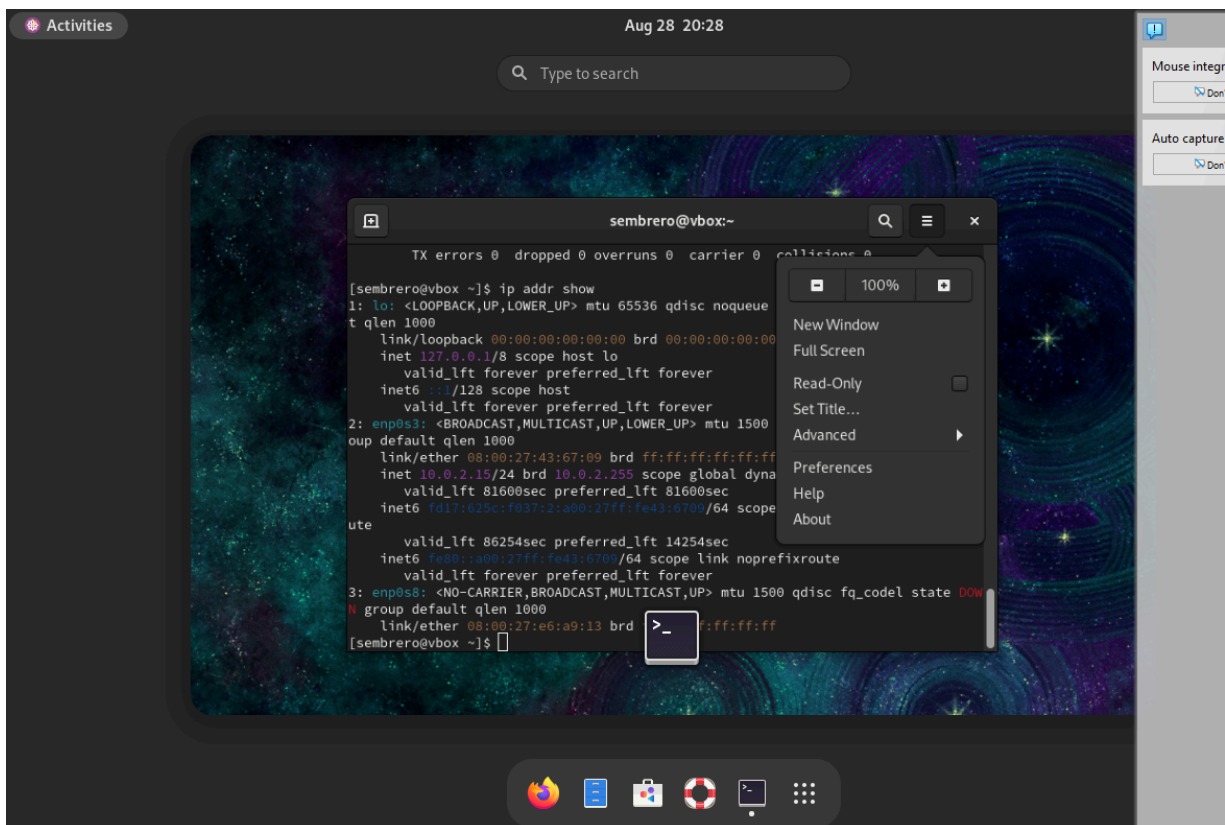


<b>Name: Sembrero, Christian Dane D.</b>	<b>Date Performed: 08/29/25</b>
<b>Course/Section: CPE 212-CPE31S2</b>	<b>Date Submitted: 08/29/25</b>
<b>Instructor: Engr. Robin Valenzuela</b>	<b>Semester and SY: 1st sem 2025-2026</b>
<b>Activity 3: Install SSH server on CentOS or RHEL 8</b>	
<b>1. Objectives:</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*

```

REDHAT_SUPPORT_PRODUCT="Red Hat Enterprise Linux 9"
REDHAT_SUPPORT_PRODUCT_VERSION="CentOS Stream"
[semlbrero@vbox ~]$ sudo dnf install -y openssh-server

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for semlbrero:
semlbrero is not in the sudoers file. This incident will be reported.
[semlbrero@vbox ~]$ su
Password:
[root@vbox semlbrero]# sudo dnf install -y openssh-server
CentOS Stream 9 - BaseOS                               452 kB/s | 8.8 MB      00:19
CentOS Stream 9 - AppStream                             1.9 MB/s | 25 MB      00:13
CentOS Stream 9 - Extras packages                       3.2 kB/s | 19 kB      00:05
Package openssh-server-8.7p1-46.el9.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@vbox semlbrero]#

```

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

```

```

[root@vbox semlbrero]# systemctl start sshd
[root@vbox semlbrero]# systemctl enable sshd
[root@vbox semlbrero]# systemctl status sshd
• sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: ena>
   Active: active (running) since Thu 2025-08-28 18:56:23 PST; 1h 0min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 930 (sshd)
     Tasks: 1 (limit: 10505)
    Memory: 2.8M (peak: 3.0M)
       CPU: 19ms
    CGroup: /system.slice/ssh.service
            └─930 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Aug 28 18:56:23 localhost.localdomain systemd[1]: Starting OpenSSH server daemon>
Aug 28 18:56:23 vbox sshd[930]: Server listening on 0.0.0.0 port 22.
Aug 28 18:56:23 vbox sshd[930]: Server listening on :: port 22.
Aug 28 18:56:23 vbox systemd[1]: Started OpenSSH server daemon.
lines 1-16/16 (END)

```

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

```
$ firewall-cmd --zone=public --permanent --add-service=ssh
$ firewall-cmd --reload
```

```
[root@vbox sembrero]# sudo firewall-cmd --zone=public --permanent --add-service=ssh
