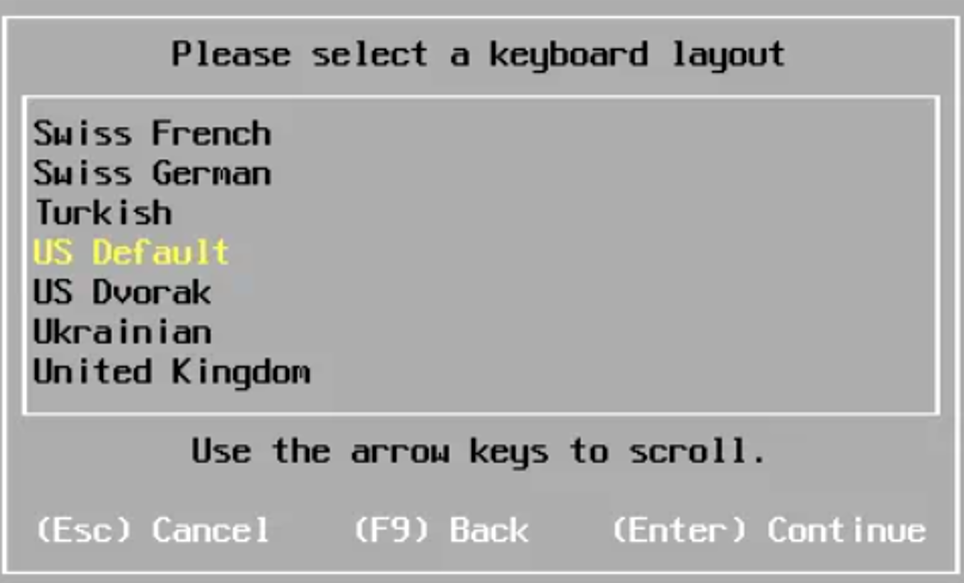
1. Put the file into a USB drive using **Rufus**
2. Plug the USB drive into your server and boot the USB drive.
3. Once it runs, you will be greeted by this screen. Press **enter**



1. To accept the End User License Agreement, press **F11.**
2. The menu below should appear. This is where you will select which storage device to install ESXi to. The largest volume is most likely going to be the internal hard drive of your server.



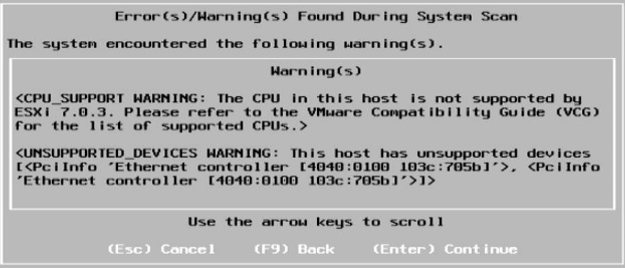
1. Press **Enter.**
2. Select **“install ESXi, overwrite VMFS datastore,”** and press **space**, then **enter**
3. Now, select your keyboard layout



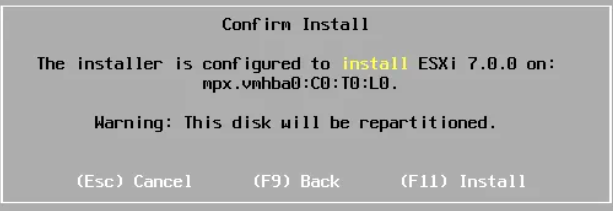
1. Enter a password



1. Press **enter** to install



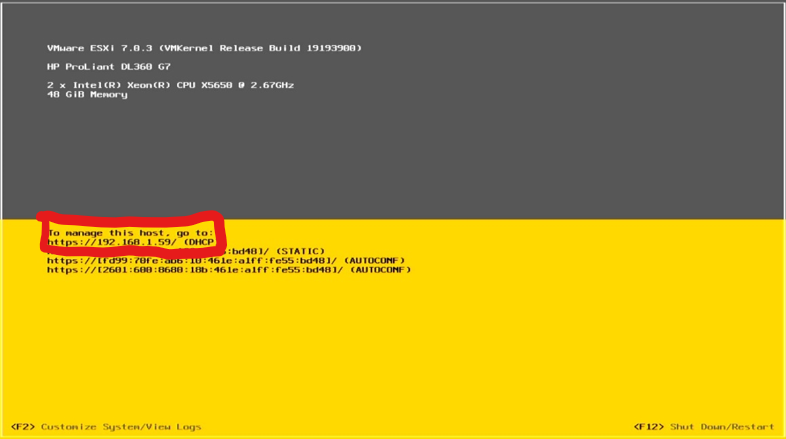
1. To confirm the install, press **F11**



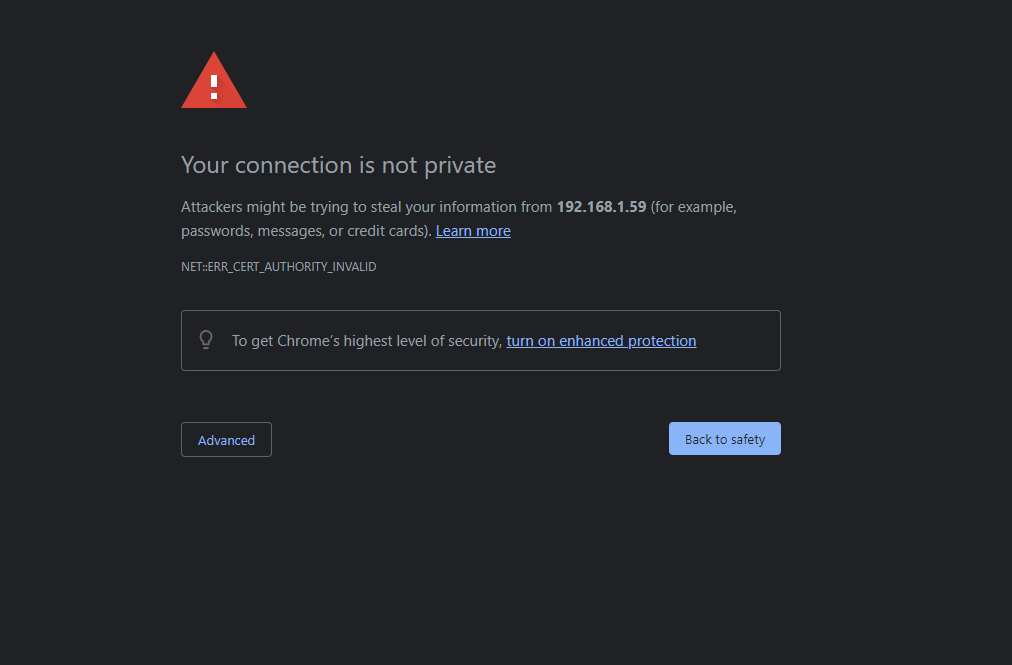
1. When the installation is complete it should look something like the image below. Once it’s installed, remove the USB drive and press **enter** to reboot the server



1. Connect the server to your network
2. Once it finishes booting, it should look something like this. Keep not of the IPv4 address since you will use this to manage the server.



