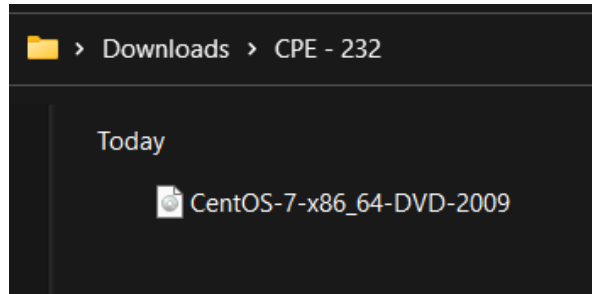


<b>Name:</b> Taller, Rose Ann A.	<b>Date Performed:</b> August 29, 2023
<b>Course/Section:</b> CPE31S4	<b>Date Submitted:</b> September 5, 2023
<b>Instructor:</b> Dr. Jonathan V. Taylor	<b>Semester and S.Y.:</b> First Sem, S.Y. 2023-2024
<b>Activity 3: Install SSH server on CentOS or RHEL 8</b>	
<b>1. Objectives:</b> 1.1 Install Community Enterprise OS or Red Hat Linux OS 1.2 Configure remote SSH connection from remote computer to CentOS/RHEL-8	
<b>2. Discussion:</b>  <b>CentOS vs. Debian: Overview</b>  CentOS and Debian are Linux distributions that spawn from opposite ends of the candle.  CentOS is a free downstream rebuild of the commercial Red Hat Enterprise Linux distribution where, in contrast, Debian is the free upstream distribution that is the base for other distributions, including the Ubuntu Linux distribution.  As with many Linux distributions, CentOS and Debian are generally more alike than different; it isn't until we dig a little deeper that we find where they branch.  <b>CentOS vs. Debian: Architecture</b>  The available supported architectures can be the determining factor as to whether a distro is a viable option or not. Debian and CentOS are both very popular for x86_64/AMD64, but what other archs are supported by each?  Both Debian and CentOS support AArch64/ARM64, armhf/armhfp, i386, ppc64el/ppc64le. (Note: armhf/armhfp and i386 are supported in CentOS 7 only.)  CentOS 7 additionally supports POWER9 while Debian and CentOS 8 do not. CentOS 7 focuses on the x86_64/AMD64 architecture with the other archs released through the AltArch SIG (Alternate Architecture Special Interest Group) with CentOS 8 supporting x86_64/AMD64, AArch64 and ppc64le equally.  Debian supports MIPSel, MIPS64el and s390x while CentOS does not. Much like CentOS 8, Debian does not favor one arch over another—all supported architectures are supported equally.  <b>CentOS vs. Debian: Package Management</b>  Most Linux distributions have some form of package manager nowadays, with some more complex and feature-rich than others.  CentOS uses the RPM package format and YUM/DNF as the package manager.  Debian uses the DEB package format and dpkg/APT as the package manager.	

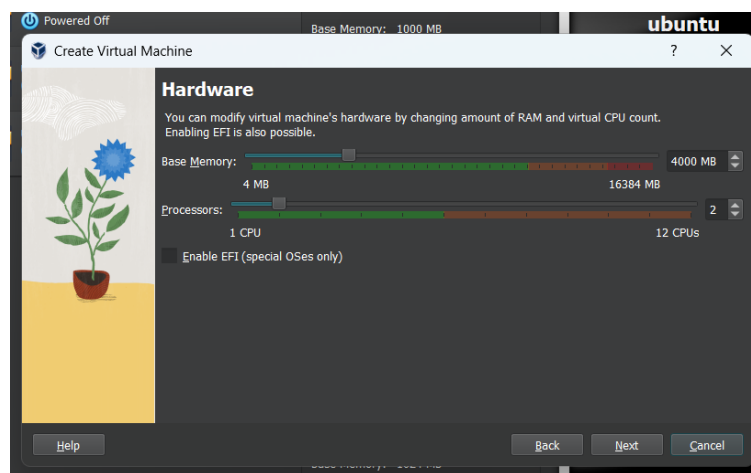
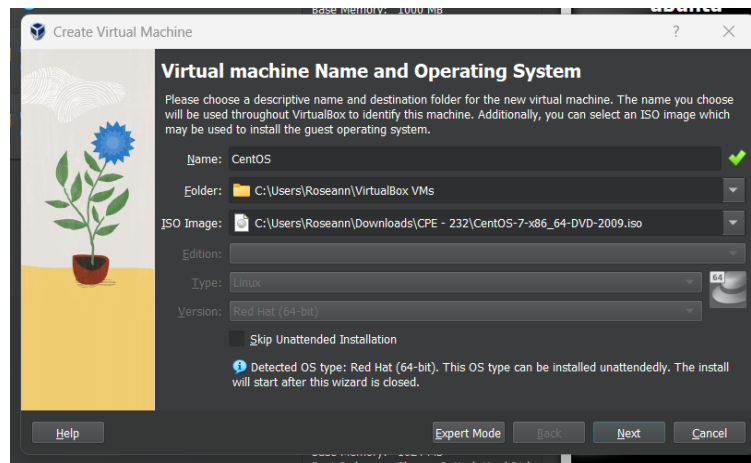
Both offer full-feature package management with network-based repository support, dependency checking and resolution, etc.. If you're familiar with one but not the other, you may have a little trouble switching over, but they're not overwhelmingly different. They both have similar features, just available through a different interface.

