Name: Ashley L. Mallari	Date Performed: 09/07/2023
Course/Section: CPE232 - CPE31S6	Date Submitted: 09/07/2023
Instructor: Dr. Jonathan V. Taylae	Semester and SY: 1st Sem / 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.

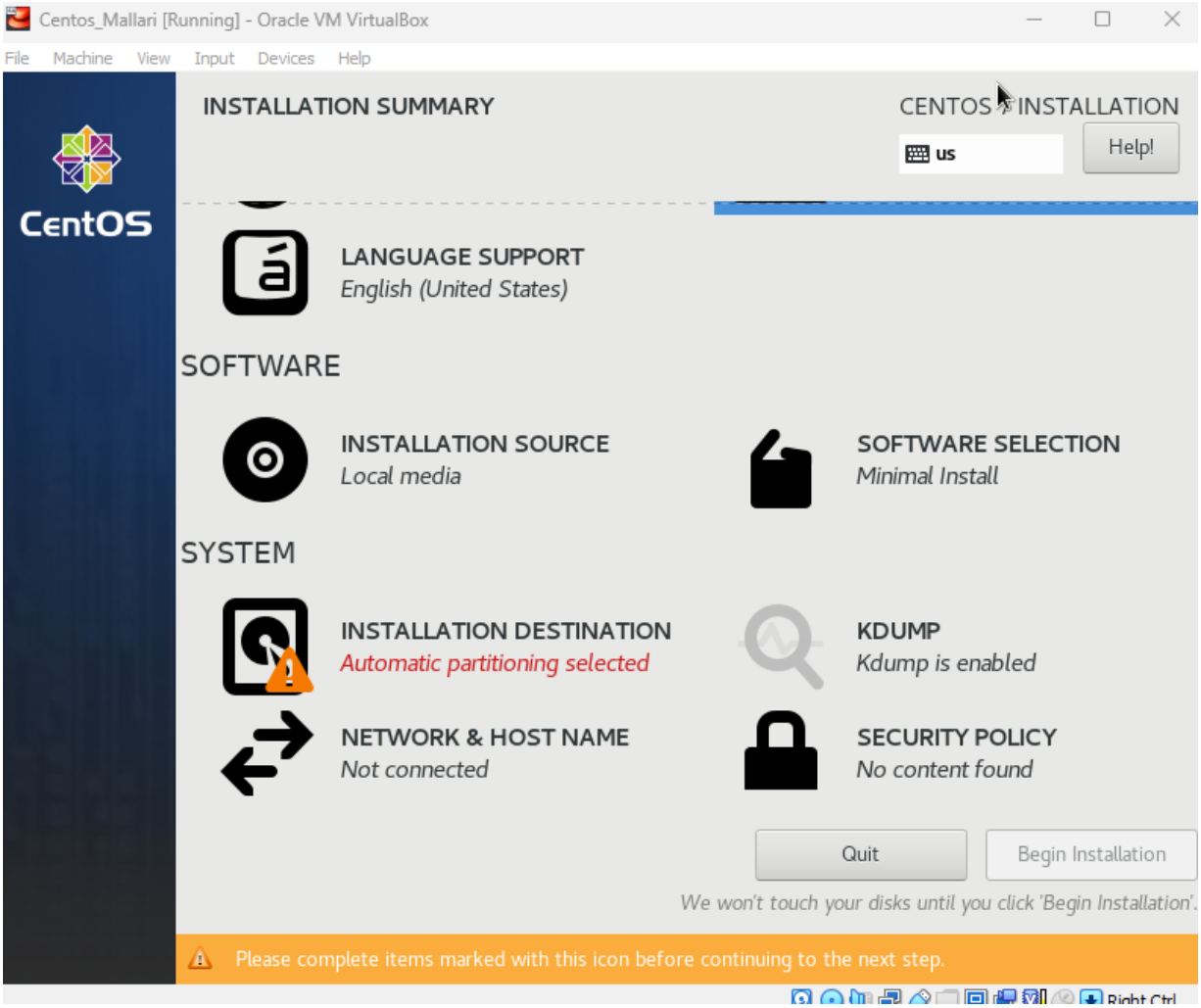
CentOS uses the RPM package format and YUM/DNF as the package manager.

