Name:	**Date Performed:**
Course/Section:	**Date Submitted:**
Instructor:	**Semester and SY:**

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.

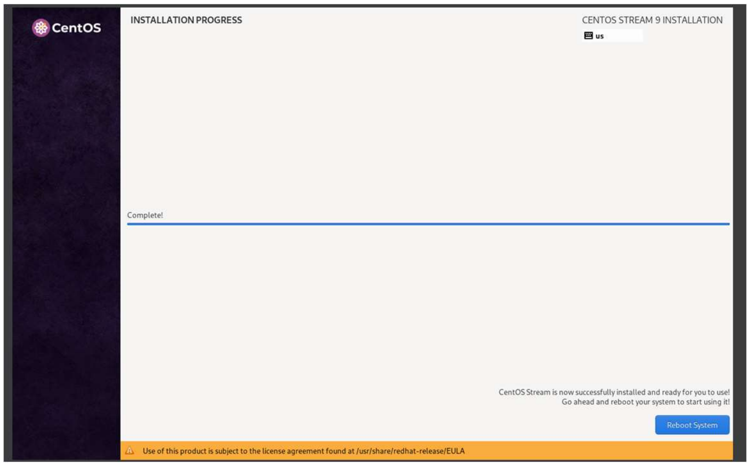
CentOS uses the RPM package format and YUM/DNF as the package manager.

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)

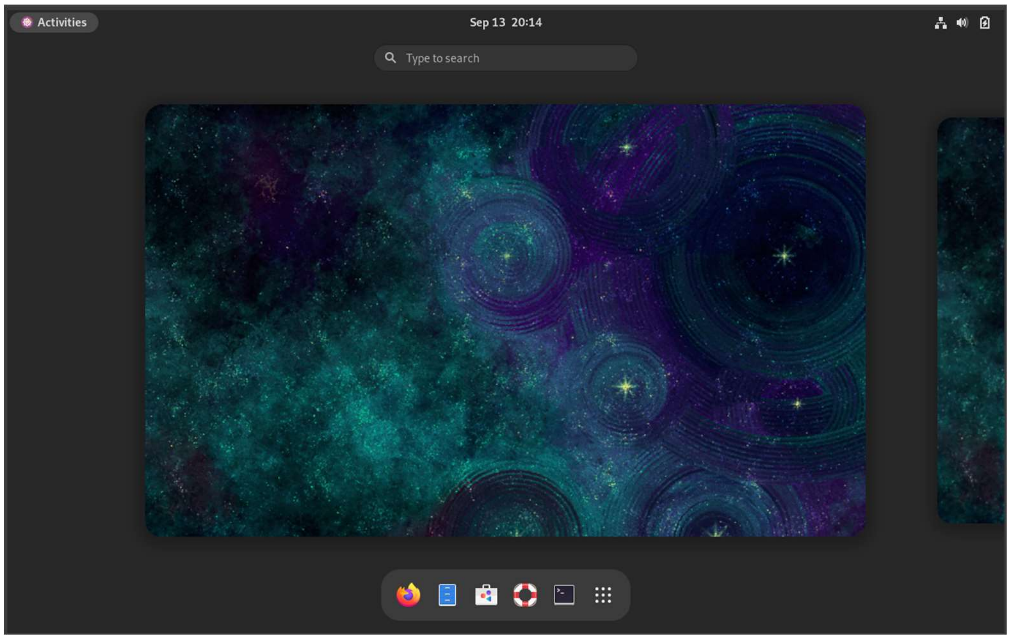
1. Download the image of the CentOS here: http://mirror.rise.ph/centos/7.9.2009/isos/x86_64/
2. Create a VM machine with 2 Gb RAM and 20 Gb HD.
3. Install the downloaded image.
4. Show evidence that the OS was installed already.



Complete!