**Task 1: Download the CentOS or RHEL-8 image (Create screenshots of the following)**

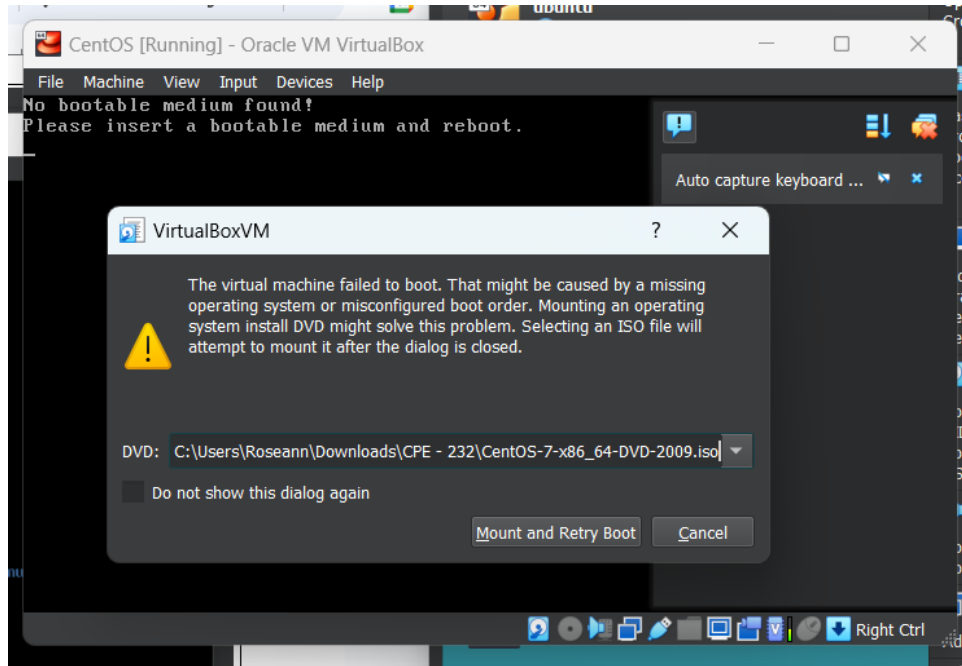
1. Download the image of the CentOS here:  
[http://mirror.rise.ph/centos/7.9.2009/isos/x86\\_64/](http://mirror.rise.ph/centos/7.9.2009/isos/x86_64/)



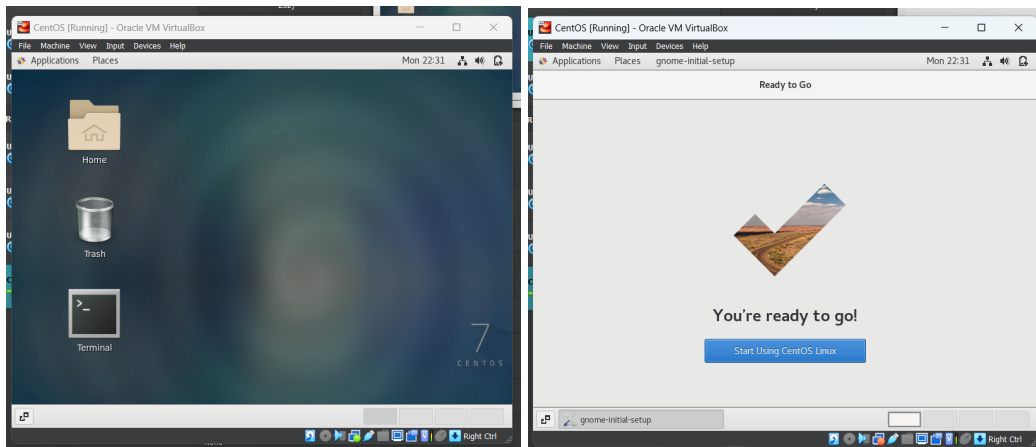
2. Create a VM machine with 2 Gb RAM and 20 Gb HD.