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

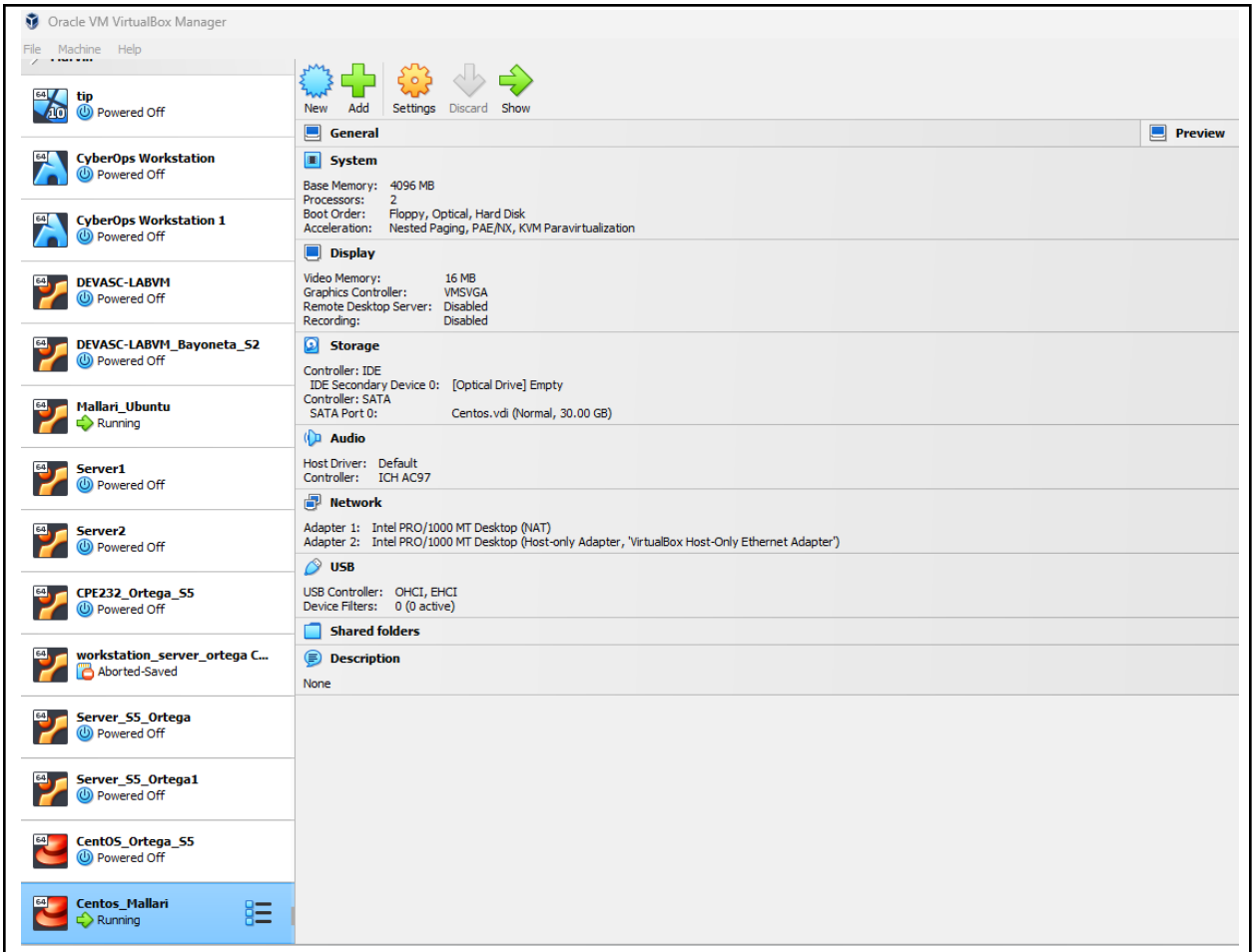
Both offer full-feature package management with network-based repository support, dependency checking and resolution, etc.. If you're familiar with one but not the other, you may have a little trouble switching over, but they're not overwhelmingly different. They both have similar features, just available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

