Name: Bacong, El Cid A.	**Date Performed:** 9/5/2025
Course/Section: CPE212 - CPE31S4	**Date Submitted:** 9/5/2025
Instructor: Engr. Robin Valenzuela	**Semester and SY:** 1st Sem 2025-2026

Activity 3: Install SSH server on CentOS or RHEL 8

1. **Objectives:**

1.1 Install Community Enterprise OS or Red Hat Linux OS

1.2 Configure remote SSH connection from remote computer to CentOS/RHEL-8

2. **Discussion:**

CentOS vs. Debian: Overview

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.

CentOS vs. Debian: Architecture

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.

CentOS vs. Debian: Package Management

Most Linux distributions have some form of package manager nowadays, with some more complex and feature-rich than others.

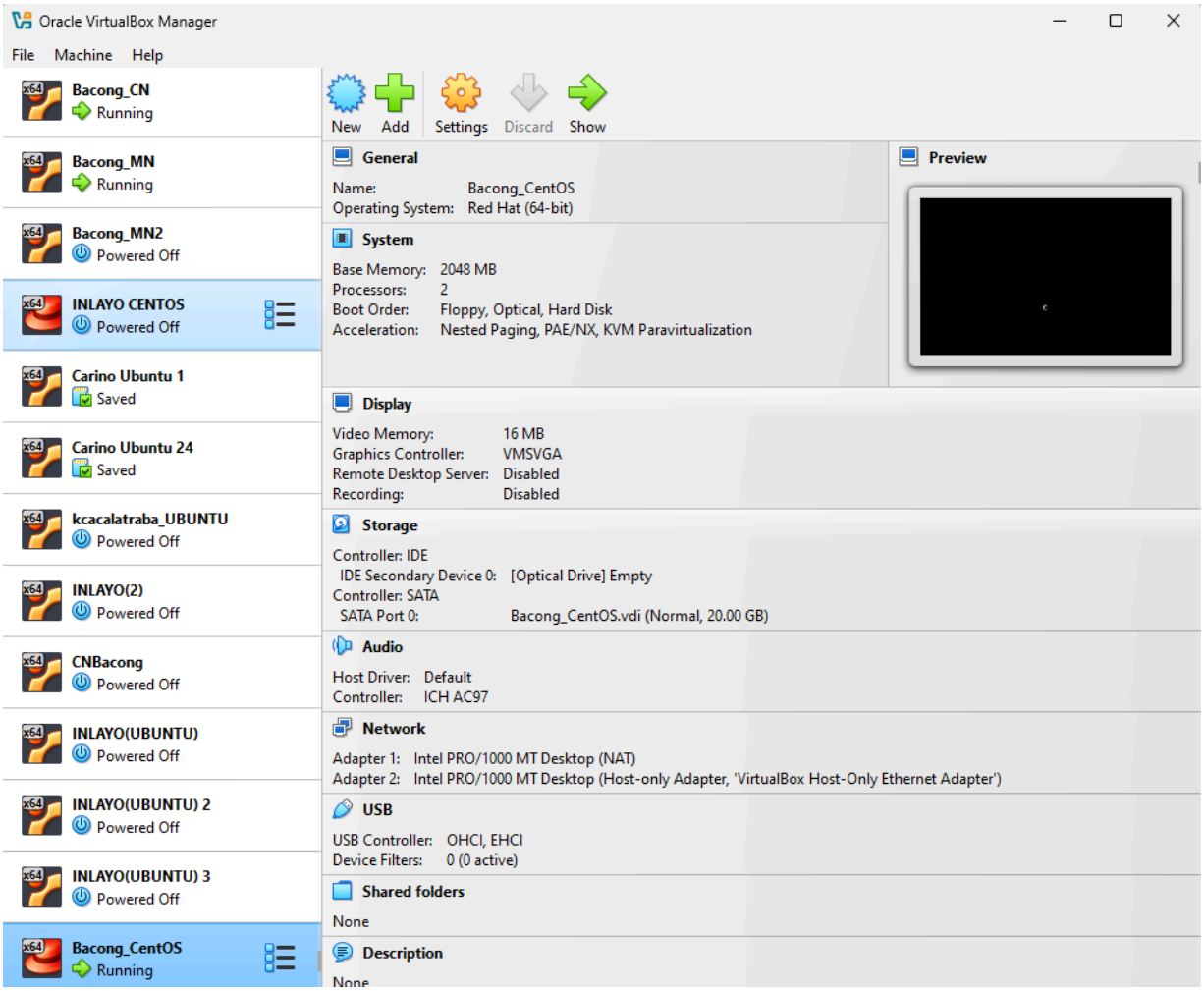
CentOS uses the RPM package format and YUM/DNF as the package manager.