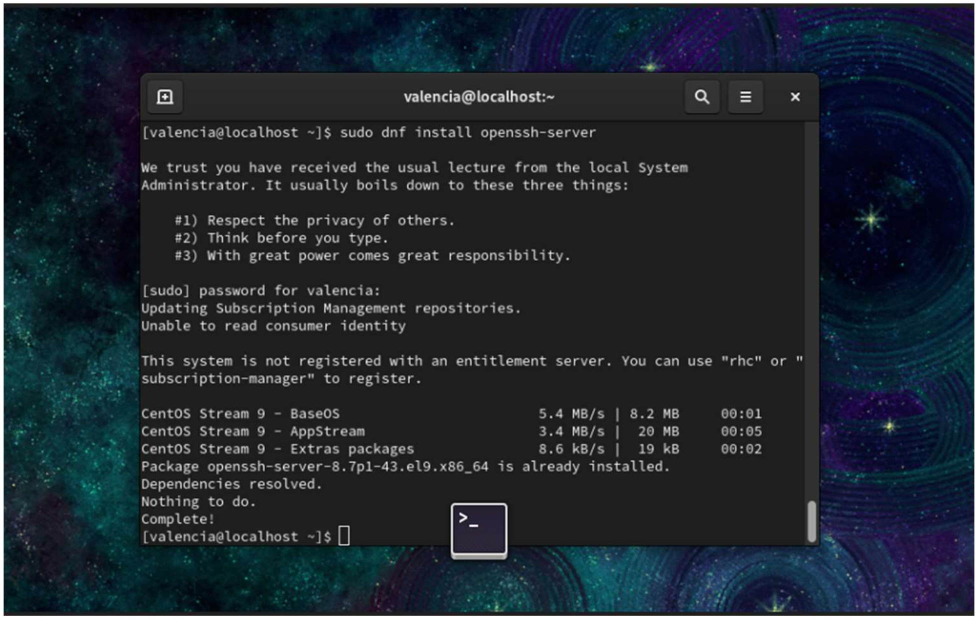
CentOS Stream is now successfully installed and ready for you to use!
Go ahead and reboot your system to start using it!

Use of this product is subject to the license agreement found at /usr/share/redhat-release/EULA



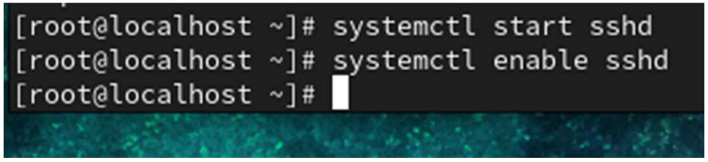
Task 2: Install the SSH server package *openssh*

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



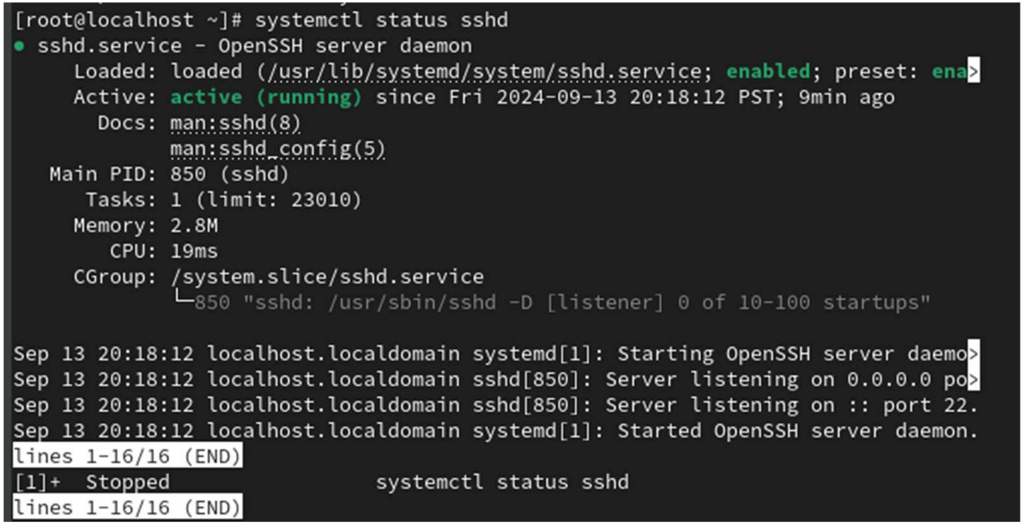
```
[valencia@localhost ~]$ sudo dnf install openssh-server

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 valencia:
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use "rhc" or "
subscription-manager" to register.

CentOS Stream 9 - BaseOS                        5.4 MB/s | 8.2 MB     00:01
CentOS Stream 9 - AppStream                     3.4 MB/s |  20 MB     00:05
CentOS Stream 9 - Extras packages               8.6 kB/s |  19 kB     00:02
Package openssh-server-8.7p1-43.el9.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[valencia@localhost ~]$
```