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

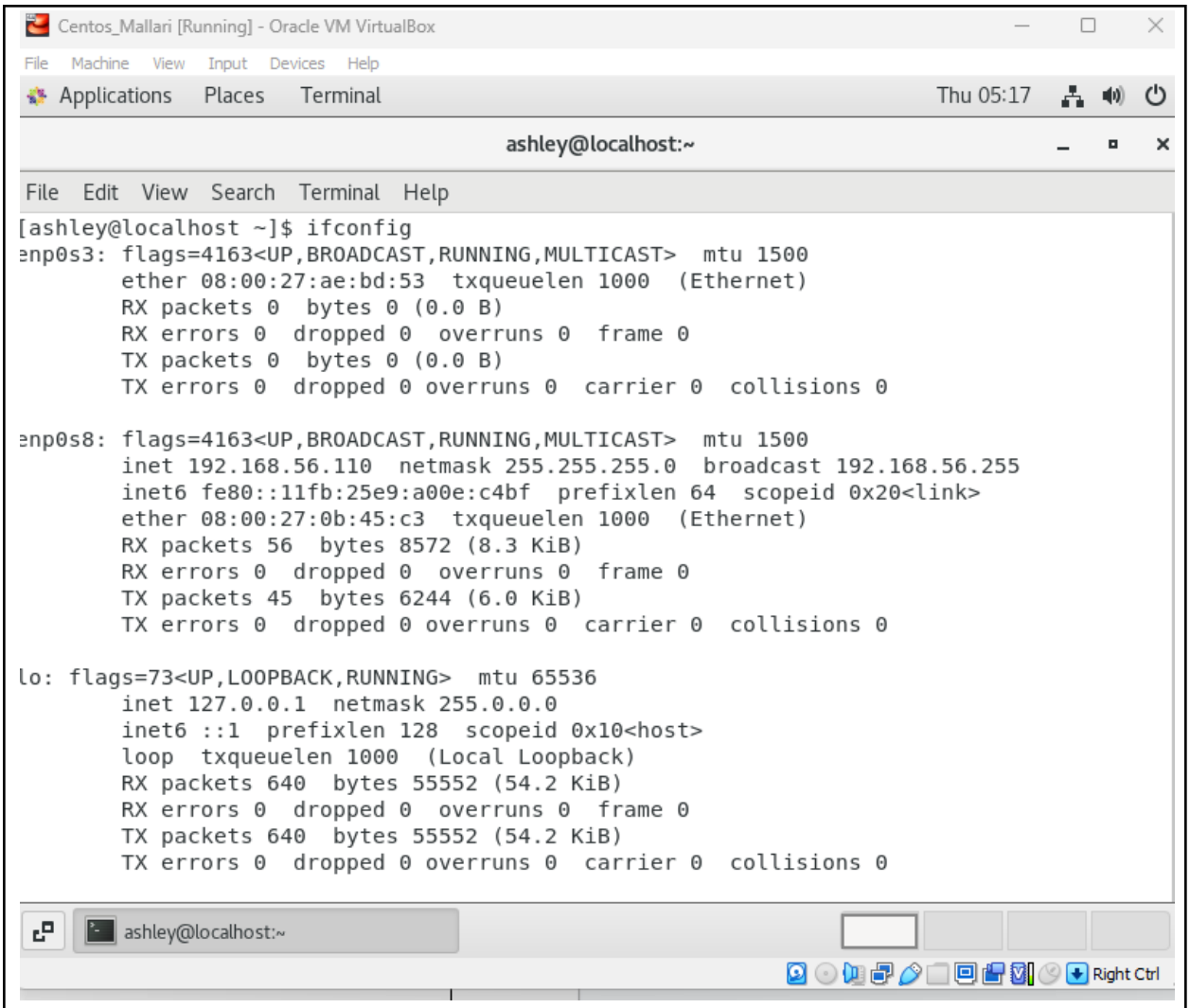
1. Download the image of the CentOS here: http://mirror.rise.ph/centos/7.9.2009/isos/x86_64/



