Name: Ejercito, Marlon Jason A.	Date Performed:29/08/2023
Course/Section: CPE232 / CPE31S4	Date Submitted:03/09//2023
Instructor: Dr. Jonathan Taylar	Semester and SY: 1st Semester and 2023 - 2024

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.

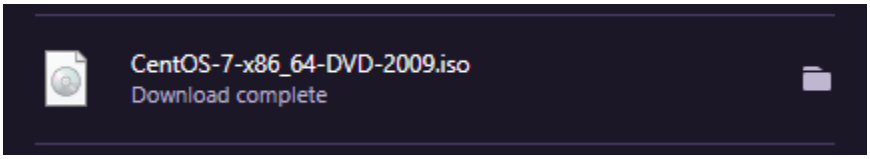
CentOS uses the RPM package format and YUM/DNF as the package manager.

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

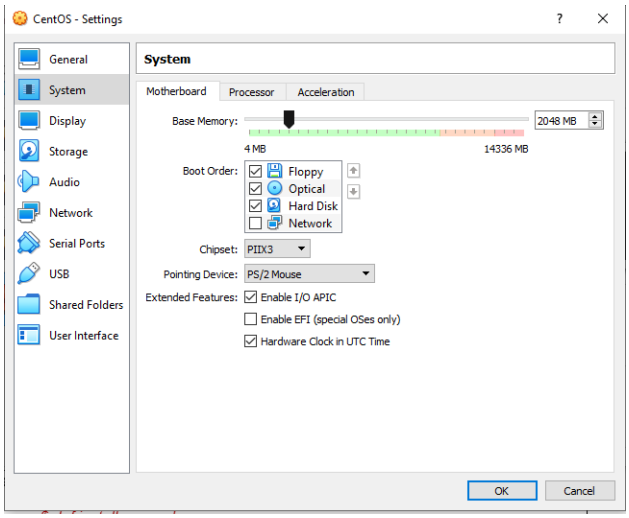
Both offer full-feature package management with network-based repository support, dependency checking and resolution, etc.. If you're familiar with one but not the other, you may have a little trouble switching over, but they're not overwhelmingly different. They both have similar features, just available through a different interface.

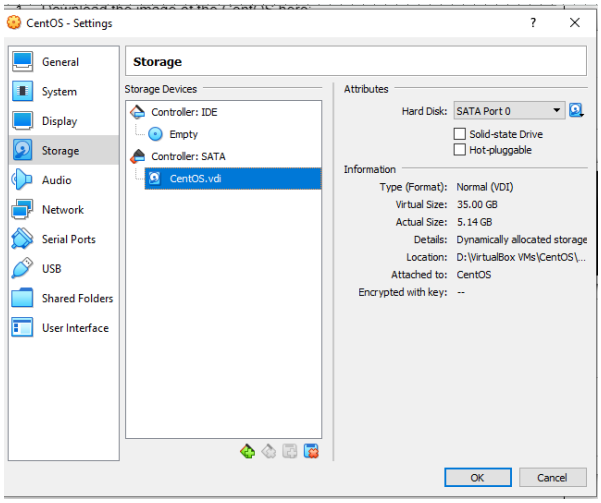
Task 1: Download the CentOS or RHEL-8 image (Create screenshots of the following)