1. Open your web browser and enter the **IP address** of the server into the search bar. Something like this might appear. To get past this, press **advanced,** then press **proceed.**

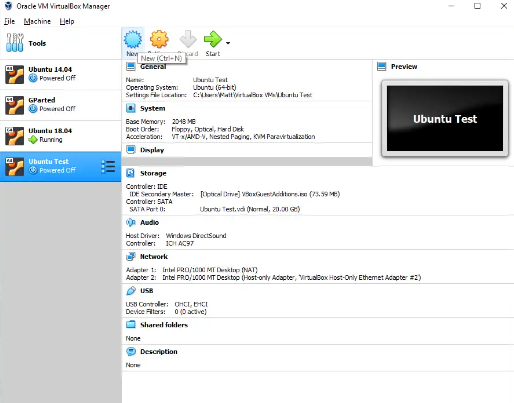


1. A screen like this should now appear. Login with the username root and password you set.

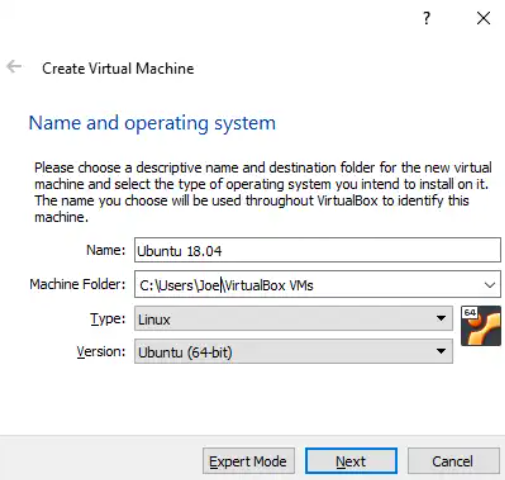


Creating an Ubuntu server virtual machine

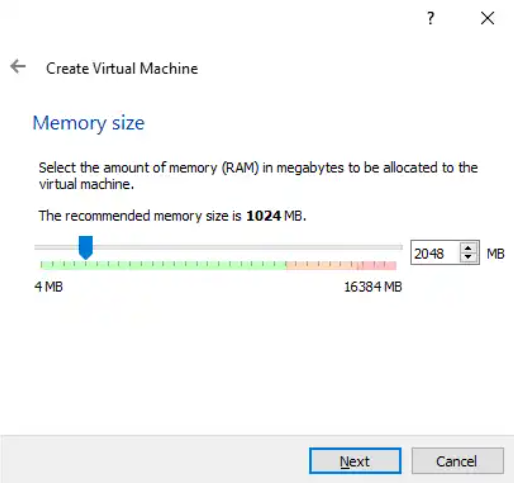
1. Download and install the latest version of **Oracle VM VirtualBOx** for your host OS. <https://www.virtualbox.org/wiki/Downloads>
2. Download a copy of Ubuntu Server v18.04.3 LTS 64-bit (Bionic Beaver) from <https://releases.ubuntu.com/18.04/>
3. Now, we’re going to create a Virtual Machine. First, **start VirtualBox**.
4. Next, click the **New** button



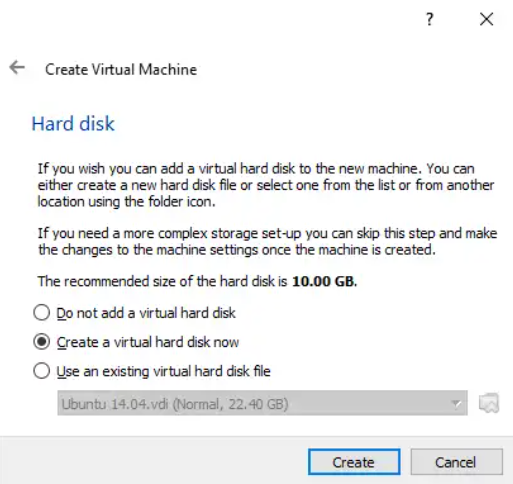
1. Fill out the name and operating system



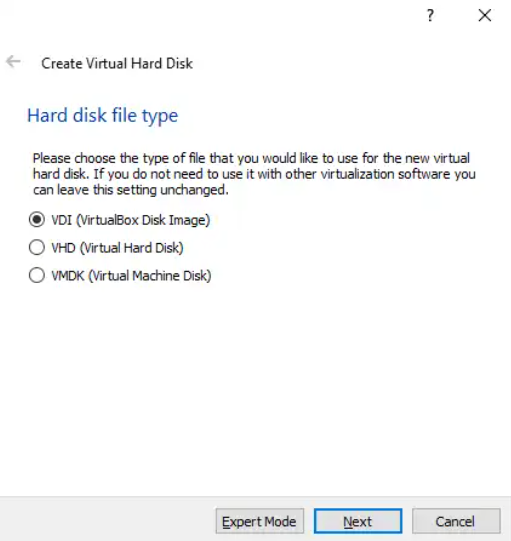
1. Now, choose how much memory you need. I used 1GB.



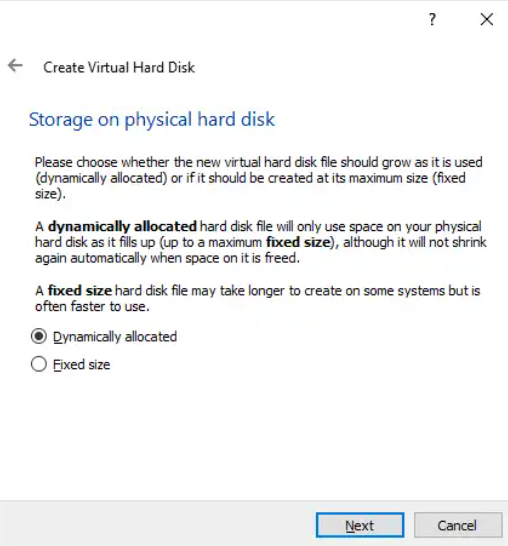
1. Create a virtual hard drive



1. Select the **VDI (VirtualBox Disk Image)** as your hard drive file type



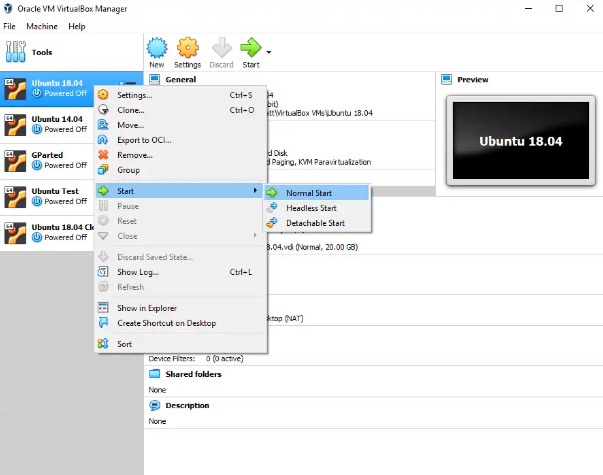
1. Set Storage on physical hard disk to **dynamically allocated**