### 3. Install the downloaded image.



### 4. Show evidence that the OS was installed already.



## Task 2: Install the SSH server package *openssh*

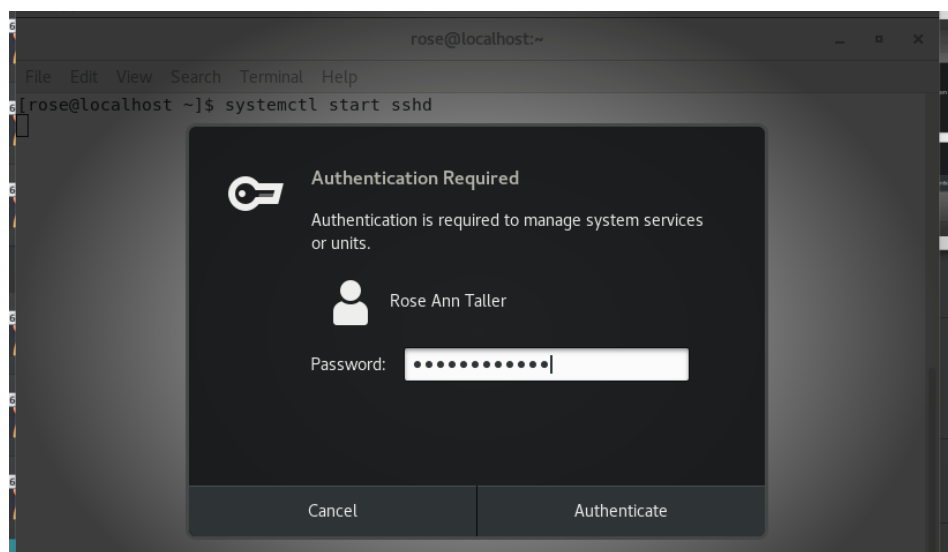
### 1. Install the ssh server package *openssh* by using the *dnf* command:

*\$ dnf install openssh-server*

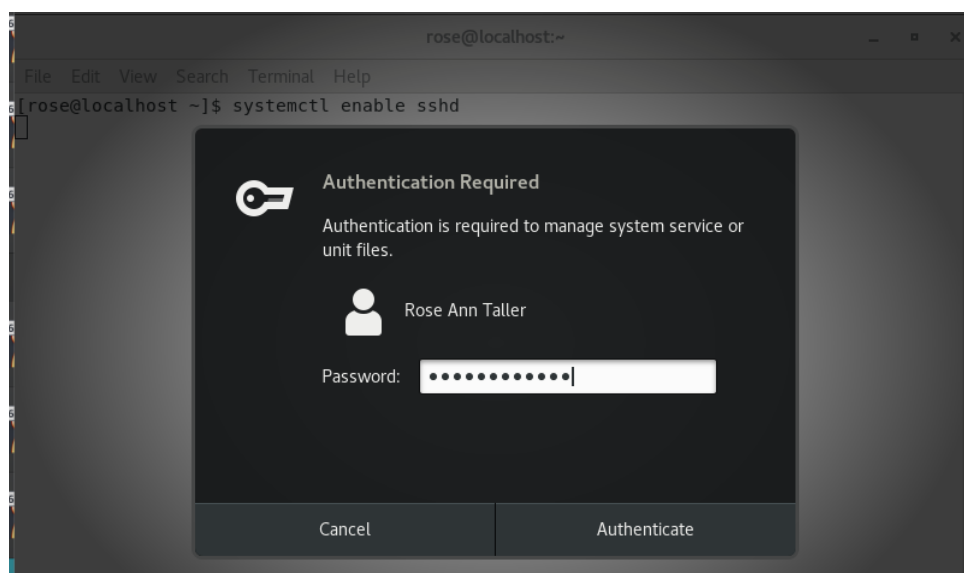
```
rose@localhost:~  
File Edit View Search Terminal Help  
[rose@localhost ~]$ sudo dnf install openssh-server  
[sudo] password for rose:  
Extra Packages for Enterprise Linux 7 - x86_64      8.9 MB/s | 17 MB   00:01  
CentOS-7 - Base                                   12 MB/s | 10 MB   00:00  
CentOS-7 - Updates                               19 MB/s | 28 MB   00:01  
CentOS-7 - Extras                                1.6 MB/s | 360 kB  00:00  
Package openssh-server-7.4p1-21.el7.x86_64 is already installed.  
Dependencies resolved.  
Nothing to do.  
Complete!  
[rose@localhost ~]$
```