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.

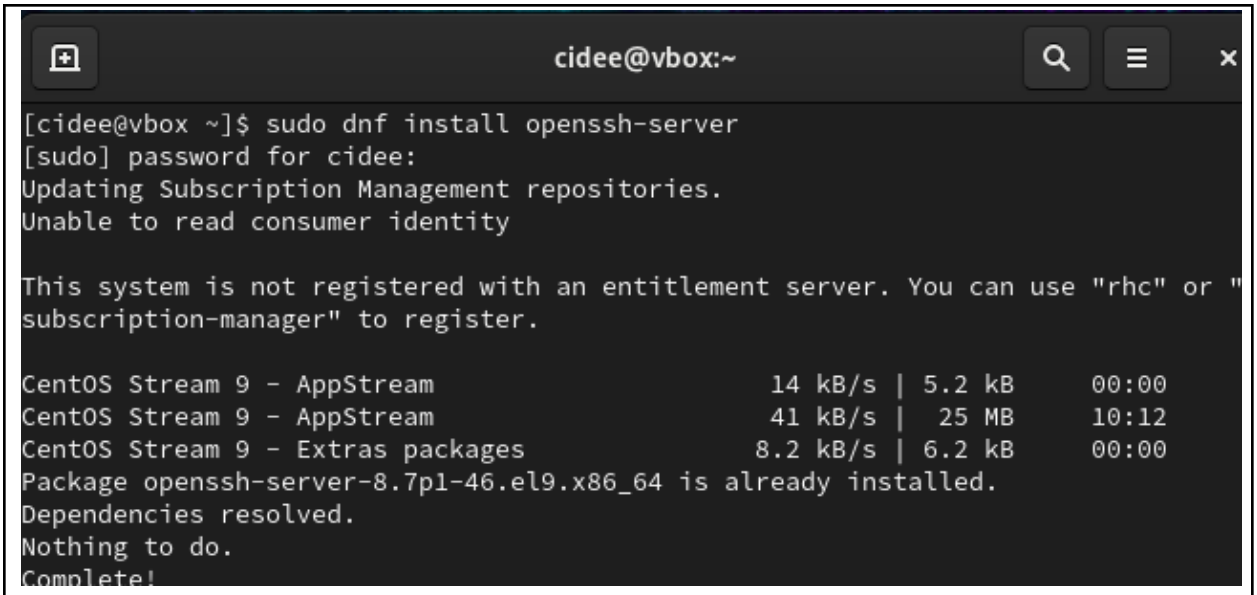
Task 1: Download the CentOS or RHEL-8 image (Create screenshots of the following)

1. Download the image of the CentOS here: http://mirror.rise.ph/centos/7.9.2009/isos/x86_64/
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.

