

Name: Potestades, North Nygel G.	Date Performed: 8/15/25
Course/Section: CPE31S2	Date Submitted: 8/22/25
Instructor: Engr. Robin Valenzuela	Semester and SY: 1st Semester 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.

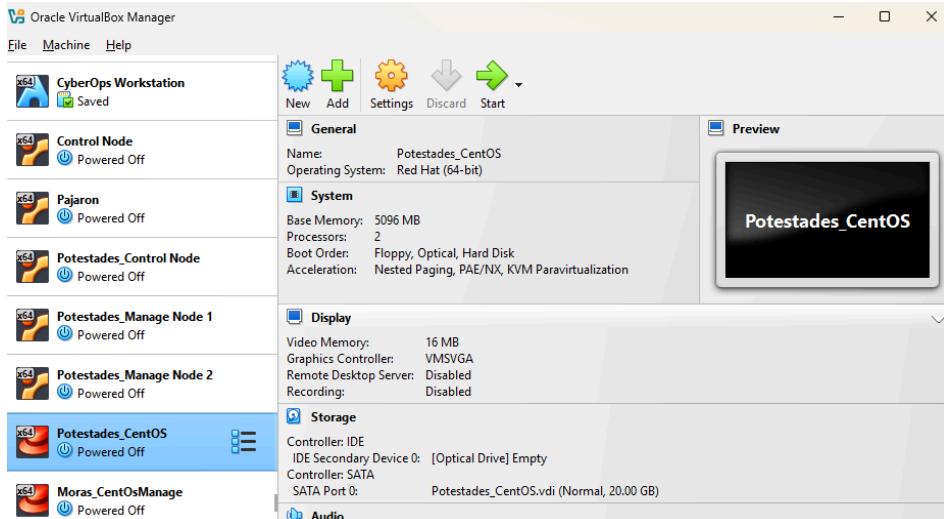
Debian uses the DEB package format and dpkg/APT as the package manager.

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
```

```
[root@vbox north]# dnf install openssh-server
Updating Subscription Management repositories.
Unable to read consumer identity
```

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

```
$ systemctl start sshd
```

```
$ systemctl enable sshd
```

```
[root@vbox north]# systemctl start sshd
[root@vbox north]# systemctl enable sshd
```

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

```
$ systemctl status sshd
```

```
[root@vbox north]# systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: enabled)
  Active: active (running) since Fri 2025-08-15 18:18:43 PST; 40s ago
    Docs: man:sshd(8)
           man:sshd_config(5)
           man:ssh(1)
           man:sshuttle(8)
```

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

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

```
$ firewall-cmd --reload
```

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

- 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
```

```
[root@vbox north]# systemctl reload sshd
```

Task 3: Copy the Public Key to CentOS

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

```
north@workstation:~$ systemctl status sshd
● ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: ena
  Active: active (running) since Fri 2025-08-15 18:23:38 +08; 47s ago
```

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

```
north@workstation:~$ ssh-copy-id -i ~/.ssh/id_rsa north@centos
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/north/.ssh
/id_rsa.pub"
/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
north@centos's password:
```

- On CentOS, verify that you have the `authorized_keys`.

Task 4: Verify ssh remote connection

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

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

Last login: Fri Aug 15 18:31:40 2025 from 192.168.56.104
[north@centos ~]$
```

Reflections:

Answer the following:

- What do you think we should look for in choosing the best distribution between Debian and Red Hat Linux distributions?

I think that we should look for criteria related to the specific task they need to perform, such as choosing Ubuntu for general use, or choosing Kali Linux for cybersecurity applications.

- What are the main differences between Debian and Red Hat Linux distributions?

Their main differences are in the package managers used, as Debian uses APT while Red Hat uses YUM, and in the fact that Debian is free and community-driven while Red Hat is a commercial product.

Conclusion:

In conclusion, this activity helped me in setting up CentOS as a managed node, setting up SSH on it and testing it out. This was a good learning experience, finally exploring a new Linux distro, as it seems to be important to know the ins and outs of different distros for different applications. Overall, this was a good learning experience, seeing that certain commands are universal to all Linux distros and that different distros can SSH into each other.