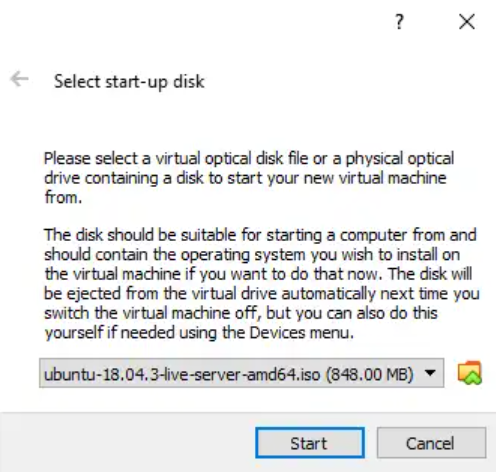
1. Change the size to be at least **20GB.**



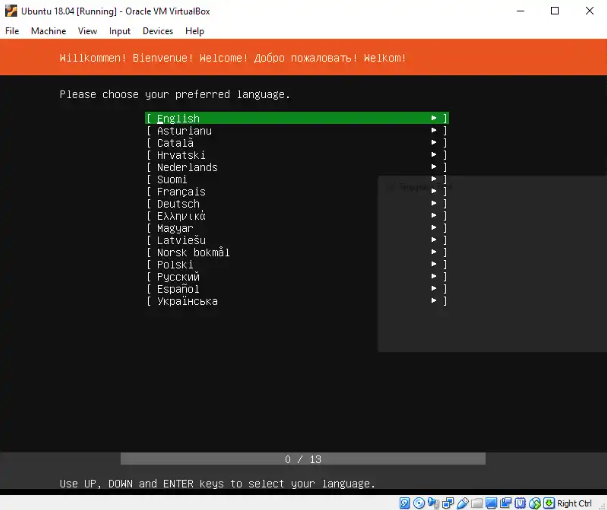
1. Click **Create.**
2. Now, start the virtual machine.



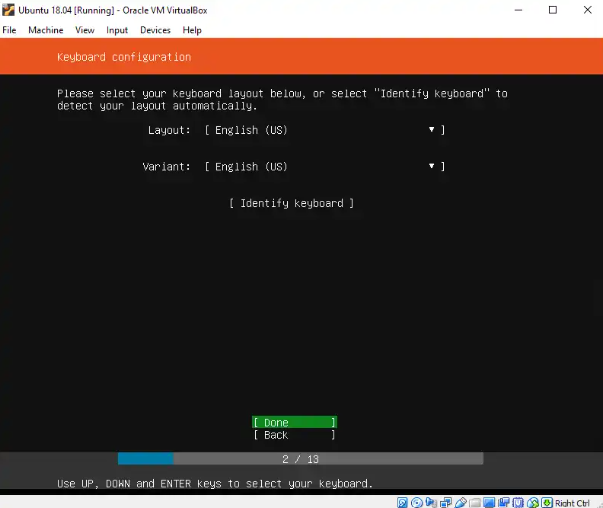
1. Once prompted, select the **Ubuntu 18.04 iso** you downloaded in step 17.



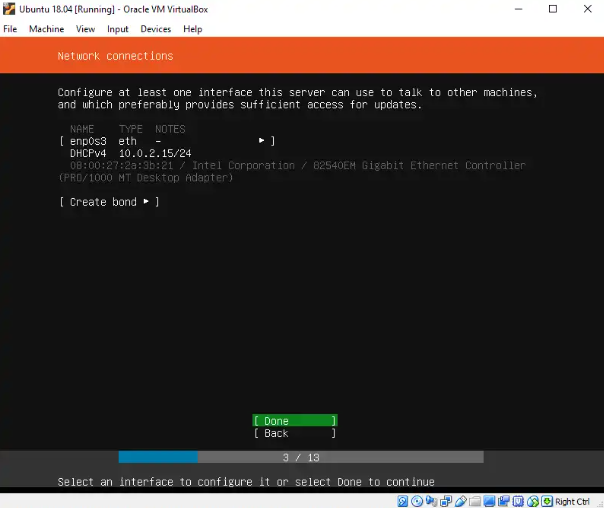
1. Select your preferred language



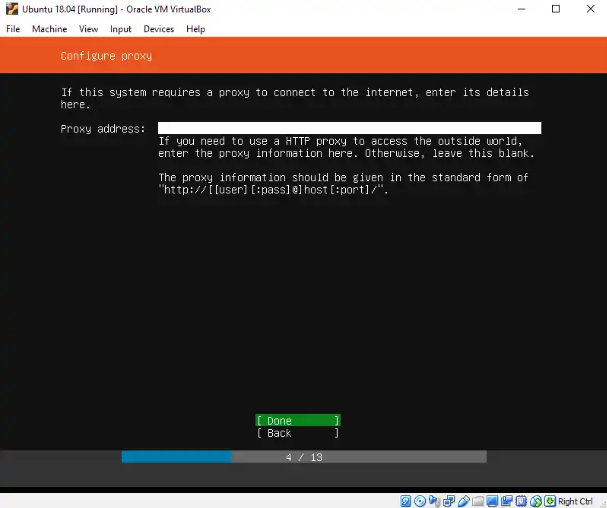
1. Select your keyboard config



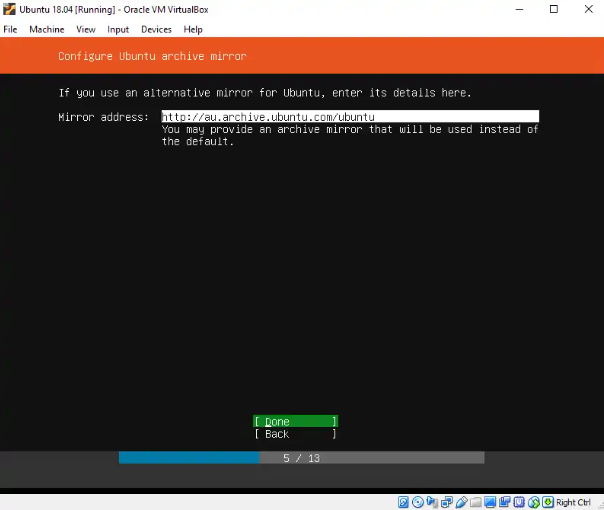
1. To select the default network connection configuration, press **Enter**