2. Start the *sshd* daemon and set to start after reboot:

*\$ systemctl start sshd*

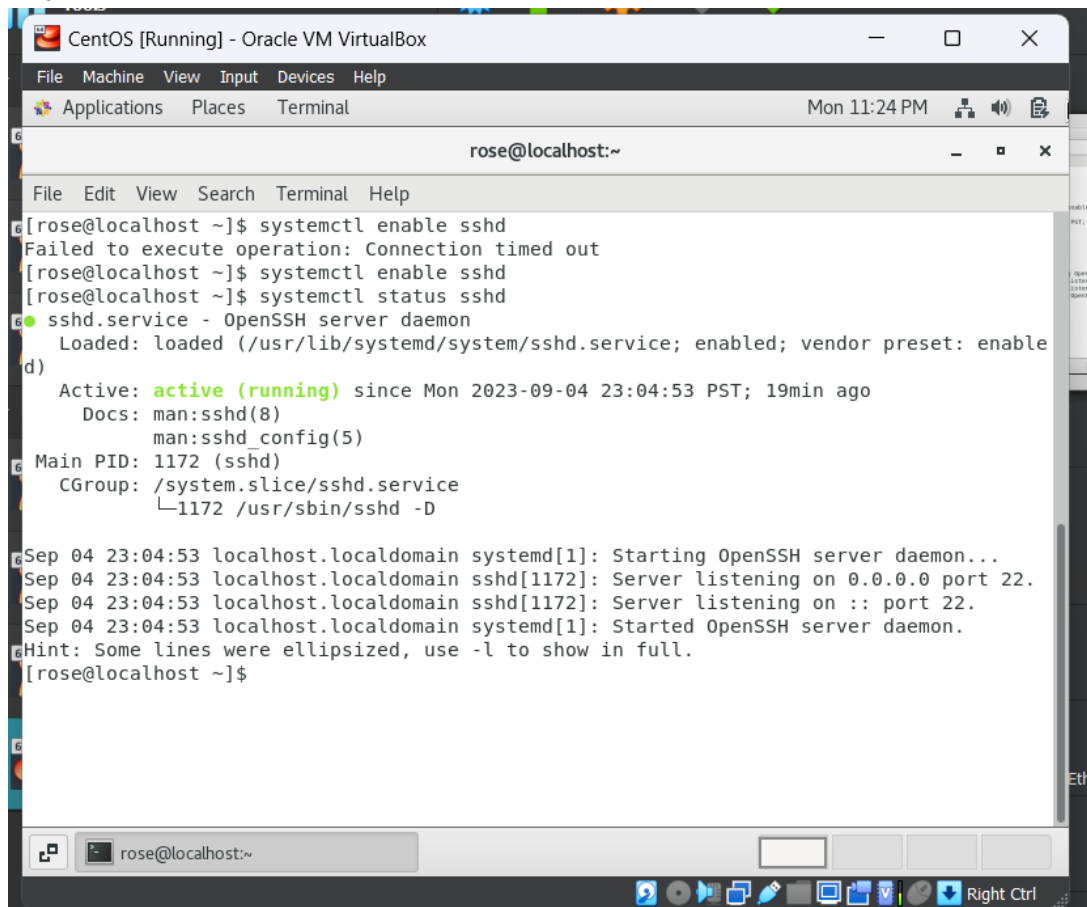


*\$ systemctl enable sshd*



3. Confirm that the sshd daemon is up and running:

*\$ systemctl status sshd*

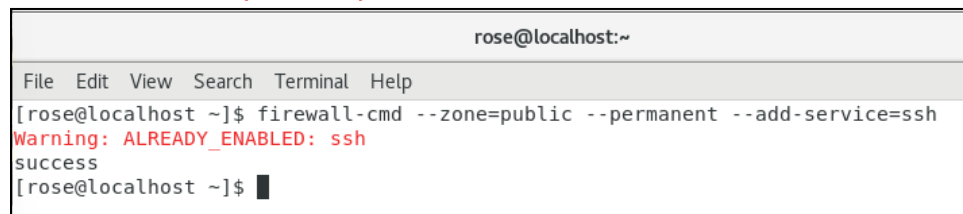


```
CentOS [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Terminal Mon 11:24 PM
rose@localhost:~
File Edit View Search Terminal Help
[rose@localhost ~]$ systemctl enable sshd
Failed to execute operation: Connection timed out
[rose@localhost ~]$ systemctl enable sshd
[rose@localhost ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enable
   Active: active (running) since Mon 2023-09-04 23:04:53 PST; 19min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 1172 (sshd)
    CGroup: /system.slice/sshd.service
            └─1172 /usr/sbin/sshd -D

Sep 04 23:04:53 localhost.localdomain systemd[1]: Starting OpenSSH server daemon...