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

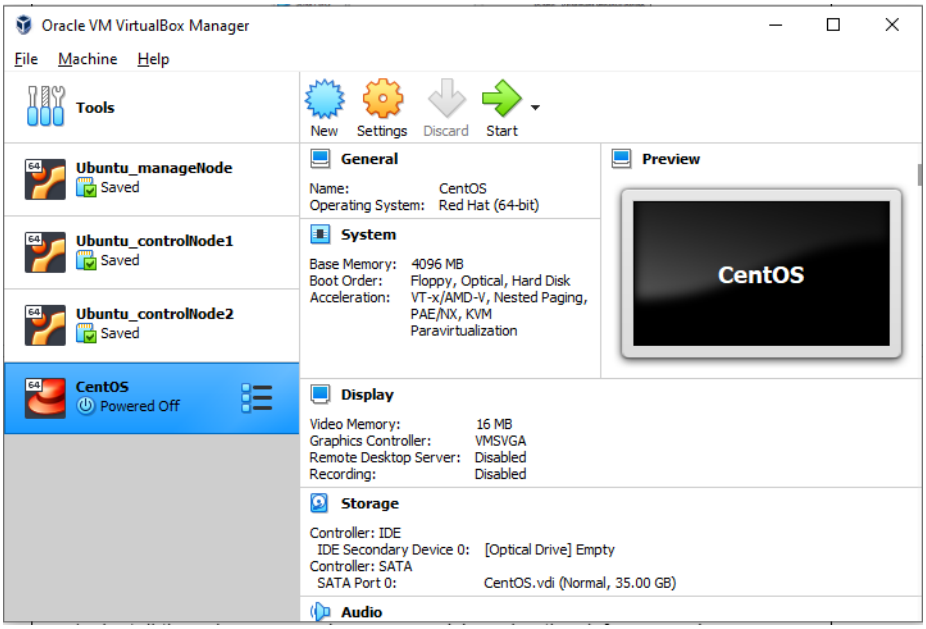


2. Create a VM machine with 2 Gb RAM and 20 Gb HD.

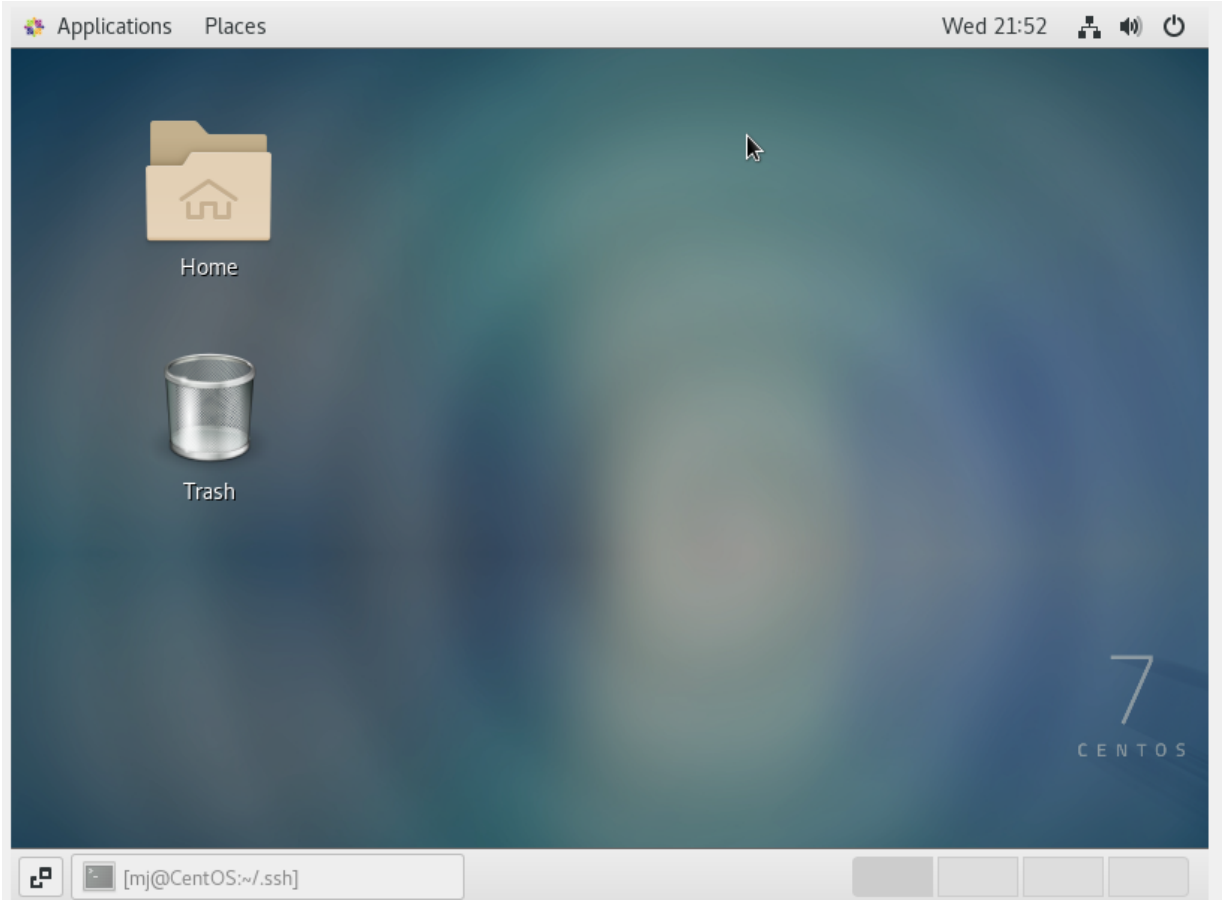




3. Install the downloaded image.



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

```
[mj@localhost ~]$ dnf install openssh-server
Error: This command has to be run under the root user.
[mj@localhost ~]$ sudo dnf install openssh-server
Extra Packages for Enterprise Linux 7 - x86_64         1.2 MB/s |  17 MB
CentOS-7 - Base                                        3.2 MB/s |  10 MB
CentOS-7 - Updates                                     3.4 MB/s |  28 MB
CentOS-7 - Extras                                      786 kB/s | 360 kB
Package openssh-server-7.4p1-21.el7.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
```

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

   $ systemctl start sshd

```
[mj@localhost ~]$ sudo systemctl start sshd
```

   $ systemctl enable sshd

```
[mj@localhost ~]$ sudo systemctl enable sshd
```

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

   $ systemctl status sshd

```
[mj@localhost ~]$ sudo systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor pr
d)