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

```
[ashley@localhost ~]$ sudo yum 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 ashley:
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirror-hk.koddos.net
 * extras: mirror-hk.koddos.net
 * updates: mirror-hk.koddos.net
Resolving Dependencies
--> Running transaction check
---> Package openssh-server.x86_64 0:7.4p1-21.el7 will be updated
---> Package openssh-server.x86_64 0:7.4p1-23.el7_9 will be an update
--> Processing Dependency: openssh = 7.4p1-23.el7_9 for package: openssh-server-7.4p1-2
3.el7_9.x86_64
--> Running transaction check
---> Package openssh.x86_64 0:7.4p1-21.el7 will be updated
--> Processing Dependency: openssh = 7.4p1-21.el7 for package: openssh-clients-7.4p1-21
.el7.x86_64
---> Package openssh.x86_64 0:7.4p1-23.el7_9 will be an update
```

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

```
[ashley@localhost ~]$ systemctl start sshd
[ashley@localhost ~]$ systemctl enable sshd
```

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

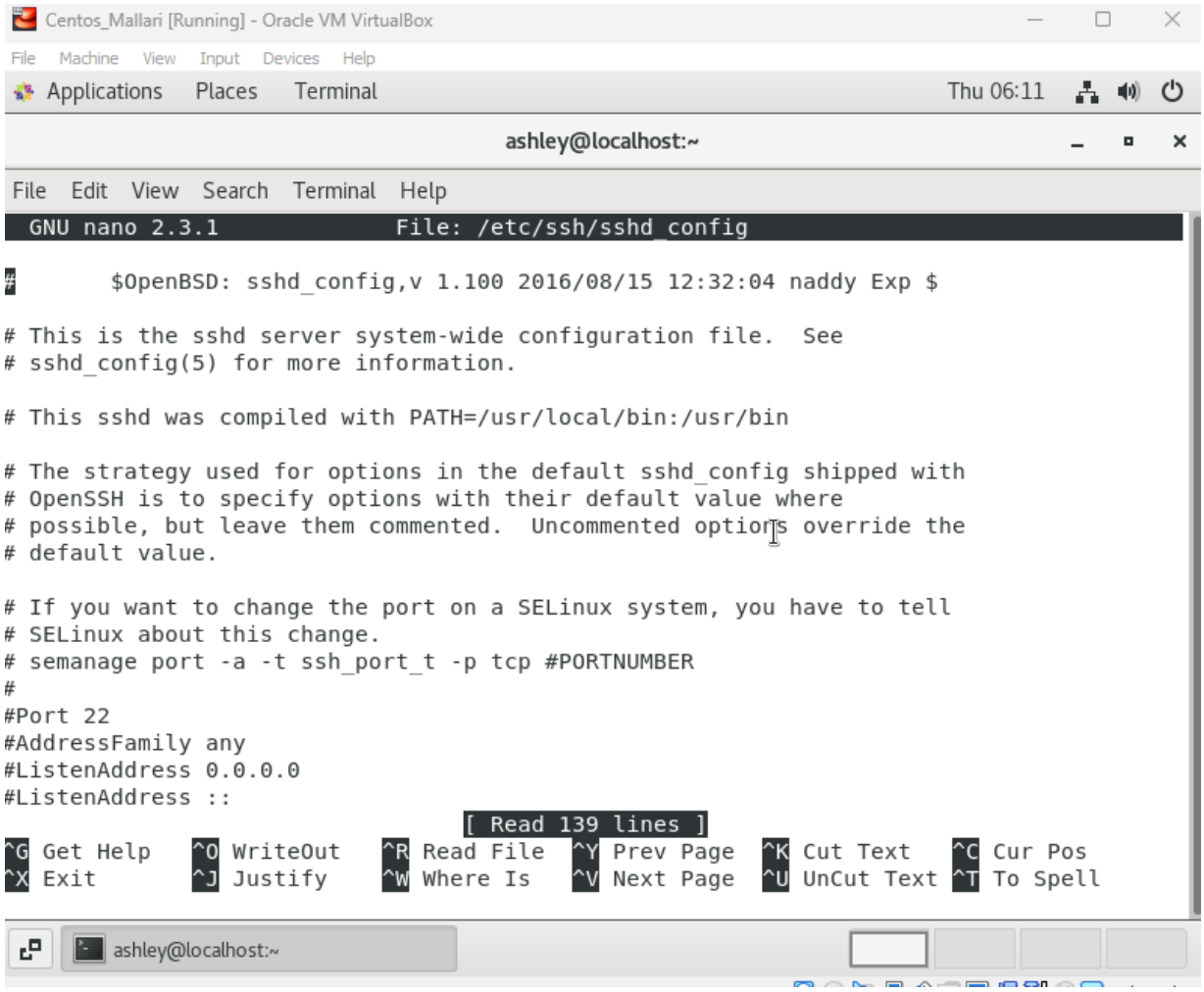
```
[ashley@localhost ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enable
d)
   Active: active (running) since Thu 2023-09-07 05:43:14 EDT; 1min 27s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
 Main PID: 10317 (sshd)
   CGroup: /system.slice/sshd.service
           └─10317 /usr/sbin/sshd -D

Sep 07 05:43:14 localhost.localdomain systemd[1]: Starting OpenSSH server daemon...
Sep 07 05:43:14 localhost.localdomain sshd[10317]: Server listening on 0.0.0.0 port 22.
Sep 07 05:43:14 localhost.localdomain sshd[10317]: Server listening on :: port 22.
Sep 07 05:43:14 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
Hint: Some lines were ellipsized, use -l to show in full.