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



```
[root@localhost ~]# systemctl start sshd
[root@localhost ~]# systemctl enable sshd
[root@localhost ~]#
```

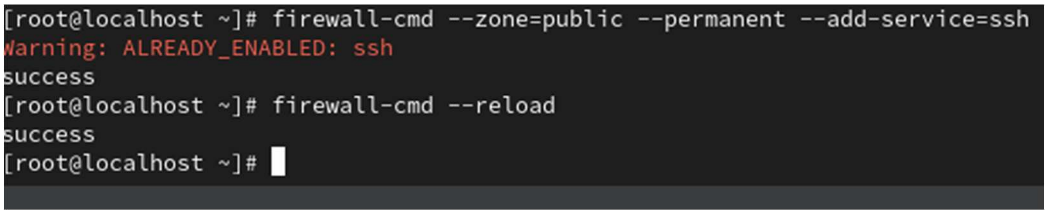
3. Confirm that the sshd daemon is up and running:
   $ systemctl status sshd



```
[root@localhost ~]# systemctl status sshd
● sshd.service - OpenSSH server daemon
     Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: ena>
     Active: active (running) since Fri 2024-09-13 20:18:12 PST; 9min ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 850 (sshd)
      Tasks: 1 (limit: 23010)
     Memory: 2.8M
        CPU: 19ms
     CGroup: /system.slice/sshd.service
             └─850 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Sep 13 20:18:12 localhost.localdomain systemd[1]: Starting OpenSSH server daemo>
Sep 13 20:18:12 localhost.localdomain sshd[850]: Server listening on 0.0.0.0 po>
Sep 13 20:18:12 localhost.localdomain sshd[850]: Server listening on :: port 22.
Sep 13 20:18:12 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
lines 1-16/16 (END)
[1]+  Stopped                 systemctl status sshd
lines 1-16/16 (END)
```

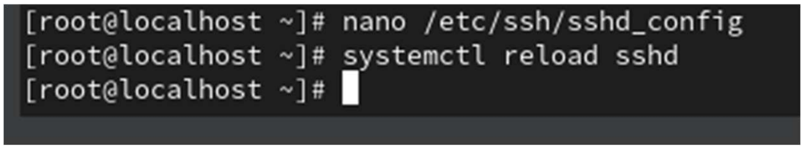
4. Open the SSH port 22 to allow incoming traffic:

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



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

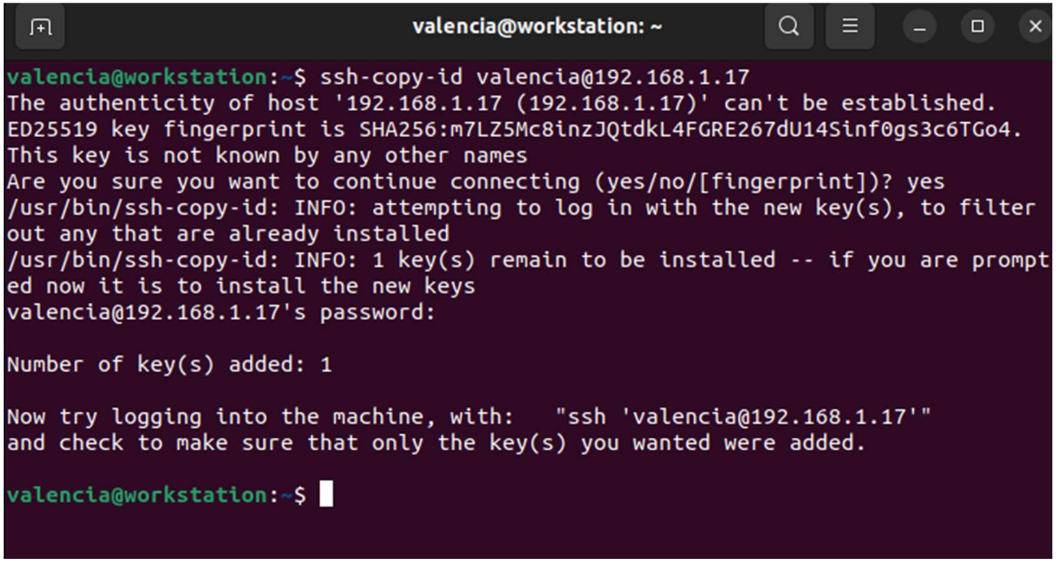
5. Locate the ssh server man config file /*etc/ssh/sshd_config* and perform custom configuration. Every time you make any change to the /*etc/ssh/sshd-config* configuration file reload the *sshd* service to apply changes:
   $ systemctl reload sshd



```
[root@localhost ~]# nano /etc/ssh/sshd_config
[root@localhost ~]# systemctl reload sshd
[root@localhost ~]# 
```

Task 3: Copy the Public Key to CentOS

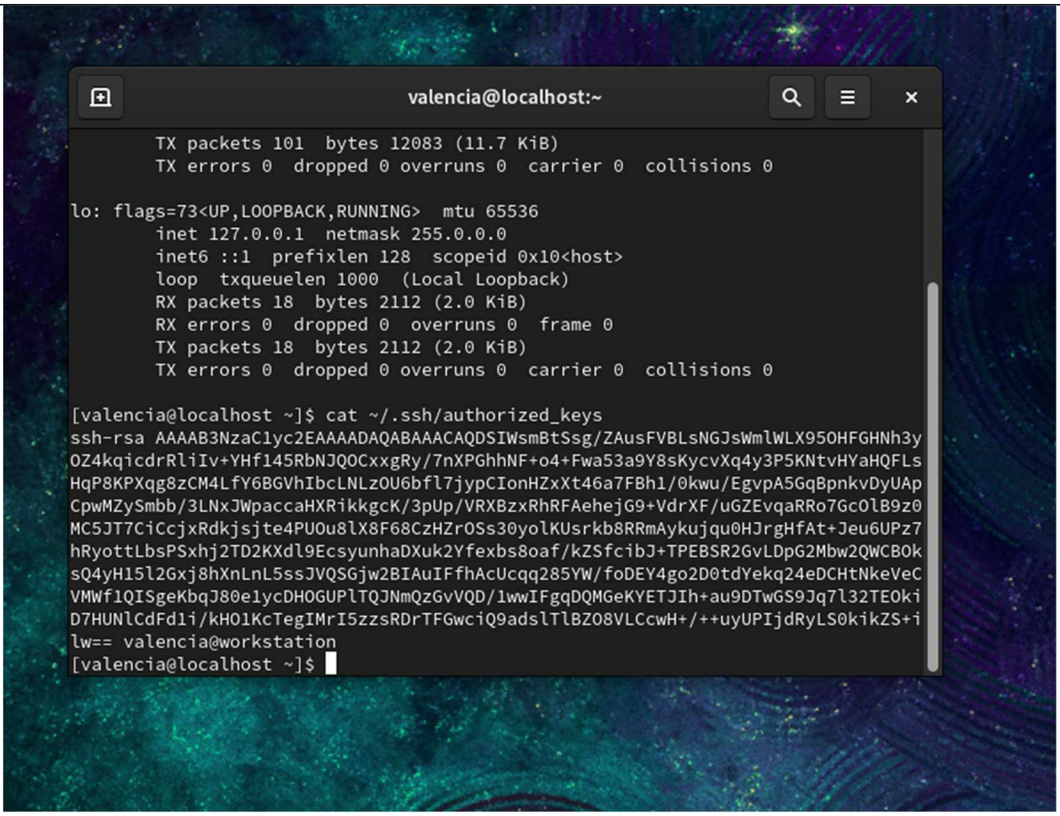
1. Make sure that *ssh* is installed on the local machine.
2. Using the command *ssh-copy-id*, connect your local machine to CentOS.