Sep 04 23:04:53 localhost.localdomain sshd[1172]: Server listening on 0.0.0.0 port 22.
Sep 04 23:04:53 localhost.localdomain sshd[1172]: Server listening on :: port 22.
Sep 04 23:04:53 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
Hint: Some lines were ellipsized, use -l to show in full.
[rose@localhost ~]$
```

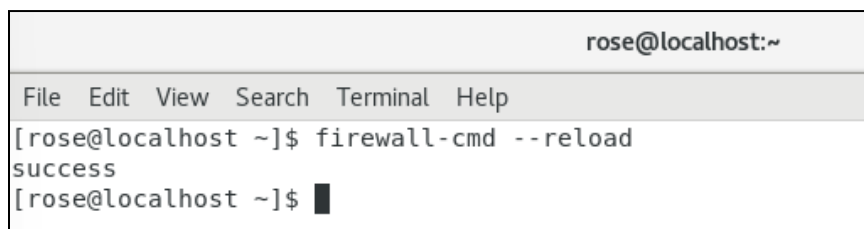
4. Open the SSH port 22 to allow incoming traffic:

*\$ firewall-cmd --zone=public --permanent --add-service=ssh*



```
rose@localhost:~
File Edit View Search Terminal Help
[rose@localhost ~]$ firewall-cmd --zone=public --permanent --add-service=ssh
Warning: ALREADY_ENABLED: ssh
success
[rose@localhost ~]$
```

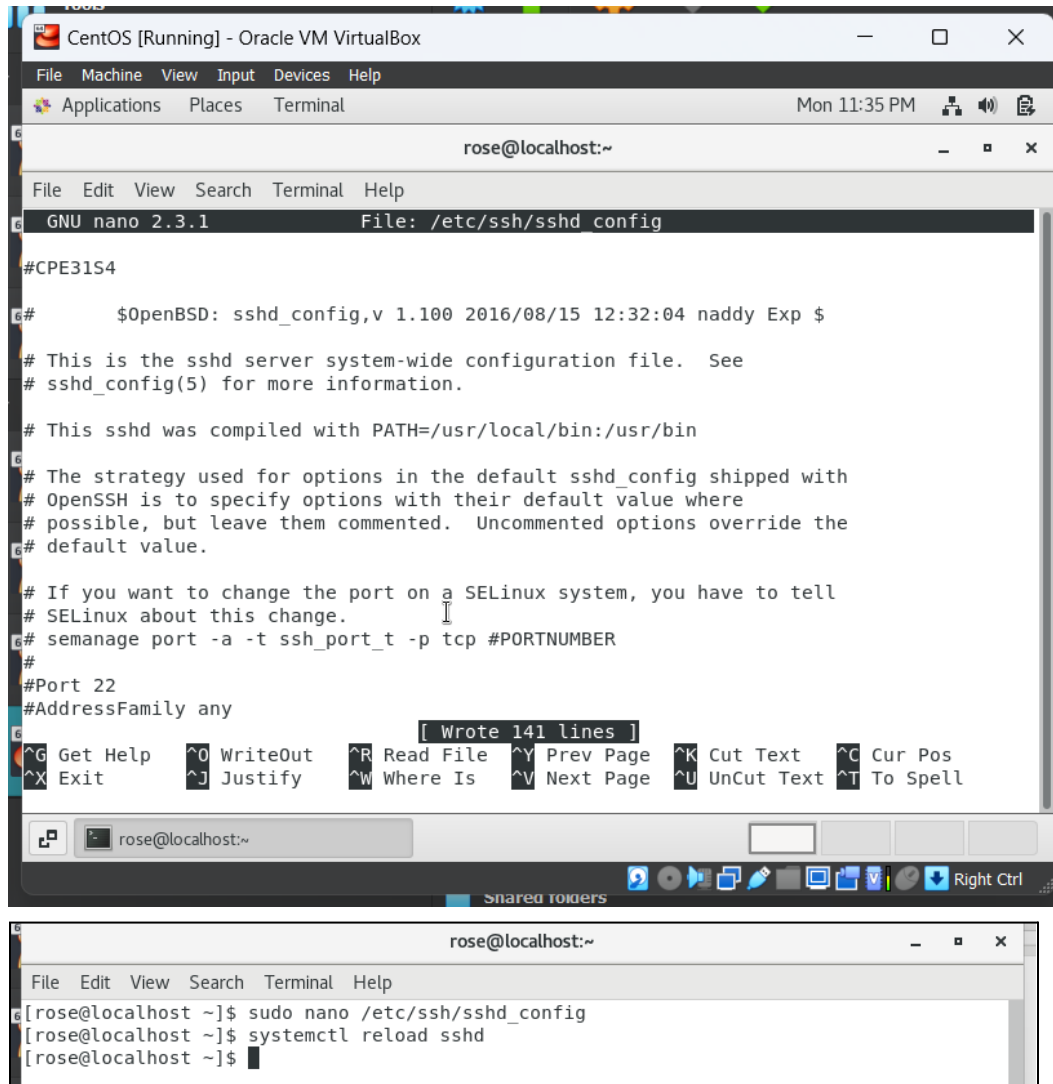
*\$ firewall-cmd --reload*



```
rose@localhost:~
File Edit View Search Terminal Help
[rose@localhost ~]$ firewall-cmd --reload
success
[rose@localhost ~]$
```