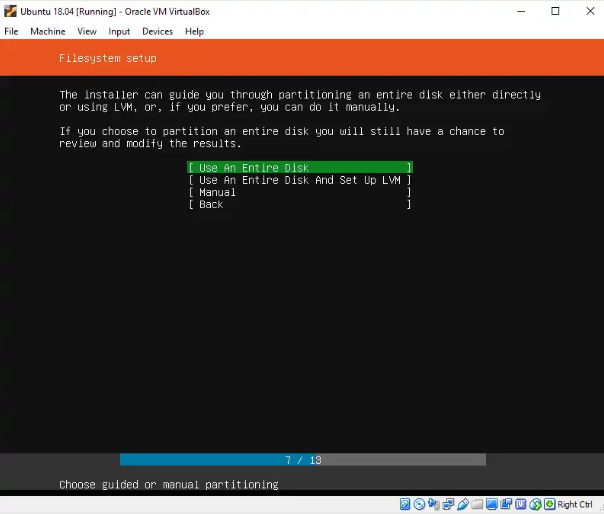
1. To skip adding a proxy, press **Enter**



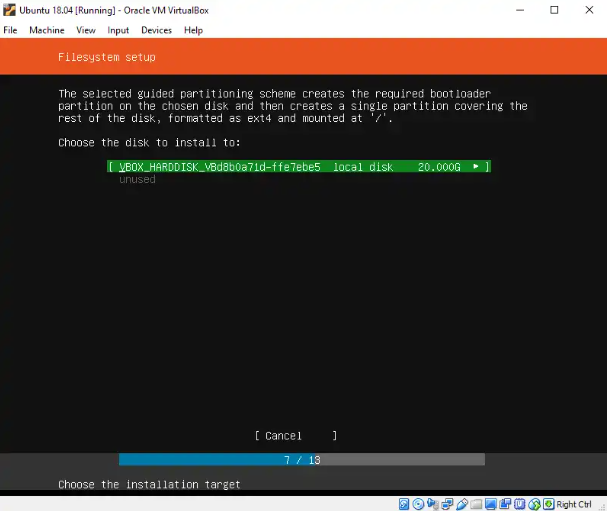
1. To select the default Mirror address, press **Enter**



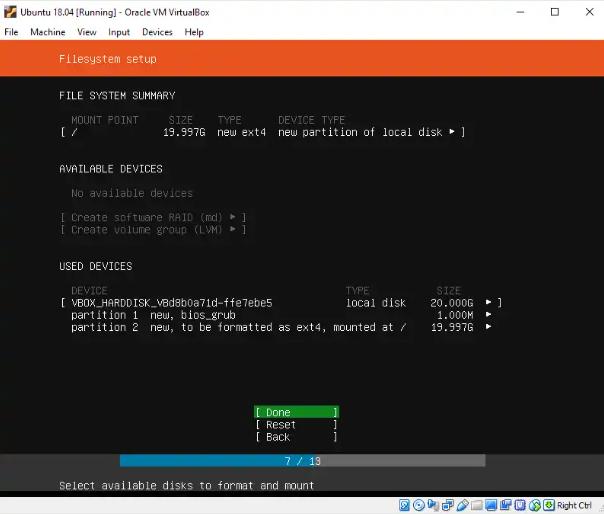
1. Make sure that **“Use an Entire Disk”** is selected, and press **Enter**



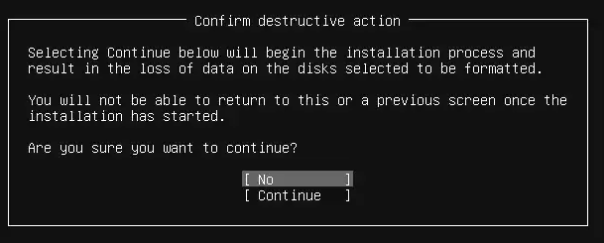
1. You will be presented with the virtual disk image you created earlier. **Press enter**



1. You should then be presented with the hard disk partition schema of the new virtual machine. Make sure that **“Done”** is highlighted, and press **Enter.**



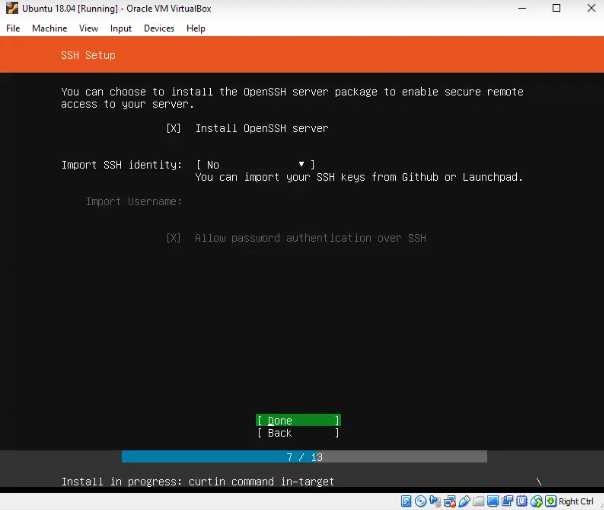
1. To confirm the hard disk config, select **“Continue”** and press **Enter.**



1. While it’s installing, you need to enter some user information to create the default user.



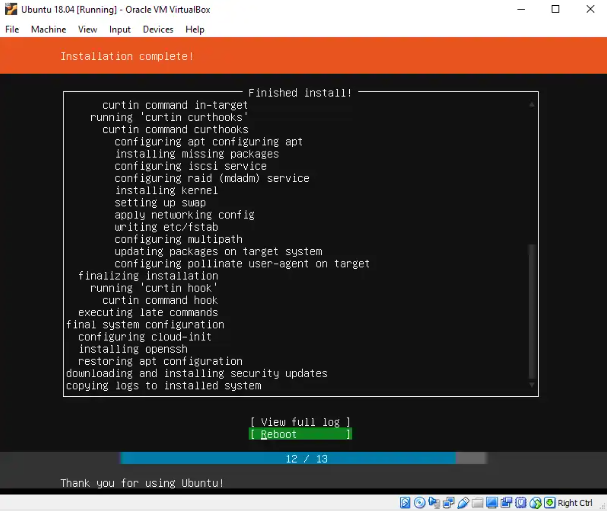
1. Click **“Done”**
2. Now, install OpenSSH so you can SSH into your server. Press the **Space Bar** to select the **“OpenSSH server”** option, select **“Done”** and press **Enter.**