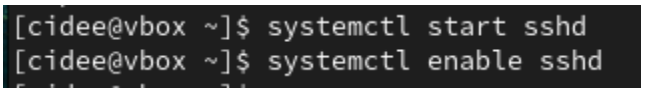


Task 2: Install the SSH server package *openssh*

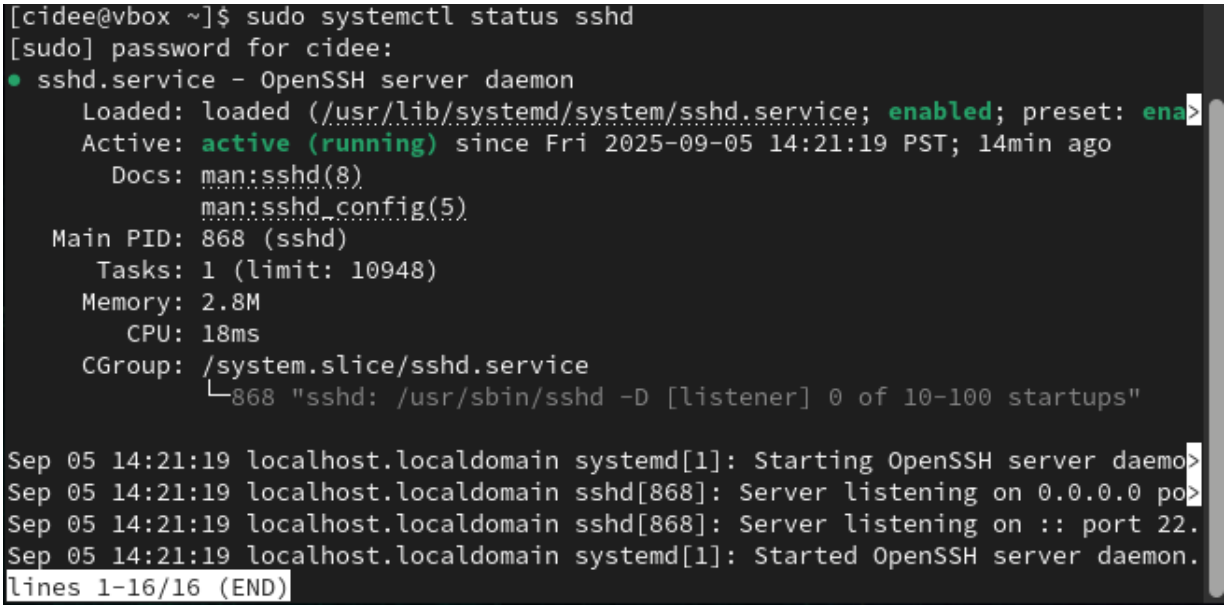
1. Install the ssh server package *openssh* by using the *dnf* command:
   $ dnf install openssh-server



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



4. Open the SSH port 22 to allow incoming traffic:
   $ firewall-cmd --zone=public --permanent --add-service=ssh
   $ firewall-cmd --reload

```
[cidee@vbox ~]$ sudo firewall-cmd --zone=public --permanent --add-service=ssh
Warning: ALREADY_ENABLED: ssh
success
[cidee@vbox ~]$ sudo firewall-cmd --reload
success
```

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

```
[cidee@vbox ~]$ sudo nano /etc/ssh/sshd_config
[cidee@vbox ~]$ systemctl reload sshd
```

Task 3: Copy the Public Key to CentOS

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

```
cidee@workstation:~$ ssh -v
usage: ssh [-46AaCfGgKkMNnqsTtVvXxYy] [-b bind_address] [-c cipher_spec]
           [-D [bind_address:]port] [-E log_file] [-e escape_char]
           [-F configfile] [-I pkcs11] [-i identity_file]
           [-J [user@]host[:port]] [-L address] [-l login_name] [-m mac_spec]
           [-O ctl_cmd] [-o option] [-p port] [-Q query_option] [-R address]
           [-S ctl_path] [-W host:port] [-w local_tun[:remote_tun]]
           [user@]hostname [command]
```

2. Using the command *ssh-copy-id*, connect your local machine to CentOS.

```
cidee@workstation:~$ ssh-copy-id cidee@192.168.56.110
The authenticity of host '192.168.56.110 (192.168.56.110)' can't be established
.
ECDSA key fingerprint is SHA256:cUI09V6cds+usz90oQGaS+I4Zu8aR3OcFJ4dQLciEqI.
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
cidee@192.168.56.110's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'cidee@192.168.56.110'"
and check to make sure that only the key(s) you wanted were added.
```