valencia@workstation: ~

```
valencia@workstation:~$ ssh-copy-id valencia@192.168.1.17
The authenticity of host '192.168.1.17 (192.168.1.17)' can't be established.
ED25519 key fingerprint is SHA256:m7LZ5Mc8inzJQtdkL4FGRE267dU14Sinf0gs3c6TGo4.
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: 1 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
valencia@192.168.1.17's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'valencia@192.168.1.17'"
and check to make sure that only the key(s) you wanted were added.

valencia@workstation:~$ 
```

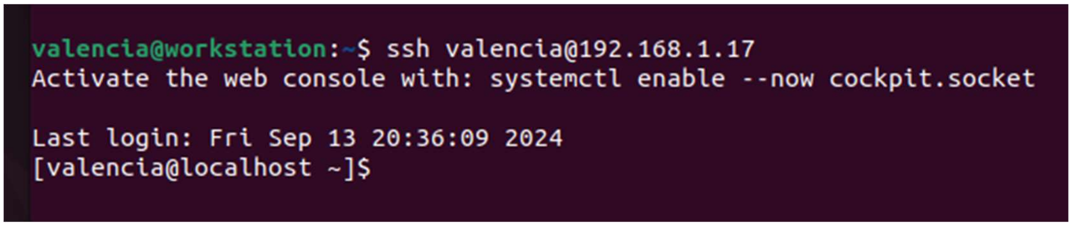
3. On CentOS, verify that you have the *authorized_keys*.



```
        TX packets 101  bytes 12083 (11.7 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 18  bytes 2112 (2.0 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 18  bytes 2112 (2.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[valencia@localhost ~]$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQDSIWsmBtSsg/ZAusFVBLsNGJsWmlWLX95OHFGHNh3y
OZ4kqicdrRliIv+YHf145RbNJQOCxxgRy/7nXPGhhNF+o4+Fwa53a9Y8sKycvXq4y3P5KNtvHYaHQFLs
HqP8KPXqg8zCM4LfY6BGVhIbcLNLzOU6bfl7jypCIonHZxXt46a7FBh1/0kwu/EgvpA5GqBpnkvDyUAp
CpwMZySmbb/3LNxJWpaccaHXRikkgcK/3pUp/VRXBzxRhRFAehejG9+VdrXF/uGZEvqaRRo7GcOlB9z0
MC5JT7CiCcjxRdkjsjte4PUOu8lX8F68CzHZrOSs30yolKUsrkb8RRmAykujqu0HJrgHfAt+Jeu6UPz7
hRyottLbsPSxhj2TD2KXdl9EcsyunhaDXuk2Yfexbs8oaf/kZSfcibJ+TPEBSR2GvLDpG2Mbw2QWCBOk
sQ4yH15l2Gxj8hXnLnL5ssJVQSGjw2BIAuIFfhAcUcqq285YW/foDEY4go2D0tdYekq24eDCHtNkeVeC
VMWf1QISgeKbqJ80e1ycDHOGUPlTQJNmQzGvVQD/1wwIFgqDQMGeKYETJIh+au9DTwGS9Jq7l32TEOki
D7HUNlCdFd1i/kHO1KcTegIMrI5zzsRDrTFGwciQ9adslTlBZO8VLCcwH+/++uyUPIjdRyLS0kikZS+i
lw== valencia@workstation
[valencia@localhost ~]$ 
```

Task 4: Verify ssh remote connection

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



```
valencia@workstation:~$ ssh valencia@192.168.1.17
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Fri Sep 13 20:36:09 2024
[valencia@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?
   - It depends on how and where you would use the distributions since Debian is more leaning for small projects while Red Hat Linux is used more on enterprise level projects.
2. What are the main difference between Debian and Red Hat Linux distributions?
   - Debian is more on smaller form servers while Red Hat Linux is more on bigger and enterprise level servers.