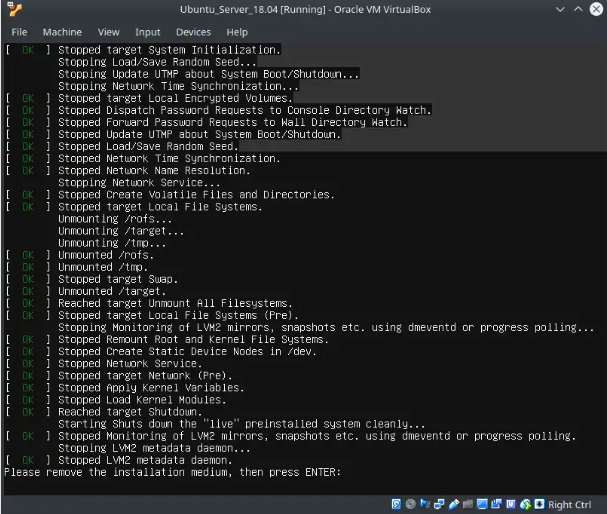
1. Select **Done** by press tab multiple times, then press **Enter.**



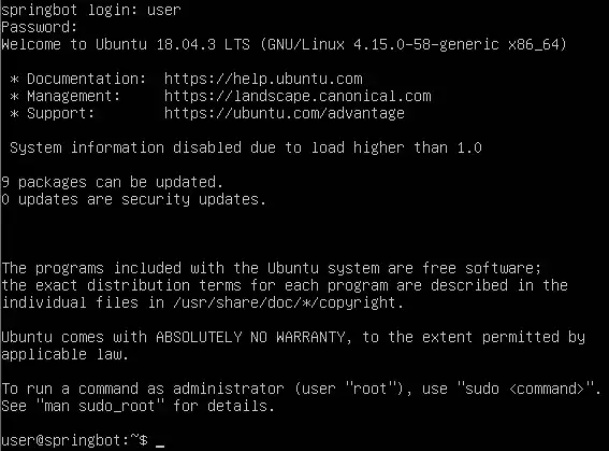
1. When the installation finishes, select **“Reboot Now”** and press **Enter.** Note that if you’re using a laptop using battery power, the update may fail, so make sure to plug it into a power source.



1. You’ll then be prompted to remove the installation media before the reboot. Press **Enter**.



1. Using the credentials from step 38, login to the server



1. Update the OS by entering the following commands.

sudo rm -rf /var/lib/apt/lists/\*

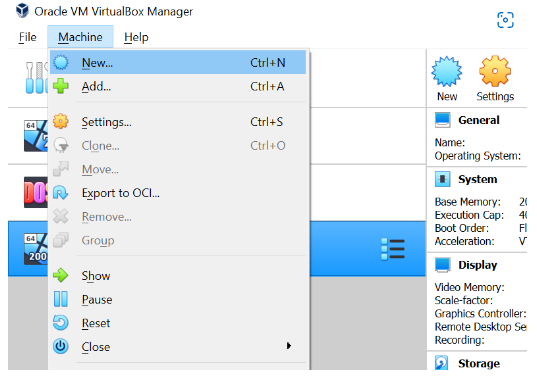
sudo apt-get update

sudo apt-get upgrade

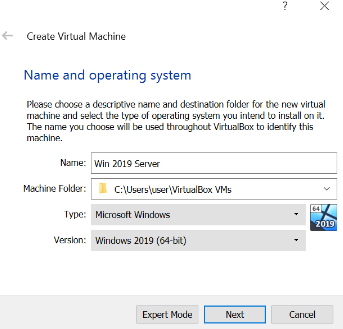
Creating a Windows server virtual machine

1. The process for creating a Windows server virtual machine is the same as for TACACS+. But here are the steps incase you need a refresher.

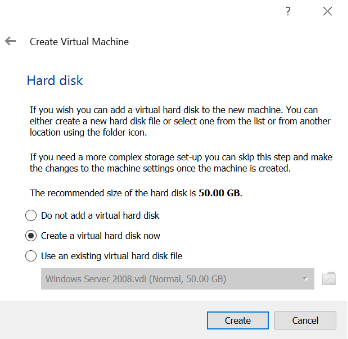
Open VirtualBox. Click the **Machine** button at the top and click **New.** This will create a new machine.



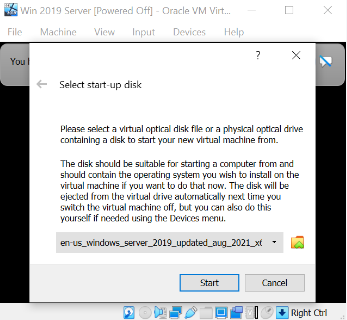
1. Name the Virtual Machine and select a version. We will be using **Windows 2019 (64-bit).**



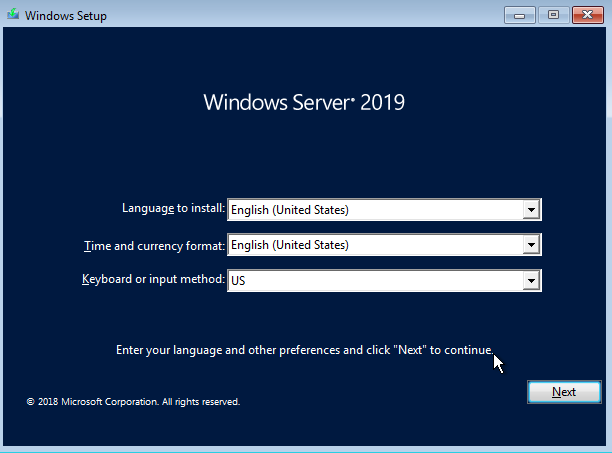
1. Create a Disk Drive and allocate it space. 50 GB should be fine.