3. On CentOS, verify that you have the *authorized_keys*.

```
[cidee@vbox ~]$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDdgq8eux6/JZOsUi1qEVaFjynQIL3pD7mwvpyfPqN
ZtkwjC6tcSrGpdlIY5ddsrDs3h1ASAxJNZyO+8yZYc6MJhoi0KQIb5HfhOP5yfCHoGhpqfC/Y+Us0ZE
uXhaBA2ewnrl7DY5N1dV+8j4kieDfB2dmbhCaO8sKbguALGwx2pSbfghpqq7btIvnNbIZj3JMhaO7HI
g7KsEmNrw3xZRqHtoYbKZjaCAZeXk/6vuzecYyzL4EGrSmSPBs1nv8xhAeIKPX7nOwHfmkfbKWzY2KO
JZUa1nw1Innr5rhBdnuZ/+Jesv9KSpgPhFmoos6kJmdyzdfN8hKgPk9j0bJYV7Xp cidee@workstat
ion
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCkoBI8XhWXMzKZ00mB1WGEGXmwF1D7NtL/m0fV6cF
vjlonigxdQ9Z3zzRxGozjCYKnStogEIeRw5NDuiBHDmLdtNCZFW4TT+RkhR3ha9JZkHr7jatth3QEfz
Hi074S85pWwjAe6auh15Aw1oWpGEGcqPKC9bE/7ePVECQ2XDOnBVIPQWAt6d/yIattS32feci2hQRcm
ND09TaDwSJgaBqd8abg0NdO6G8GJF9hxQgg4VfxgvDTTY1X9qayxmBREEh4rDunW5LJuWoKrCkL+a2c
FPzTxj6EAJ8hJXuir5c0mfVk7ohGAZSp15g2rM5Qwif4d1FebhMQeKjzvGd7B7cf cidee@workstat
ion
```

```
cidee@workstation:~$ ssh cidee@192.168.56.110
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Fri Sep  5 14:21:31 2025
[cidee@vbox ~]$ ls -la ~/.ssh
total 16
drwx------.  2 cidee cidee   61 Sep  5 14:42 .
drwx------. 15 cidee cidee 4096 Aug 15 16:23 ..
-rw-------.  1 cidee cidee  399 Sep  5 14:42 authorized_keys
-rw-------.  1 cidee cidee 2590 Aug 15 16:23 id_rsa
-rw-r--r--.  1 cidee cidee  564 Aug 15 16:23 id_rsa.pub
[cidee@vbox ~]$ cat !/.ssh/authorized_keys
-bash: !/.ssh/authorized_keys: event not found
[cidee@vbox ~]$ cat~/.ssh/authorized_keys
-bash: cat~/.ssh/authorized_keys: No such file or directory
[cidee@vbox ~]$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDdgq8eux6/JZOsUi1qEVaFjynQIL3pD7mwvpyfPqN
ZtkwjC6tcSrGpdlIY5ddsrDs3h1ASAxJNZyO+8yZYc6MJhoi0KQIb5HfhOP5yfCHoGhpqfC/Y+Us0ZE
uXhaBA2ewnrl7DY5N1dV+8j4kieDfB2dmbhCaO8sKbguALGwx2pSbfghpqq7btIvnNbIZj3JMhaO7HI
g7KsEmNrw3xZRqHtoYbKZjaCAZeXk/6vuzecYyzL4EGrSmSPBs1nv8xhAeIKPX7nOwHfmkfbKWzY2KO
JZUa1nw1Innr5rhBdnuZ/+Jesv9KSpgPhFmoos6kJmdyzdfN8hKgPk9j0bJYV7Xp cidee@workstat
ion
```

Task 4: Verify ssh remote connection

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

```
cidee@workstation:~$ ssh cidee@192.168.56.110
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Fri Sep  5 14:53:03 2025 from 192.168.56.104
[cidee@vbox ~]$ whoami && pwd
cidee
/home/cidee
```

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?
   For me, it comes down to what I need for a specific project. If I'm setting up a server where I want it to be super stable and never break, I go with Debian. If I'm installing an OS for my laptop and I want the latest software and hardware support, I would choose something from the Red Hat family, like Fedora. I also think about the package manager; I find apt (Debian) a bit simpler, while dnf (Red Hat) feels more modern and powerful.
2. What are the main diffence between Debian and Red Hat Linux distributions?

   The biggest difference I see is their core philosophy. Debian is like the community-driven ideal of free software, it’s stable but can feel a bit behind on the latest versions. Red Hat Enterprise Linux is the polished, corporate-backed product built for big businesses that pay for support. Fedora is its free, cutting-edge software that tests all the new tech. So for me, Debian is about stability and philosophy, while the Red Hat world is about innovation and enterprise strength.