[ashley@localhost ~]$
```

4. Open the SSH port 22 to allow incoming traffic:
   $ firewall-cmd --zone=public --permanent --add-service=ssh
   $ firewall-cmd --reload

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




```
[ashley@localhost ~]$ systemctl reload sshd
[ashley@localhost ~]$
```

Task 3: Copy the Public Key to CentOS

1. Make sure that *ssh* is installed on the local machine.

```
ashleymallari@workstation:~$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ashleymallari/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ashleymallari/.ssh/id_rsa.
Your public key has been saved in /home/ashleymallari/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:3TfdBhxAjz1Zr0d6tm6/nIsZjcJ6Zy4YRLisjCgKaEI ashleymallari@workstation
The key's randomart image is:
+---[RSA 4096]----+
|        .  .o.. .|
|       . .   = +.|
|      . o   . B o|
| E     o o .   B.|
|o . o . S . . = B|
|=o . o   ..  .o*.|
|*         oo o o |
|.        ..o.o*..|
|         .. =+.*=|
+----[SHA256]-----+
ashleymallari@workstation:~$
```

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

```
ashleymallari@workstation:~$ ssh-copy-id -i ~/.ssh/id_rsa ashley@192.168.56.112
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ashleymall
ari/.ssh/id_rsa.pub"
The authenticity of host '192.168.56.112 (192.168.56.112)' can't be established
.
ECDSA key fingerprint is SHA256:JQL/z46cqOH3TKIVx+lNMewbN1O+BiHJkY0px66we24.
Are you sure you want to continue connecting (yes/no)? 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 promp
ted now it is to install the new keys
ashley@192.168.56.112's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'ashley@192.168.56.112'"
and check to make sure that only the key(s) you wanted were added.

ashleymallari@workstation:~$
```

