Name: Zamora, Denzel	Date Performed: 9 / 7 / 23
Course/Section: CPE232 - CPE32S6	Date Submitted: 9 / 7 / 23
Instructor: Dr. Jonathan Taylar	Semester and SY: 1st Sem SY 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

```
[dzamora@localhost ~]$ yum install openssh-server
Loaded plugins: fastestmirror, langpacks
You need to be root to perform this command.
[dzamora@localhost ~]$ su
Password:
[root@localhost dzamora]# yum install openssh-server
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirror.xtom.com.hk
 * extras: mirror.xtom.com.hk
 * updates: mirror.xtom.com.hk
base
extras
updates
(1/4): base/7/x86_64/group_gz
(2/4): extras/7/x86_64/primary_db
(3/4): updates/7/x86_64/primary_db
(4/4): base/7/x86_64/primary_db
Resolving Dependencies
--> Running transaction check
---> Package openssh-server.x86_64 0:7.4p1-21.el7 will
---> Package openssh-server.x86_64 0:7.4p1-23.el7_9 wi
```

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

```
[dzamora@localhost ~]$ systectl start sshd
bash: systectl: command not found...
[dzamora@localhost ~]$ systemctl start sshd
[dzamora@localhost ~]$ systemctl enable sshd
[dzamora@localhost ~]$
```

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

```
[root@localhost dzamora]# 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 05:32:11 EDT; 1min 1s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
 Main PID: 3387 (sshd)
   CGroup: /system.slice/sshd.service
           └─3387 /usr/sbin/sshd -D

Sep 07 05:32:11 localhost.localdomain systemd[1]: Starting OpenSSH server daemon...
Sep 07 05:32:11 localhost.localdomain sshd[3387]: Server listening on 0.0.0.0 port 22.
Sep 07 05:32:11 localhost.localdomain sshd[3387]: Server listening on :: port 22.
Sep 07 05:32:11 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
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
   $ firewall-cmd --reload

```
[dzamora@localhost ~]$ firewall-cmd --zone=public --permanent --add-service=ssh
Warning: ALREADY_ENABLED: ssh
success
[dzamora@localhost ~]$ firewall-cmd --reload
success
[dzamora@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 ~]# ls -la .ssh
total 8
drwx------. 2 root root   38 Sep  7 05:45 .
dr-xr-x---. 6 root root  230 Sep  7 05:45 ..
-rw-------. 1 root root 1675 Sep  7 05:45 id_rsa
-rw-r--r--. 1 root root  408 Sep  7 05:45 id_rsa.pub
[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.

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

```
zamora@workstation:~$ ssh-copy-id -i ~/.ssh/id_rsa dzamora@192.168.56.109
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/zamora/.ss
h/id_rsa.pub"
/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
dzamora@192.168.56.109's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'dzamora@192.168.56.109'"
and check to make sure that only the key(s) you wanted were added.
```

```
[dzamora@localhost ~]$ cd .ssh
[dzamora@localhost .ssh]$
```

Task 4: Verify ssh remote connection

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

```
zamora@workstation:~$ ssh dzamora@192.168.56.109
dzamora@192.168.56.109's password:
Last login: Thu Sep  7 05:29:35 2023
[dzamora@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?
   - Use Case
   - Package Management & Availability
   - Licensing
   - Security

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

Debian uses Advanced Package Tool (APT) and deb packages, while Redhat uses RPM package manager and rpm packages.