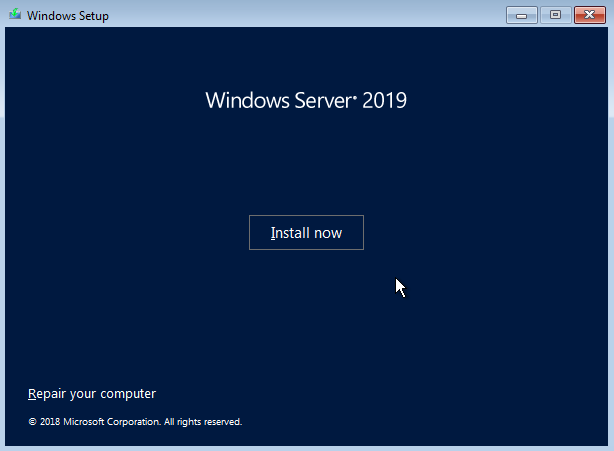
1. Launch the VM once the disk is created. When prompted, click the dropdown and find the **Window Server ISO file location.** Click **start.**



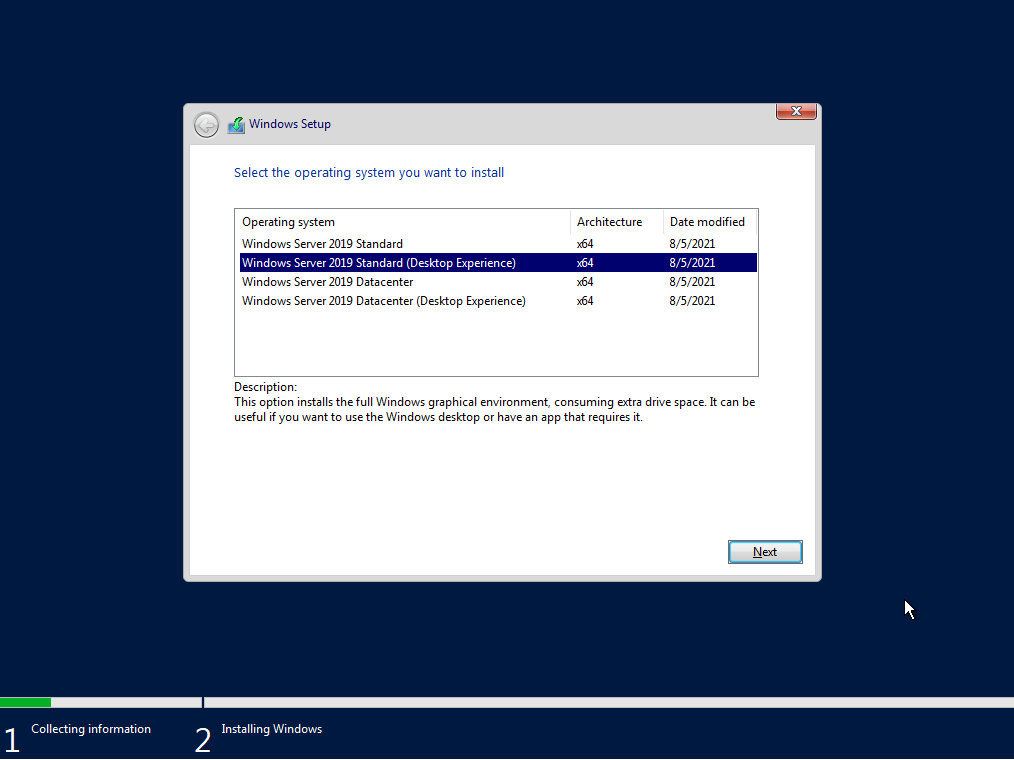
1. Set your language preferences.



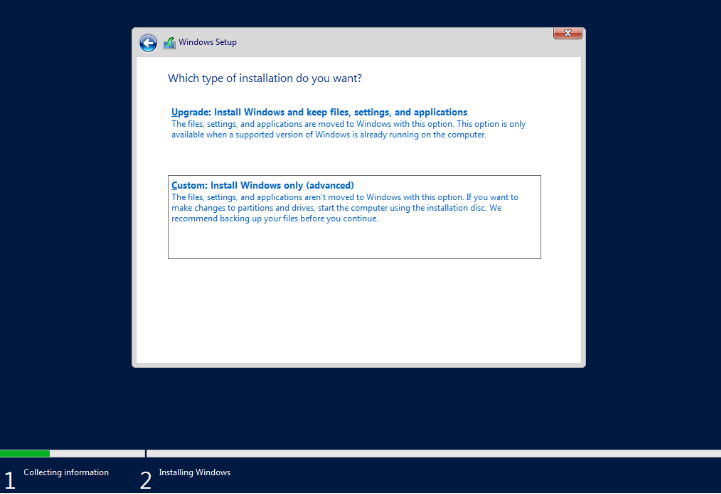
1. Click **Install Now.**



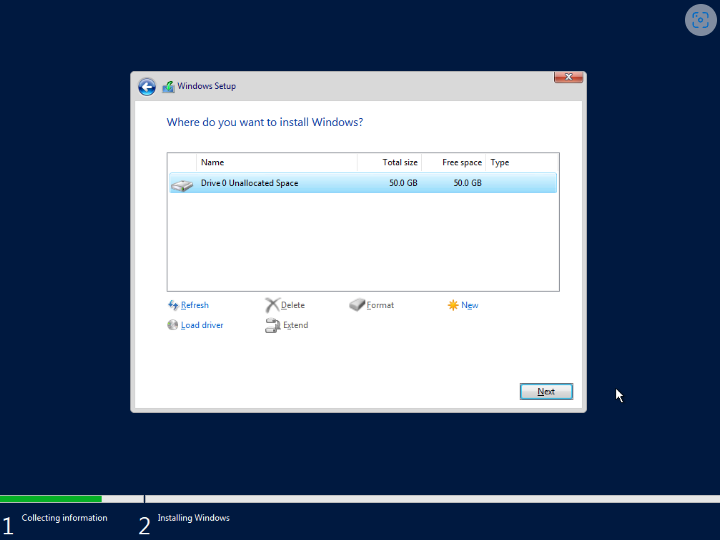
1. Select the operating system you want to install. I used **Windows Server 2019 Standard (Desktop Experience).** Click **next.**



1. Select **Custom: Install Windows only (advanced).**

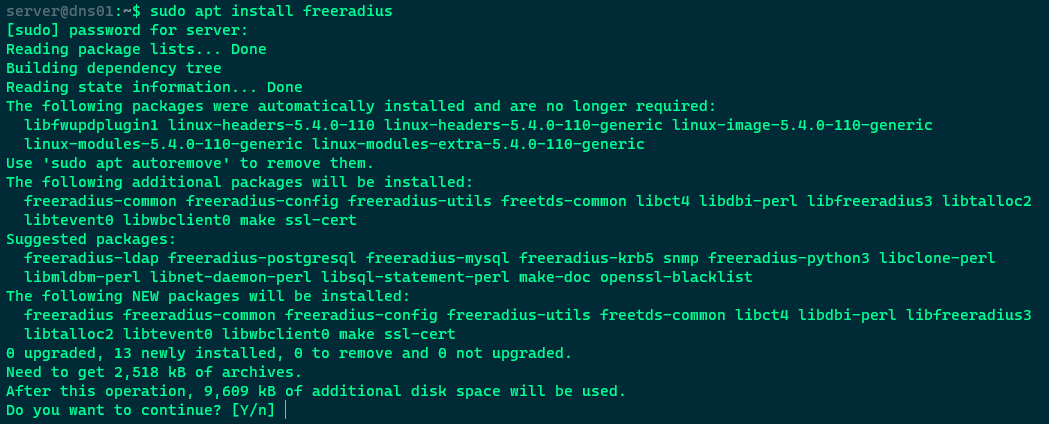


1. Select the available partition and click **Next.** Your windows server should be installed.



Install and configure Freeradius on Ubuntu

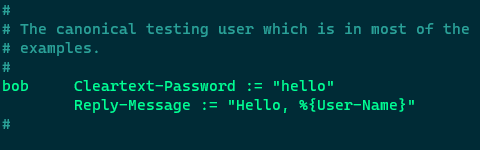
1. Log into your virtual machine
2. Type the **sudo apt install freeradius** command
3. Enter the password you set in step 10