   Active: active (running) since Wed 2023-08-30 20:53:28 EDT; 11min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
 Main PID: 1175 (sshd)
   CGroup: /system.slice/sshd.service
           └─1175 /usr/sbin/sshd -D

Aug 30 20:53:28 localhost.localdomain systemd[1]: Starting OpenSSH server d
Aug 30 20:53:28 localhost.localdomain sshd[1175]: Server listening on 0.0.0
Aug 30 20:53:28 localhost.localdomain sshd[1175]: Server listening on :: po
Aug 30 20:53:28 localhost.localdomain systemd[1]: Started OpenSSH server da
Hint: Some lines were ellipsized, use -l to show in full.
```

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

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

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

   $ firewall-cmd --reload

```
[mj@localhost ~]$ 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:

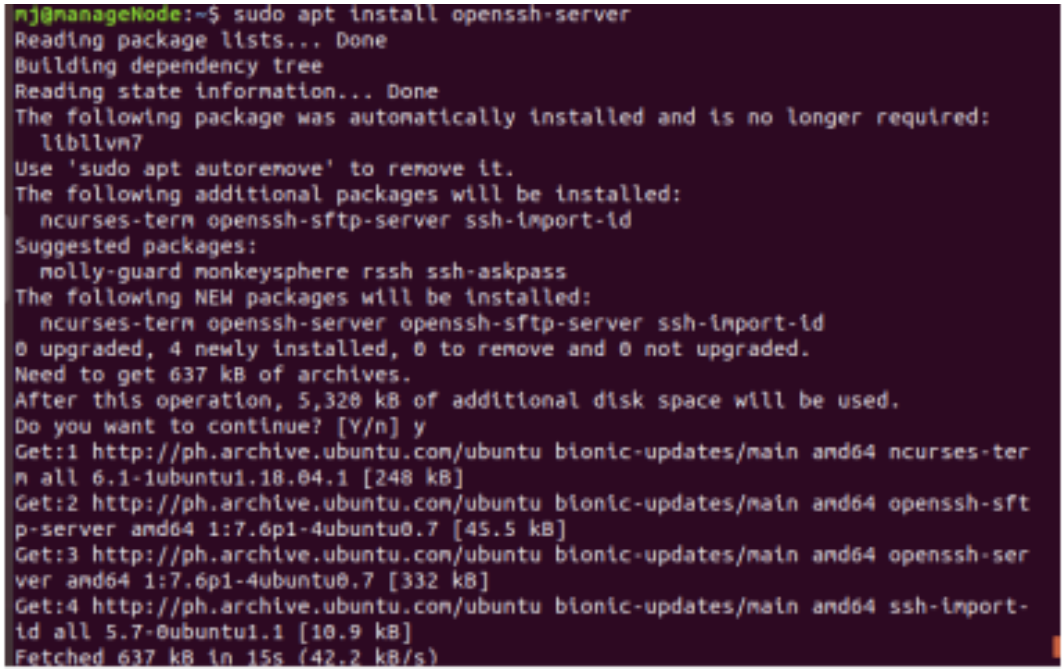
```
[mj@localhost ~]$ ls /etc/ssh
moduli       ssh_host_ecdsa_key      ssh_host_ed25519_key.pub
ssh_config   ssh_host_ecdsa_key.pub  ssh_host_rsa_key
sshd_config  ssh_host_ed25519_key    ssh_host_rsa_key.pub
```

$ systemctl reload sshd

```
[mj@localhost ~]$ sudo systemctl reload sshd
```

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.