Warning: ALREADY_ENABLED: ssh
success
[root@vbox sembrero]# sudo firewall-cmd --reload
success
[root@vbox sembrero]#
```

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
```

```
GNU nano 5.6.1 /etc/ssh/sshd_config
# $OpenBSD: sshd_config,v 1.104 2021/07/02 05:11:21 dtucker 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:/usr/local/sbin:/u

# 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.

# To modify the system-wide sshd configuration, create a *.conf file under
# /etc/ssh/sshd_config.d/ which will be automatically included below
Include /etc/ssh/sshd_config.d/*.conf

# 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
```

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

1. Make sure that `ssh` is installed on the local machine.

```

[root@vbox sembrero]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:rE3Bcrs13w3YSC09FeL3VQrZtefStWhV3P0c3vLzVno root@vbox
The key's randomart image is:
+---[RSA 3072]-----+
|          . oo o=|
|          . o ..o.o*|
|          . = = o .+B|
|          + = 0 .o=*|
|          S = oo+.+|
|          + = ..o.oo|
|          . o . . .+.|
|                  ..E|
|                  o+|
+-----[SHA256]-----+
[root@vbox sembrero]# █

```

2. Using the command `ssh-copy-id`, connect your local machine to CentOS.
3. On CentOS, verify that you have the `authorized_keys`.

#### 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?

In my opinion, when deciding on Debian and Red Hat we need to first look at the end goal. If you are mainly learning about Linux and want quick and easy access to lots of packages for installation, Debian or Ubuntu is probably the better option! However, if you have plans to use this in a business setting or on a server, where stability and security is more important, then I would suggest Red Hat variety (or CentOS).

2. What are the main difference between Debian and Red Hat Linux distributions?

The main differentiator I notice between the two is their package management. Debian uses apt for package management while Red Hat uses yum, or dnf. Also, generally speaking, Debian tends to be more community based support, while Red

Hat is business-based support. Both are good options, but ultimately depends on what the user is attempting to accomplish.