5. Locate the ssh server man config file `/etc/ssh/sshd_config` and perform custom configuration. Every time you make any change to the `/etc/ssh/sshd-config` configuration file reload the `sshd` service to apply changes:

*\$ systemctl reload sshd*



```
CentOS [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Terminal Mon 11:35 PM
rose@localhost:~
File Edit View Search Terminal Help
GNU nano 2.3.1 File: /etc/ssh/sshd_config
#CPE31S4
# $OpenBSD: sshd_config,v 1.100 2016/08/15 12:32:04 naddy Exp $
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.
# This sshd was compiled with PATH=/usr/local/bin:/usr/bin
# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.
# If you want to change the port on a SELinux system, you have to tell
# SELinux about this change.
# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER
#
#Port 22
#AddressFamily any
[ Wrote 141 lines ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
rose@localhost:~
[rose@localhost ~]$ sudo nano /etc/ssh/sshd_config
[rose@localhost ~]$ systemctl reload sshd
[rose@localhost ~]$
```

### Task 3: Copy the Public Key to CentOS

1. Make sure that `ssh` is installed on the local machine.
2. Using the command `ssh-copy-id`, connect your local machine to CentOS.

```
CentOS [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Terminal Mon 11:50 PM
rose@localhost:~

[rose@localhost ~]$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::afe5:ac50:a9b6:c3f5 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:0f:35:3c txqueuelen 1000 (Ethernet)