```
mj@manageNode:~$ ssh-copy-id -i  ~/.ssh/id_rsa mj@192.168.56.104
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/mj/.ssh/i
d_rsa.pub"
The authenticity of host '192.168.56.104 (192.168.56.104)' can't be establishe
d.
ED25519 key fingerprint is SHA256:vNb856iT7n+MGxvgH7Zgu6szhwjXAZ057UAeLQ97+7E.
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 filte
r out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prom
pted now it is to install the new keys
mj@192.168.56.104's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'mj@192.168.56.104'"
and check to make sure that only the key(s) you wanted were added.
```

```
mj@manageNode:~$ ssh mj@CentOS
The authenticity of host 'centos (192.168.56.104)' can't be established.
ED25519 key fingerprint is SHA256:vNb856iT7n+MGxvgH7Zgu6szhwjXAZ057UAeLQ97+7E.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:10: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'centos' (ED25519) to the list of known hosts.
Last login: Wed Aug 30 21:39:23 2023
[mj@CentOS ~]$ ssh mj@192.168.56.104
The authenticity of host '192.168.56.104 (192.168.56.104)' can't be establishe
d.
ECDSA key fingerprint is SHA256:uctbEIs/wtz9iMgRBg1h6WiXsvH1qMBEoBx0p3Unso8.
ECDSA key fingerprint is MD5:31:aa:00:50:f9:58:9f:3b:38:ab:35:92:1f:91:7e:a5.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': ye