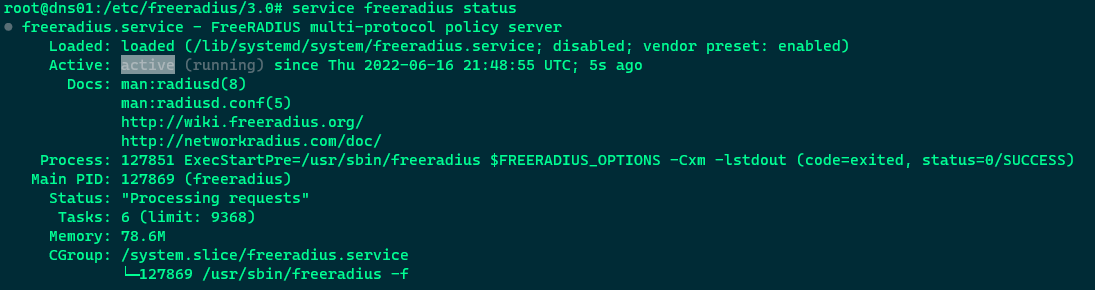
1. Type **y** and press **enter.** Freeradius should now be installed
2. To add users to the radius server, type **sudo su** into the terminal, then enter your password. Since the version number is 3.0, and we need it to be x.x, we need to change the version. To do this, enter **cd /etc/freeradius/x.x/.**



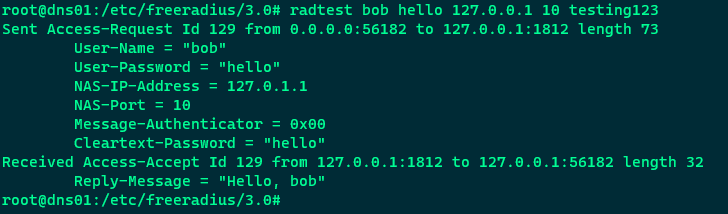
1. Next, type **nano users** to edit the user database file.
2. Scroll down to the user bob, uncomment it, and edit it to the username and password you want.



1. Press **ctrl+x** then enter **y** to save and exit the user file.
2. To restart freeradius, type **service freeradius restart**
3. Once it finishes restarting, check the status of freeradius. To do this, type **service freeradius status.** It should say **active**



1. To test if your RADIUS server is working, use the **radtest** command. An example of this command follows: **radtest [username] [password] [servername] [port] [secret].**



Configuring RADIUS on your Cisco Router

1. To define a new AAA model, set a fail message and a banner, state we want radius to act as the default login server, and set the enable mode to use our radius server, enter the following commands:

**aaa new-model**

**aaa authentication attempts login 5**

**aaa authentication banner `Unauthorized Access is Prohibited!.`**

**aaa authentication fail-message `Try again.`**

**aaa authentication login default group radius**

**aaa authentication enable default group radius**

1. Now, define the radius server that the router should use. The IP address should be the IP address of the Virtual Machine. To do this, enter the following commands:

**radius server mainlinuxserver**

**address ipv4 <ip> auth-port 1812 acct-port 1813**

**timeout 30**

**retransmit 3**

**key secretkey**

**Lab Summary**

In this lab, I used one 4321 Cisco Router, one HP ProLiant Dl360 G7 server, one copper-straight through cables, and one roll-over cable. I connected the server and the router using a copper-straight through cable going from the server to the g0/0/1 interface in the router. After plugging those in, using DHCP, I assigned an IP address to the server, and manually assigned an IP address to the router. Next, I configured TACACS+ on the router and server. Finally, I pinged both my routers to verify connectivity, and did other show commands like **show tacacs** to ensure that TACACS+ was working on both the server and the router. After making sure TACACS+ was working, I went into User EXEC mode and entered the username and password I created to ensure that it worked.

**Lab Commands**

Router(config)**#aaa new-model**

This command enables AAA on your Cisco device

Router(config)**#aaa authentication login default group tacacs+**

This command authenticates the default login group using all configured TACACS+ servers.

Router(config)**# aaa authentication enable default group tacacs+**

This command enables AAA authentication to determine if a user can access the privileged command level

Router(config)**# aaa authentication [banner/fail-message] “message”**

This command creates a banner/fail-message for the router login interface

This command enables AAA authentication to determine if a user can access the privileged

Router(config)**#tacacs/radius server <name>**

This command creates a TACACS/RADIUS server name and enters TACACS/RADIUS server configuration mode

Router(config-server-tacacs/radius)**#key <Server Key>**

This command is used to set authentication and encryption. It must match the key used in the TACACS+/RADIUS setup.

Router(config-server-tacacs)**#address ipv4 <ip-address>**

This command specifies the ipv4 address in TACACS+.

Router(config-server-radius)**#address ipv4 <ip-address> auth-port [port number] acct-port [port number]**

This command specifies the ipv4 address in RADIUS, and also assigns authentication port numbers. For RADIUS, using port 1812 and 1813 for “auth” and “acct” respectively is best practice.

Router(config-radius)**#retransmit [number of retransmits]**

This command indicates the number of retry attempts allowed before failure is determined

Router(config-radius)**#timeout [number of seconds]**

This command indicates the number of seconds of requesting before a retry transmission is determined

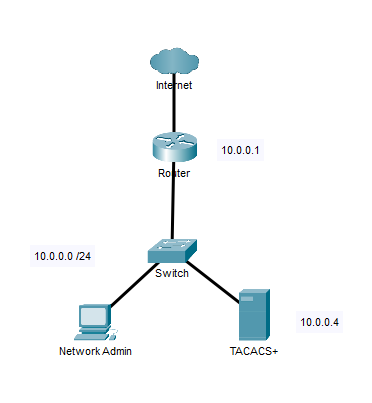
Router(config-if)**#show tacacs**

This command displays the TACACS+ configuration and status

Router(config-if)**#show radius**

This command displays the RADIUS configuration and status

**Network Diagram with IP's**



|  |  |  |
| --- | --- | --- |
| **Device** | **Interface** | **IP Address** |
| R1 | G0/0/1 | 10.0.0.1 /24 |
| Server | Eth0 | 10.0.0.4 /24 | |

**Configurations**

**Router 1**

**TACACS+ Configuration**

**show run**

Router#show run

Building configuration...