    RX packets 52510 bytes 75316447 (71.8 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5568 bytes 351023 (342.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.105 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::ba8e:ae50:6dda:4071 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:74:6d:af txqueuelen 1000 (Ethernet)
    RX packets 325 bytes 47940 (46.8 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 62 bytes 9772 (9.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

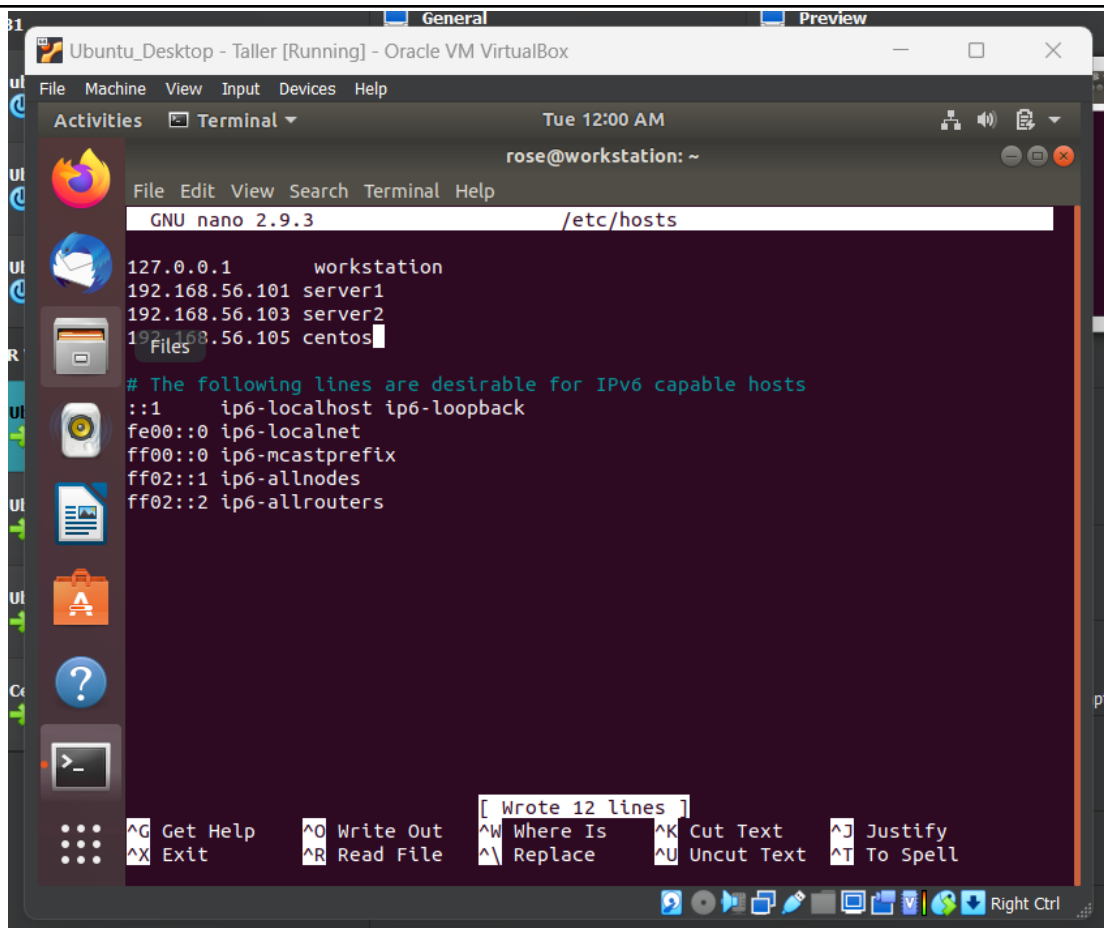
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 60 bytes 5100 (4.9 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 60 bytes 5100 (4.9 KiB)
```

```
Ubuntu_Desktop - Taller [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Mon 11:55 PM
rose@workstation: ~

rose@workstation:~$ ssh-copy-id -i ~/.ssh/id_rsa rose@192.168.56.105
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/rose/.ssh/id_rsa.pub"
The authenticity of host '192.168.56.105 (192.168.56.105)' can't be established.
ECDSA key fingerprint is SHA256:4T+liHAREvLSbc4PLPV8KykvkE8ihqa2VzlDZJcvi5g.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are promp
ted now it is to install the new keys
rose@192.168.56.105's password:
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'rose@192.168.56.105'"
and check to make sure that only the key(s) you wanted were added.
```

