

Name: Magboo, Matt Clemence C.	Date Performed: 08/28/2025
Course/Section: CPE212-CPE31S2	Date Submitted: 08/28/2025
Instructor: Engr. Robin Valenzuela	Semester and SY: 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.

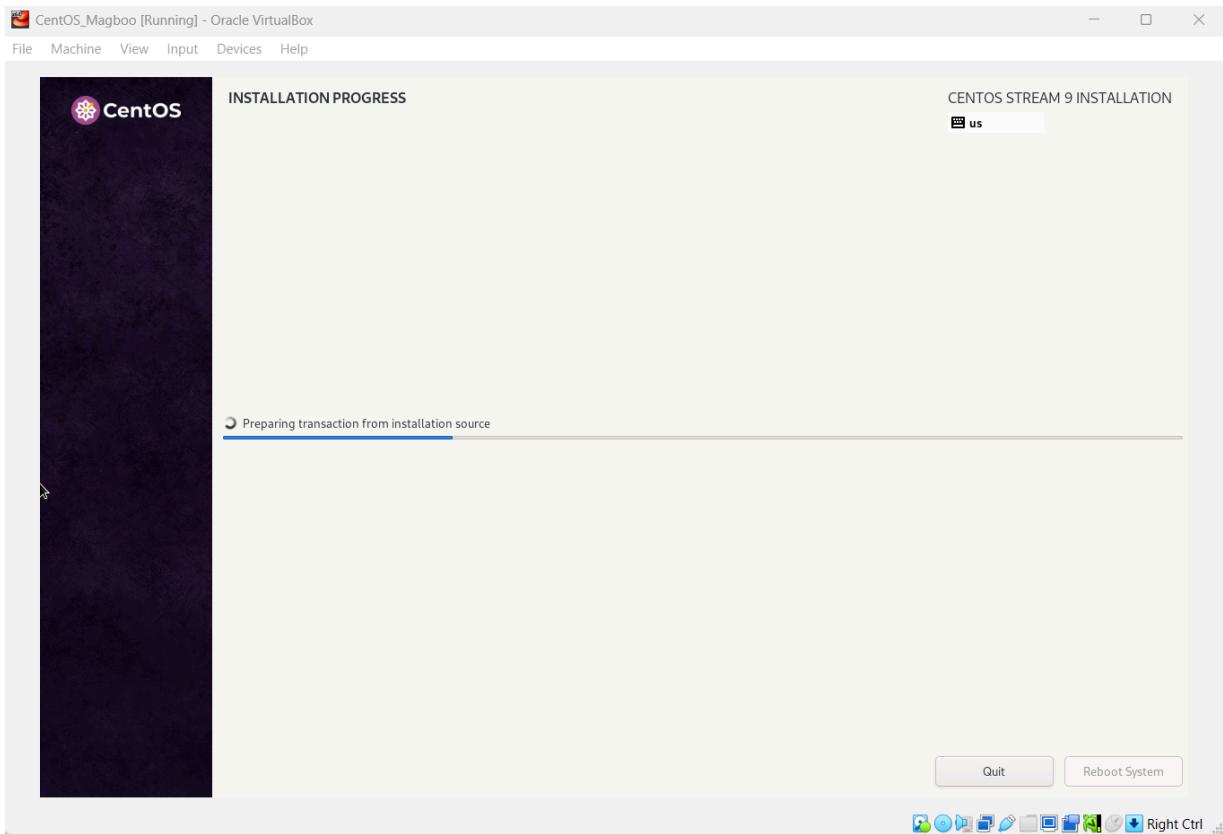
CentOS uses the RPM package format and YUM/DNF as the package manager.

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`

```
[magboo@vbox ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: ena>
  Active: active (running) since Thu 2025-08-28 13:37:33 PST; 1min 43s ago
    Docs: man:sshd(8)
          man:sshd_config(5)
  Main PID: 943 (sshd)
     Tasks: 1 (limit: 22969)
   Memory: 3.5M (peak: 3.8M)
      CPU: 16ms
     CGroup: /system.slice/sshd.service
             └─943 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Aug 28 13:37:33 localhost.localdomain systemd[1]: Starting OpenSSH server daemo>
Aug 28 13:37:33 localhost.localdomain sshd[943]: Server listening on 0.0.0.0 po>
Aug 28 13:37:33 localhost.localdomain sshd[943]: Server listening on :: port 22.
Aug 28 13:37:33 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
lines 1-16/16 (END)
```

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

```
$ systemctl start sshd
$ systemctl enable sshd
```

```
[magboo@vbox ~]$ systemctl start sshd
[magboo@vbox ~]$ systemctl enable sshd
```