Current configuration : 1709 bytes

Last configuration change at 16:06:43 UTC Fri Jun 3 2022

version 15.5

service timestamps debug datetime msec

service timestamps log datetime msec

no platform punt-keepalive disable-kernel-core

hostname Router

boot-start-marker

boot-end-marker

vrf definition Mgmt-intf

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

enable secret 5 $1$TY.w$xMmC12rBVqHba99HOFG3u1

aaa new-model

aaa authentication login default group tacacs+

aaa authentication enable default group tacacs+

aaa session-id common

ip dhcp pool 1

network 10.0.0.0 255.255.255.0

default-router 10.0.0.1

subscriber templating

vtp mode transparent

multilink bundle-name authenticated

license udi pid ISR4321/K9 sn FDO210907U3

spanning-tree extend system-id

username admin secret 5 $1$SMba$Ty1CL7lVFvU18mE2hDvFz.

redundancy

mode none

vlan internal allocation policy ascending

interface GigabitEthernet0/0/0

no ip address

shutdown

negotiation auto

interface GigabitEthernet0/0/1

ip address 10.0.0.1 255.255.255.0

negotiation auto

interface Serial0/1/0

interface Serial0/1/1

interface GigabitEthernet0/2/0

negotiation auto

interface GigabitEthernet0/2/1

negotiation auto

interface GigabitEthernet0

vrf forwarding Mgmt-intf

no ip address

shutdown

negotiation auto

interface Vlan1

no ip address

shutdown

ip forward-protocol nd

no ip http server

no ip http secure-server

ip tftp source-interface GigabitEthernet0

tacacs server TACACS+

address ipv4 10.0.0.4

key secret

control-plane

line con 0

stopbits 1

line aux 0

stopbits 1

line vty 0 4

end

**show tacacs**

Router#show tacacs

Tacacs+ Server - public :

Server name: tacacs+

Server address: 10.0.0.4

Server port: 49

Socket opens: 13

Socket closes: 13

Socket aborts: 0

Socket errors: 0

Socket Timeouts: 0

Failed Connect Attempts: 0

Total Packets Sent: 7

Total Packets Recv: 7

**RADIUS Configuration**

**show run**

R1#show run

Building configuration...

Current configuration : 2056 bytes

Last configuration change by bob

version 16.7

service timestamps debug datetime msec

service timestamps log datetime msec

platform qfp utilization monitor load 80

no platform punt-keepalive disable-kernel-core

hostname R1

boot-start-marker

boot-end-marker

vrf definition Mgmt-intf

address-family ipv4

exit-address-family

address-family ipv6

exit-address-family

enable secret 5 $1$drdy$3cVfwBxsTuOPryy7FYWA9/

aaa new-model

aaa authentication attempts login 5

aaa authentication banner ^CUnauthorized Access is Prohibited!^C

aaa authentication fail-message ^CTry again^C

aaa authentication login default group RADIUS enable

aaa authentication enable default group RADIUS enable

aaa session-id common

no ip domain lookup

subscriber templating

vtp domain cisco

vtp mode transparent

multilink bundle-name authenticated

license udi pid ISR4321/K9 sn FDO220523GF

no license smart enable

diagnostic bootup level minimal

spanning-tree extend system-id

redundancy

mode none

interface GigabitEthernet0/0/0

ip address 192.168.1.51 255.255.255.0

negotiation auto

interface GigabitEthernet0/0/1

no ip address

shutdown

negotiation auto

interface Serial0/1/0

no ip address

shutdown

interface Serial0/1/1

no ip address

shutdown

interface GigabitEthernet0/2/0

no ip address

shutdown

negotiation auto

interface GigabitEthernet0/2/1

no ip address

shutdown

negotiation auto

interface GigabitEthernet0

vrf forwarding Mgmt-intf

no ip address

shutdown

negotiation auto

ip forward-protocol nd

ip http server

ip http authentication local

ip http secure-server

ip tftp source-interface GigabitEthernet0

radius server mainlinuxserver

address ipv4 192.168.1.59 auth-port 1812 acct-port 1813

timeout 2

retransmit 2

key secretkey

control-plane

line con 0

transport input none

stopbits 1

line aux 0

stopbits 1

line vty 0 4

login authentication defauly

wsma agent exec

wsma agent config

wsma agent filesys

wsma agent notify

end

**show radius**

R1# show radius

Status and Counters - General RADIUS Information

Deadtime(min) : 5

Timeout(secs) : 10

Retransmit Attempts : 2

Global Encryption Key : secret

Dynamic Authorization UDP Port : 3799

Source IP Selection : Outgoing Interface

Auth Acct DM/ Time

Server IP Addr Port Port CoA Window Encryption Key OOBM

--------------- ---- ---- --- ------ -------------- ----

192.168.1.59 1812 1813 No 300 secretkey No

**Problems**

A problem I faced was logging into my router when using RADIUS. After I finished configuring everything, I attempted to login using a username and password that I set, which worked. But when I tried to create a new user with a different username and password, and I tried logging into my router, I was not getting access. To figure out why I wasn’t getting access to this new account, I opened the **users** file and after looking a few times, I realized that there was a comma where it shouldn’t have been. After removing the comma, I tried logging into that new account, and I had access to my router.

**Conclusion**

AAA is a protocol keeps your network secure by making sure that only the right users are authenticated, and that those users can only access the network resources they have permissions to, and that what they do is logged so network administrators can see what they did. It consists of two main protocols, TACACS+ and RADIUS. While going through these, messages are sent between hosts, clients, and server. To configure this, you need to use some AAA specific commands such as **aaa new-model, aaa authentication login default group tacacs+, tacacs server <name>,** and **key <Server Key>.** There are also some AAA specific show commands that are helpful to verify TACACS+/RADIUS is working correctly after configuring it, such as **show tacacs.** I was able to configure TACACS on 1 Cisco 4321 router, and one HP ProLiant Dl360 G7 server. Although I was having some problems getting RADIUS to work, I was able to troubleshoot them to get RADIUS to work. Through this lab, I learned how to configure TACACS+ and RADIUS, as well as develop a deeper understanding of everything needed to make it work.

**Teacher Signoff Page of Lab Completed**

**Evan Choi has completed this AAA Lab**

**June 15, 2022**

