Name: Jose Mari Tan Dela Peña	Date Performed: 09/27/2024
Course/Section: CPE31S2	Date Submitted: 09/27/2024
Instructor: Engr. Robin Valenzuela	Semester and SY: 1st Sem 2024-2025

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.

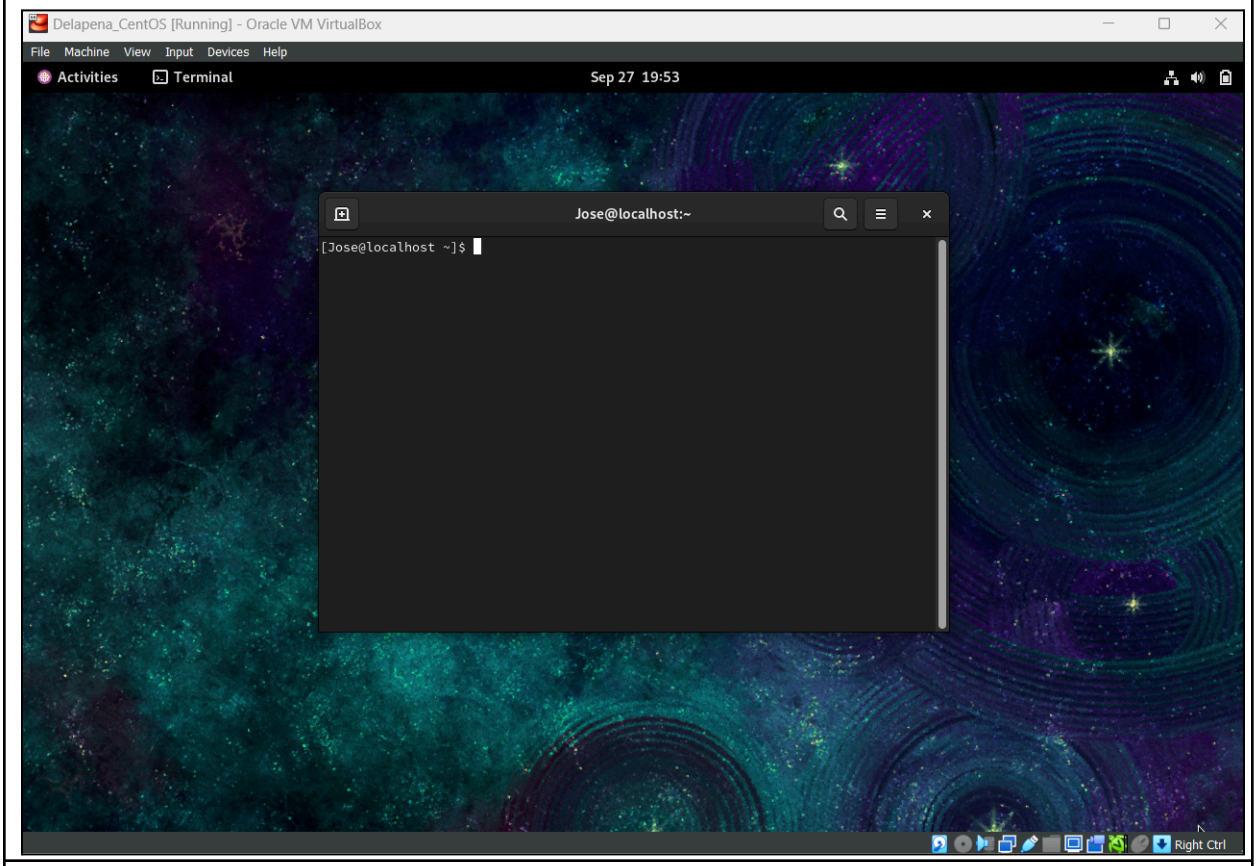
CentOS uses the RPM package format and YUM/DNF as the package manager.

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.