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

```
$ systemctl status ssh
```

```
[magboo@vbox ~]$ systemctl enable sshd
[magboo@vbox ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: ena>
  Active: active (running) since Thu 2025-08-28 13:37:33 PST; 9min ago
    Docs: man:sshd(8)
          man:sshd_config(5)
  Main PID: 943 (sshd)
     Tasks: 1 (limit: 22969)
    Memory: 3.5M (peak: 3.8M)
       CPU: 16ms
      CGroup: /system.slice/sshd.service
              └─943 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
Aug 28 13:37:33 localhost.localdomain systemd[1]: Starting OpenSSH server daemo>
Aug 28 13:37:33 localhost.localdomain sshd[943]: Server listening on 0.0.0.0 po>
Aug 28 13:37:33 localhost.localdomain sshd[943]: Server listening on :: port 22.
Aug 28 13:37:33 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
```

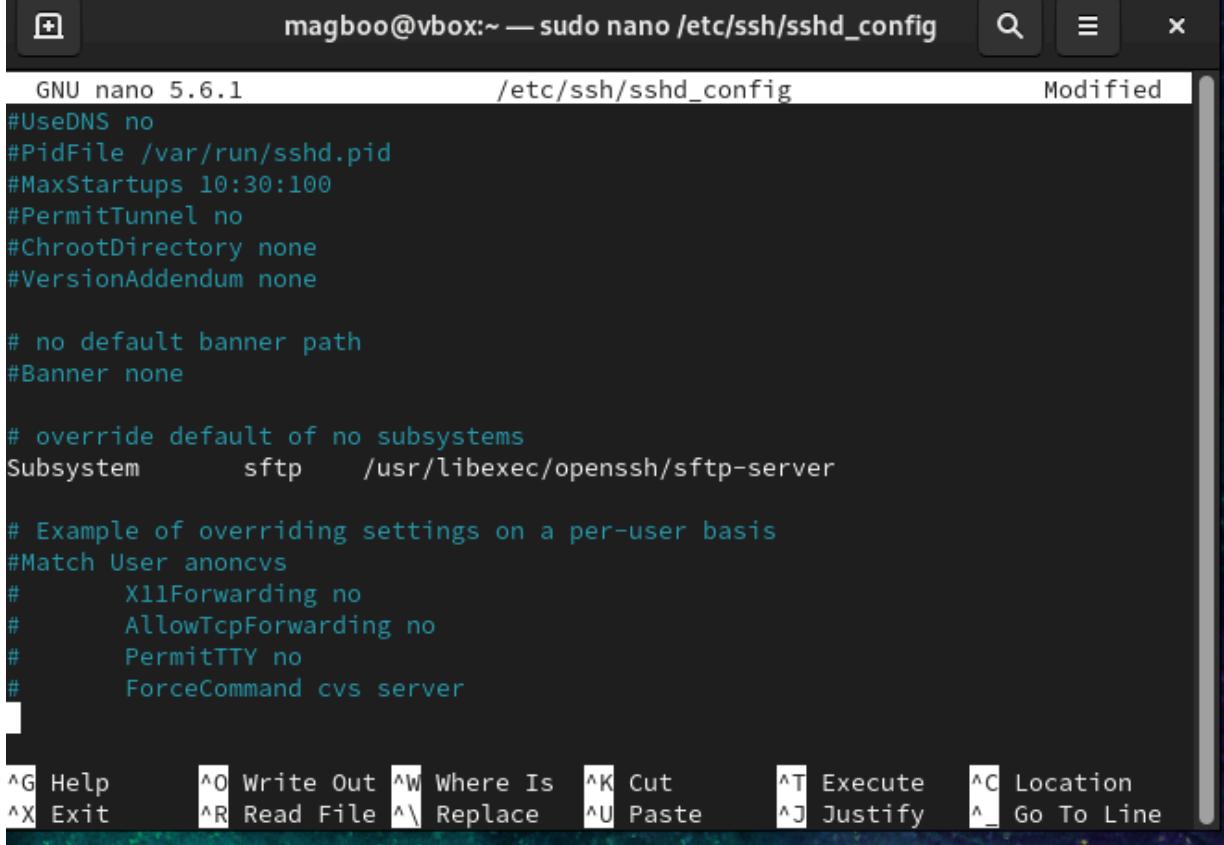
4. Open the SSH port 22 to allow incoming traffic:

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

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

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          Modified
#UseDNS no
#PidFile /var/run/sshd.pid
#MaxStartups 10:30:100
#PermitTunnel no
#ChrootDirectory none
#VersionAddendum none

# no default banner path
#Banner none

# override default of no subsystems
Subsystem      sftp      /usr/libexec/openssh/sftp-server

# Example of overriding settings on a per-user basis
#Match User anoncvs
#    X11Forwarding no
#    AllowTcpForwarding no
#    PermitTTY no
#    ForceCommand cvs server
```

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.