Please type 'yes' or 'no': yes
Warning: Permanently added '192.168.56.104' (ECDSA) to the list of known hosts
.
mj@192.168.56.104's password:
Last login: Wed Aug 30 21:41:12 2023 from 192.168.56.101
[mj@CentOS ~]$ ssh mj@CentOS
```

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

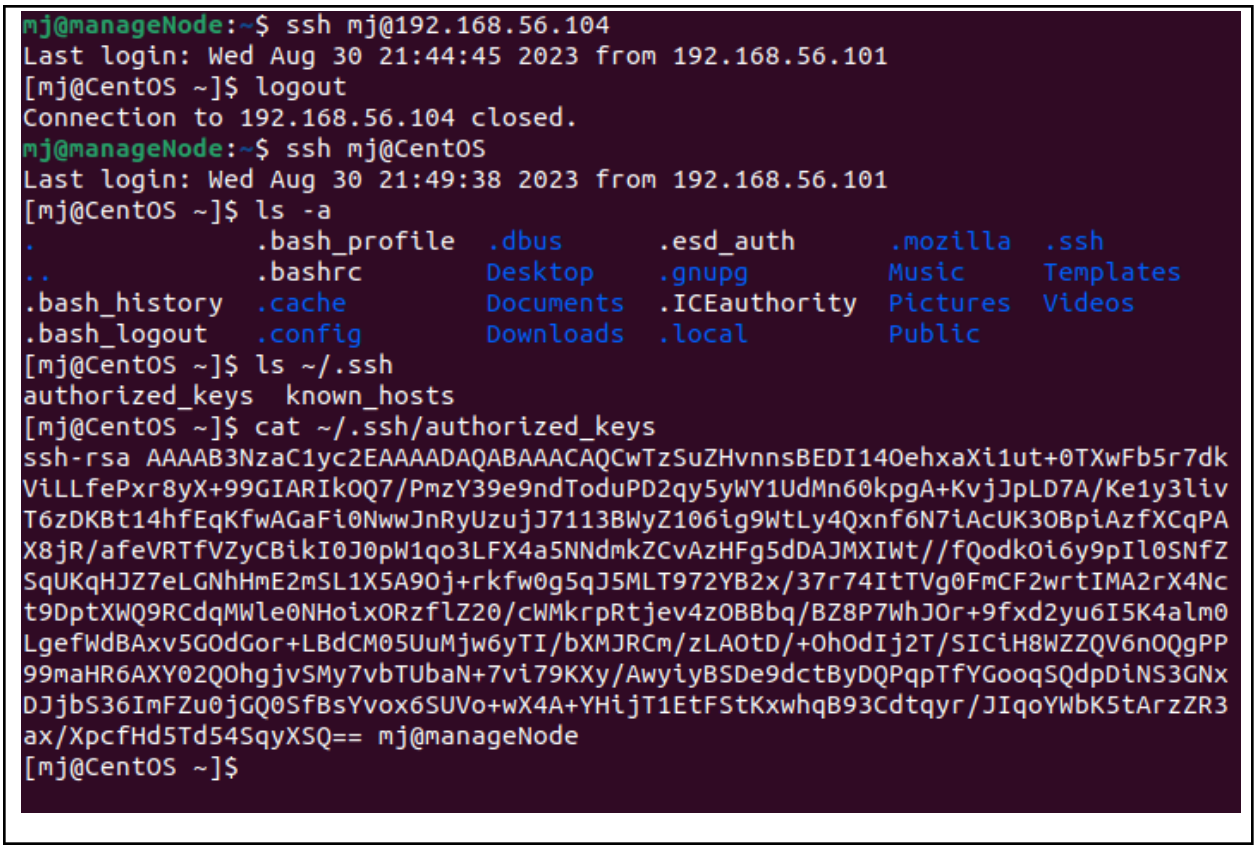
```
[mj@CentOS ~]$ cd ~/.ssh
[mj@CentOS .ssh]$ ls
authorized_keys  known_hosts
[mj@CentOS .ssh]$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQCwTzSuZHvnnsBEDI14OehxaXi1ut+0TXwFb5r7dkViLLfePxr
8yX+99GIARIkOQ7/PmzY39e9ndToduPD2qy5yWY1UdMn60kpgA+KvjJpLD7A/Ke1y3livT6zDKBt14hfEqKfwAG
aFi0NwwJnRyUzujJ7113BWyZ106ig9WtLy4Qxnf6N7iAcUK3OBpiAzfXCqPAX8jR/afeVRTfVZyCBikI0J0pW1q
o3LFX4a5NNdmkZCvAzHFg5dDAJMXIWt//fQodkOi6y9pIl0SNfZSqUKqHJZ7eLGNhHmE2mSL1X5A9Oj+rkfw0g5
qJ5MLT972YB2x/37r74ItTVg0FmCF2wrtIMA2rX4Nct9DptXWQ9RCdqMWle0NHoixORzflZ20/cWMkrpRtjev4z
OBBbq/BZ8P7WhJOr+9fxd2yu6I5K4alm0LgefWdBAxv5GOdGor+LBdCM05UuMjw6yTI/bXMJRCm/zLAOtD/+OhO
dIj2T/SICiH8WZZQV6nOQgPP99maHR6AXY02QOhgjvSMy7vbTUbaN+7vi79KXy/AwyiyBSDe9dctByDQPqpTfYG
ooqSQdpDiNS3GNxDJjbS36ImFZu0jGQ0SfBsYvox6SUVo+wX4A+YHijT1EtFStKxwhqB93Cdtqyr/JIqoYWbK5t
ArzZR3ax/XpcfHd5Td54SqyXSQ== mj@manageNode
[mj@CentOS .ssh]$
```

Task 4: Verify ssh remote connection

1. Using your local machine, connect to CentOS using ssh.