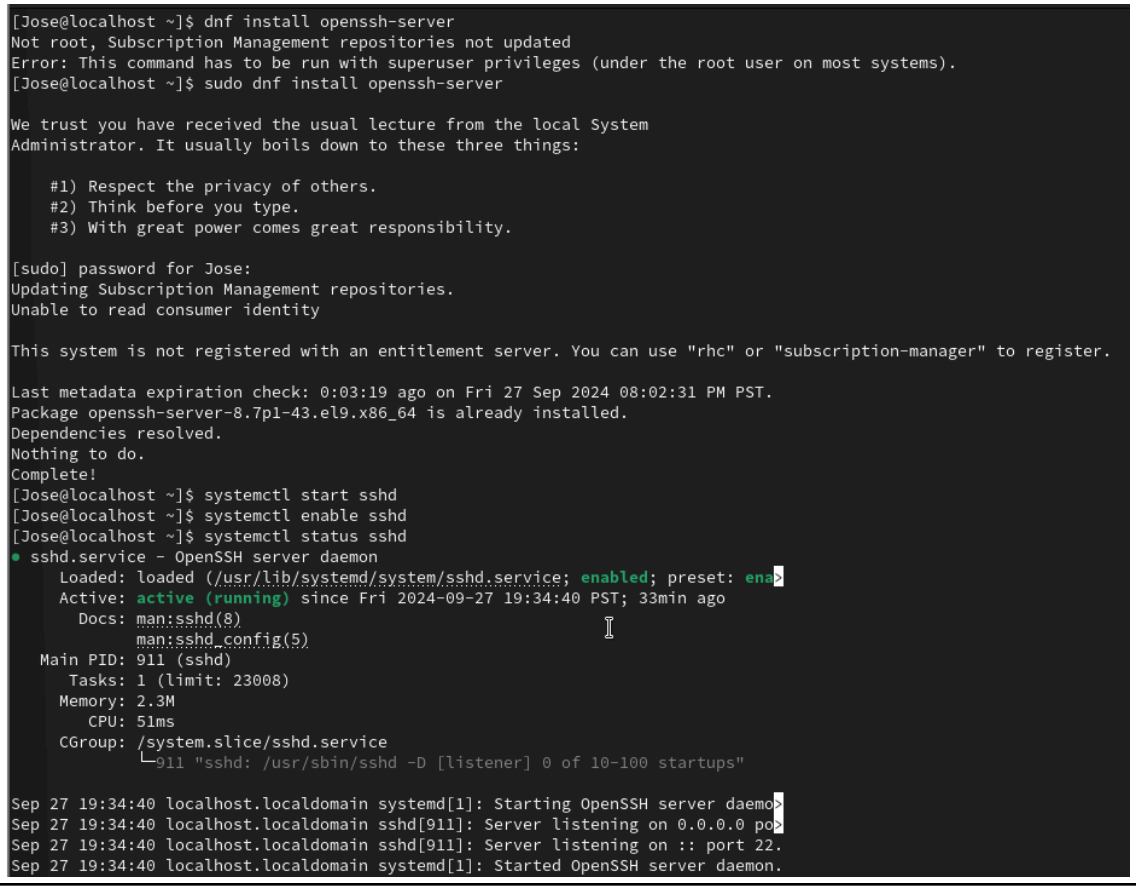


Task 2: Install the SSH server package *openssh*

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

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



```
[Jose@localhost ~]$ firewall-cmd --zone=public --permanent --add-service=ssh
Warning: ALREADY_ENABLED: ssh
success
[Jose@localhost ~]$ firewall-cmd --reload
success
[Jose@localhost ~]$ systemctl reload sshd
[Jose@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::a00:27ff:fe37:eeac  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:37:ee:ac  txqueuelen 1000  (Ethernet)
        RX packets 144425  bytes 214211367 (204.2 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 12542  bytes 795003 (776.3 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 24  bytes 2514 (2.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 24  bytes 2514 (2.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[Jose@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.

```
josemari@workstation:~$ ssh-copy-id Jose@192.168.56.13
The authenticity of host '192.168.56.13 (192.168.56.13)' can't be established.
ED25519 key fingerprint is SHA256:hjhvpPrupw34JRR2TdVZ6LCmdXvVNAM+x6CW4dAPJrE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? 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: 2 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
Jose@192.168.56.13's password:

Number of key(s) added: 2

Now try logging into the machine, with:   "ssh 'Jose@192.168.56.13'"
and check to make sure that only the key(s) you wanted were added.
```

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

```
[Jose@localhost ~]$ cat ~/.ssh/authorized_keys
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAICQrGQq4PnYEo8TflcsdTW50MPhgQRMdh5V6kyU7FpQ4 josemari@workstation
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQCf3OveHEJ18NFRjRE86bcHjugMmDRIOpfR6YhDdbuEWXljxSgpgAfmAzZr9cIFeuDveWv4lzSBUpEdJwwJ
WrY+BSym4EKzeF7fH+7kilxbYP9yF8D6tbfndARGY/VjMbUmaz6J3xKQAg7vtcX3yRYzOj6vO3mftIEcdwgzGNm01PX9P0q8zliY95YqYJogZEvdxv3aePXu
FteGPLzgfVG19gD9x7uIQU+9/A8RRWt1S0pLvqqvCkc/C6V2ILjIxhiFXn4DrFlg16OnK/Z2iVDP9q1wQCXmAqS1ERVsHghn3zAT9xFlvWl6by4CVYGyHgT6
T4c6QwUdhB5scNMFteDNhA0j35kR3mQnWlitBs+XWKCJr8kfqGtMqsKUHhf2LhMFo+ELxv9+75u/B0HiOxQMCqKOeTVrWyKsi5FVYxYja+cQRomrmfACgDrK
Kji3eODUeJWdVxFSIVLHbwMBNahVHhc3f3f7oIPzaze0Z+KfwWrp51ZnEXqnef+ai9Do9tcAvm22NOIXLDRkAbJLtQ== josemari@workstation
```

Task 4: Verify ssh remote connection

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

```
josemari@workstation:~$ ssh Jose@192.168.56.13
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Fri Sep 27 20:49:45 2024 from 192.168.56.11
[Jose@localhost ~]$
```

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?

When choosing between Debian and Red Hat, consider factors like stability, package management, and community support. Debian is great for personal use and development due to its stability, while Red Hat offers professional support and is ideal for industrial needs.

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

The main differences are in package management and use cases. Debian uses APT with .deb packages, making it popular for development and personal projects. In contrast, Red Hat uses YUM/DNF with .rpm packages, focusing more on industrial environments and providing strong support for businesses.