```
magboo@vbox:~
```

```
: qlen 1000
  link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
  inet 127.0.0.1/8 scope host lo
    valid_lft forever preferred_lft forever
  inet6 ::1/128 scope host
    valid_lft forever preferred_lft forever
: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP
  up default qlen 1000
  link/ether 08:00:27:f2:2f:8c brd ff:ff:ff:ff:ff:ff
  inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
    valid_lft 85894sec preferred_lft 85894sec
  inet6 fd17:625c:f037:2:a00:27ff:fef2:2f8c/64 scope global dynamic noprefixroute
    valid_lft 86345sec preferred_lft 14345sec
  inet6 fe80::a00:27ff:fe:f2:2f8c/64 scope link noprefixroute
    valid_lft forever preferred_lft forever
: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP
  up default qlen 1000
  link/ether 08:00:27:a1:20:8a brd ff:ff:ff:ff:ff:ff
  inet 192.168.56.105/24 brd 192.168.56.255 scope global noprefixroute enp0s8
    valid_lft forever preferred_lft forever
  inet6 fe80::6474:b3d3:b60d:bdf8/64 scope link noprefixroute
    valid_lft forever preferred_lft forever
magboo@vbox ~]$
```

```
Magboo@LocalMachine:~$ ssh-copy-id magboo@192.168.56.105
/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 prompt
ed now it is to install the new keys
magboo@192.168.56.105's password:

Number of key(s) added: 1

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

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

```
[magboo@vbox ~]$ whoami  
magboo  
[magboo@vbox ~]$ cd ~/.ssh  
[magboo@vbox .ssh]$ ls -l  
total 4  
-rw----- 1 magboo magboo 745 Aug 28 14:16 authorized_keys  
[magboo@vbox .ssh]$ cat authorized_keys  
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQAcqVBRy5gRDITtW2Du5Vjr8//eiPjbefUUAYtHlqtDP  
212Nsqqj8A7odgCItAow28Lo06jD9S01Hny2EQVsW8JF8U68ZYji8EwWHPKyaqxFTei9feK6bUyeojPag  
tdZKhYlb7/GCJNzuDHBj9r8RpNr+1/nSvJwFabUhciMFz52UpDDintgUCwx9+fGLeUZxc34Fk7WuszSt  
JRjNqpueuFvVB3k3UgJFtLmWNI1wpVqq0zxJ2ENAmPbz3i5Xf0smml9RfsqJvWHS9T23QeaIVgJ66Pwa  
xvJrBMa/zH5a15pugfbazQZOI/FhRbsJGlil4oy+yEDAShRdMgyuT3s1rlZHA3Bh29ZJrrHNp5lRdr3A  
B697gPDpBN05X1gRLKQIxKGBVHTC1i9SekykPYlgXH/bkeLKsFWj5vfs7KhANWhXkYl1ATK8NPuudU82  
sMNy6RF1KR2jj9Y2P5w3NS+P6zwFPmnkrkEKgVPsNbMtaszPhHjFJ2DapZtVbwYaqGmqr0u+V9Wyeeis  
9bJWcDK41hiKWaCET3xfhNNV2kcj3+yFVl+pxfkn3ucyQcecZgNfVBTCohkbnxEuD+vBqFcQbf0cVy/V  
yfURyDskaEP5BK0jdbMolmEt8IcK2cED+nILAWLAfjbJy7Vxvcplh+cgq7z9UxBWUNRuNbZHPgjVUU71  
/Q== Magboo@LocalMachine  
[magboo@vbox .ssh]$
```

Task 4: Verify ssh remote connection

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

```
Magboo@LocalMachine:~$ ssh magboo@192.168.56.105
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Thu Aug 28 14:01:54 2025
[magboo@vbox ~]$ hostname
vbox
[magboo@vbox ~]$ whoami
magboo
[magboo@vbox ~]$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host
                valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
    qlen 1000
        link/ether 08:00:27:f2:2f:8c brd ff:ff:ff:ff:ff:ff
        inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
            valid_lft 85242sec preferred_lft 85242sec
            inet6 fd17:625c:f037:2:a00:27ff:fef2:2f8c/64 scope global dynamic noprefixroute
                valid_lft 86146sec preferred_lft 14146sec
            inet6 fe80::a00:27ff:fef2:2f8c/64 scope link noprefixroute
                valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
    qlen 1000
        link/ether 08:00:27:a1:20:8a brd ff:ff:ff:ff:ff:ff
        inet 192.168.56.105/24 brd 192.168.56.255 scope global noprefixroute enp0s8
            valid_lft forever preferred_lft forever
            inet6 fe80::6474:b3d3:b60d:bdf8/64 scope link noprefixroute
```

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?

the security features with both have access to ssh it is safe to say that ubuntu and centOS have safety features of ssh.

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

the command apt for debian and yum for red hat.