```
mj@manageNode:~$ ssh mj@192.168.56.104
Last login: Wed Aug 30 21:44:45 2023 from 192.168.56.101
[mj@CentOS ~]$ logout
Connection to 192.168.56.104 closed.
mj@manageNode:~$ ssh mj@CentOS
Last login: Wed Aug 30 21:49:38 2023 from 192.168.56.101
[mj@CentOS ~]$
```

2. Show evidence that you are connected.


mj@manageNode:~$ ssh mj@192.168.56.104
Last login: Wed Aug 30 21:44:45 2023 from 192.168.56.101
[mj@CentOS ~]$ logout
Connection to 192.168.56.104 closed.
mj@manageNode:~$ ssh mj@CentOS
Last login: Wed Aug 30 21:49:38 2023 from 192.168.56.101
[mj@CentOS ~]$ ls -a
. .bash_profile .dbus .esd_auth .mozilla .ssh
.. .bashrc Desktop .gnupg Music Templates
.bash_history .cache Documents .ICEauthority Pictures Videos
.bash_logout .config Downloads .local Public
[mj@CentOS ~]$ ls ~/.ssh
authorized_keys known_hosts
[mj@CentOS ~]$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQCwTzSuZHvnnsBEDI14OehxaXi1ut+0TXwFb5r7dk
ViLLfePxr8yX+99GIARIkOQ7/PmzY39e9ndToduPD2qy5yWY1UdMn60kpgA+KvjJpLD7A/Ke1y3liv
T6zDKBt14hfEqKfwAGaFi0NwwJnRyUzujJ7113BWyZ106ig9WtLy4Qxnf6N7iAcUK3OBpiAzfXCqPA
X8jR/afeVRTfVZyCBikI0J0pW1qo3LFX4a5NNdmkZCvAzHFg5dDAJMXIWt//fQodkOi6y9pIl0SNfZ
SqUKqHJZ7eLGNhHmE2mSL1X5A9Oj+rkfw0g5qJ5MLT972YB2x/37r74ItTVg0FmCF2wrtIMA2rX4Nc
t9DptXWQ9RCdqMWle0NHoixORzflZ20/cWMkrpRtjev4zOBBbq/BZ8P7WhJOr+9fxd2yu6I5K4alm0
LgefWdBAxv5GOdGor+LBdCM05UuMjw6yTI/bXMJRCm/zLAOtD/+OhOdIj2T/SICiH8WZZQV6nOQgPP
99maHR6AXY02QOhgjvSMy7vbTUbaN+7vi79KXy/AwyiyBSDe9dctByDQPqpTfYGooqSQdpDiNS3GNx
DJjbS36ImFZu0jGQ0SfBsYvox6SUVo+wX4A+YHijT1EtFStKxwhqB93Cdtqyr/JIqoYWbK5tArzZR3
ax/XpcfHd5Td54SqyXSQ== mj@manageNode
[mj@CentOS ~]$

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?

In my opinion, both distributions are the best in their field. Debian is commonly used in community, as it gives the users freedom, stability and versatility - general purpose computing. Red Hat is commonly used in commercial or enterprises, as it gives the users long term support,enterprise-grade features, and paid support services. It is commonly used in enterprise environments where stability and vendor support are essential considerations. Overall, we should look for the purpose, cost, and the future use of the distribution.

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

The main difference between Debian and Red Hat is that the packages, release cycles, and support models are different.

For Debian, the package management that they use is the APT or the Advanced Package Tool, along with DEB packages - it simplifies software installation, updates and removal. While Red Hat uses RPM or the Red Hat Package Manager - as it is used for software distro, and the ‘yum’ (Yellowdog Updater, Modified) or ‘dnf’ (Dandified YUM) are used for package management tasks.

In their release cycles, Debian has multiple release branches, which includes “stable”, “testing”, and “unstable”. Debian’s release cycle is known for its long release cycles, which can accumulate for several years. While RHEL (Red Hat Enterprise Linux) has a predictable and long-term support (LTS) release cycle, which spans to ten years or more.

The support model of Debian offers community support through forums, and community contributions while Red Hat offers commercial support through subscriptions - paying for technical support, or access to certified software and security updates.

Conclusion:

In this activity, we were able to apply what we've learned in Activity 1 and 2, which is configuring, and installing remote SSH connection to a remote server/computer which is the CentOS. In addition, we were able to differentiate the two distributions: CentOS and Debian. The two distributions have different uses for different purposes - if we want to use a distribution for general purpose, Debian should be our choice. On the other hand, if we want it for the commercial or enterprise industry, we should use Red Hat.

In conclusion, we were able to attain the objectives of this activity, which is to install community enterprise OS or Red Hat Linux OS which is CentOS 7, and configure remote SSH connection from remote computer to CentOS/RHEL-8, in this case, we were able to make a remote connection from a Ubuntu distribution, to CentOS distribution, with the Local Machine in the Ubuntu Server.