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

```
[ashley@localhost ~]$ cd ~/.ssh
[ashley@localhost .ssh]$ ls
authorized_keys
[ashley@localhost .ssh]$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQC1ntZZFid9fi7ClIQC7EyBOWF3RDAncXoGVmX2wgBqRLkYNCg
kK1Mn0KHdkaN2YoIwD7BROgNtOosDfby4HrEdt7p0e4mO4NBuytAjhJA7iiNGuEV/Tsjv5vmoB/etCDtTM2uWQs
AJiOUFxDMrkXvy8e3sGpUIE/uh+RkhJm2tMbdFDN10quDO3RzRannH0AEFZ7k1WYLokaqFdMNbkWCrU+KXnfMc+
J4NJGfRWLRkkLnB/9uWPUA5jPdyiKYkcigCV+oVvOhE7ImiBNeLmM3f1Z/duUYPYpb6Y6dK6di2P9HyfE9LfICk
qGeLsTAd67HX0Nt7DQKcOMlmuvEV1SlhIc8U4FoeSnGWyOZYWiIWLoYKaePk6nfl/CWECf0Il+IMEB0IjdUo9PB
QsGo7ixWldNhVqi17EqgFtlskuIeKhQmeSp59uNi5IQfN6Fa7+qAKJ+TnER1RgdfN+I955rATtJLUAe/cfDZrMP
dTaCKVXeWEW95xakX+Wkv1fycWaFefmYAd8NfkxZiKN/WjmX+NXeVJqO9uu5qCM9RNakzj2nUg4XBEqjLWK4u+k
cn2nsQwv3JYITaWaDzECPC/r0BB7uFcckCY4ivWRKmW32jsWCzT4tu+tWuLqPO8SGzZ4f7CFQSV1J1unbwpWTUC
c6EULzx2N5irTqllbmEMpLmqFQ== ashleymallari@workstation
[ashley@localhost .ssh]$
```

Task 4: Verify ssh remote connection

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

```
ashleymallari@workstation:~$ ssh ashley@192.168.56.112
Last login: Thu Sep  7 06:08:55 2023
[ashley@localhost ~]$
```

2. Show evidence that you are connected.

```
File  Edit  View  Search  Terminal  Help
ashleymallari@workstation:~$ ping 192.168.56.112
PING 192.168.56.112 (192.168.56.112) 56(84) bytes of data.
64 bytes from 192.168.56.112: icmp_seq=1 ttl=64 time=0.406 ms
64 bytes from 192.168.56.112: icmp_seq=2 ttl=64 time=0.172 ms
64 bytes from 192.168.56.112: icmp_seq=3 ttl=64 time=0.289 ms
64 bytes from 192.168.56.112: icmp_seq=4 ttl=64 time=0.149 ms
64 bytes from 192.168.56.112: icmp_seq=5 ttl=64 time=0.148 ms
^Z
[1]+  Stopped                 ping 192.168.56.112
ashleymallari@workstation:~$
```

```
[ashley@localhost ~]$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
[ashley@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 Linux distributions, consider your specific requirements and priorities. Debian is known for its stability and extensive package repository, making it suitable for servers and desktops alike. Red Hat, on the other hand, offers strong support and certification for

enterprise environments, making it a robust choice for mission-critical systems.

2. What are the main differences between Debian and Red Hat Linux distributions?
   - Red Hat relies on the Red Hat Package Manager (RPM). RPM is a widely used package management system known for its flexibility and compatibility. However, managing RPM-based systems may require a slightly steeper learning curve, especially for those more familiar with Debian's APT.