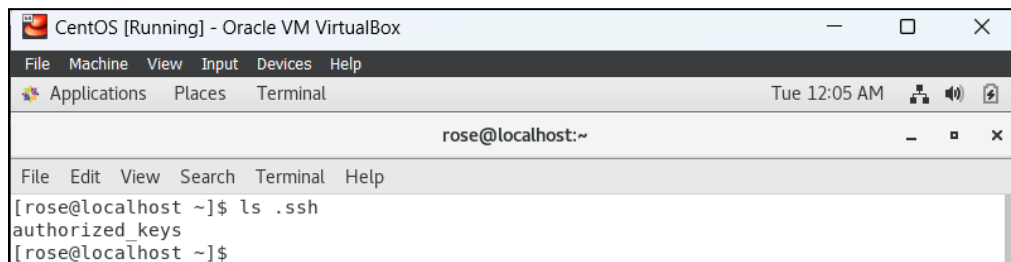
```
rose@workstation:~$ ssh rose@192.168.56.105
Last login: Mon Sep 4 23:05:14 2023
[rose@localhost ~]$
```



```
GNU nano 2.9.3 /etc/hosts
127.0.0.1 workstation
192.168.56.101 server1
192.168.56.103 server2
192.168.56.105 centos

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

3. On CentOS, verify that you have the *authorized\_keys*.

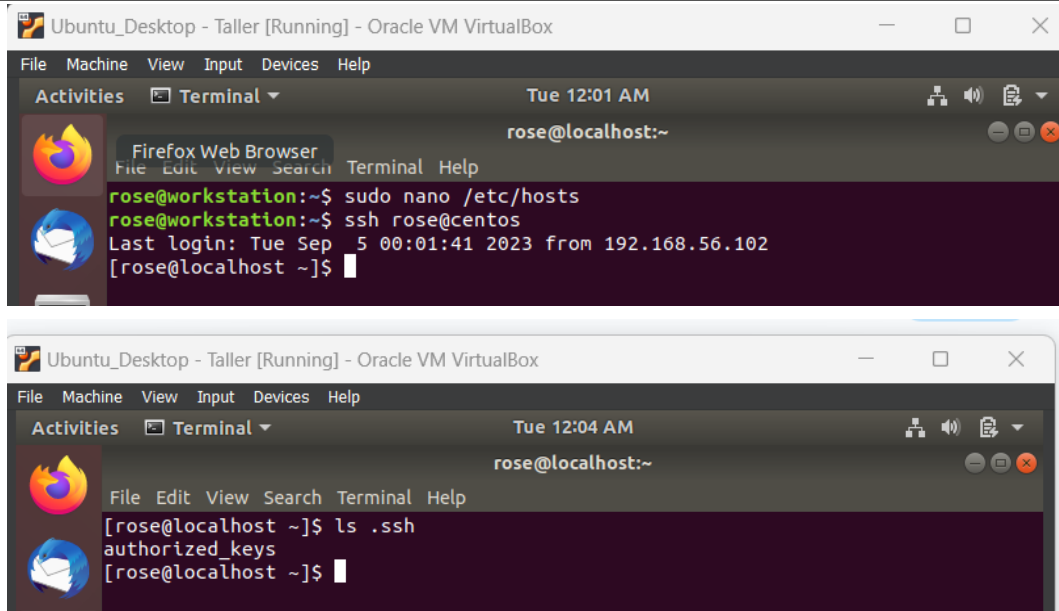


```
CentOS [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Terminal Tue 12:05 AM
rose@localhost: ~
File Edit View Search Terminal Help
[rose@localhost ~]$ ls .ssh
authorized_keys
[rose@localhost ~]$
```

#### Task 4: Verify ssh remote connection

1. Using your local machine, connect to CentOS using ssh.
2. Show evidence that you are connected.





### Reflections:

Answer the following:

1. What do you think we should look for in choosing the best distribution between Debian and Red Hat Linux distributions?
  - *As a student, I think we should look for a Linux distribution that's easy to use, works well with other software, doesn't crash, is dependable, and, most importantly, keeps our stuff safe. Learning becomes faster when the distribution we use has a simple interface that's not confusing. It should also be compatible and provide quality performance.*
2. What are the main differences between Debian and Red Hat Linux distributions?
  - *Debian uses Debian Package Management System (DPKG) and Advanced Packaging Tool (APT) package management while Red Hat uses RedHat Package Manager (RPM) and Yellowdog Updater, Modified (YUM) package management. And based on what I've learned and understood, Debian is a huge community project where volunteers from different places work for its development while Red Hat is like a project which is developed and supported by a company.*

### Conclusions/Learnings:

*In this activity, I easily finished Task 1 because the procedure was straightforward. For Task 2, the **dnf** command didn't work when I first tried it, so I had to install and set up the epel repository using the **yum install epel-release** command before I could install **dnf** using **yum install dnf**. I did all this as the root user (superuser).*

*When it came to Task 3, I completed it quickly because I was already familiar with the procedures I needed to follow to finish the task. Lastly, I enjoyed Task 4 the most*

*because it was simply about connecting to the local host in CentOS and verifying the connection.*

*Overall, I had a good time with this activity, and I've learned how to install and configure remote ssh connections from localhost/workstation to CentOS. I also learned why there are different Linux distributions and understood their purposes better.*

### **Honor Pledge for Graded Activity**

*"I affirm that I shall not give and receive any unauthorized help on this activity, and that this work is my own."*