

Name: Flores, Marc Oliver U.	Date Performed: 29/08/2023
Course/Section: CPE 232-CPE31S4	Date Submitted: 05/09/2023
Instructor: Dr. Jonathan V. Taylar	Semester and SY: First Semester 2023-2024
Activity 3: Install SSH server on CentOS or RHEL 8	
<p>1. Objectives:</p> <p>1.1 Install Community Enterprise OS or Red Hat Linux OS</p> <p>1.2 Configure remote SSH connection from remote computer to CentOS/RHEL-8</p>	
<p>2. Discussion:</p> <p>CentOS vs. Debian: Overview</p> <p>CentOS and Debian are Linux distributions that spawn from opposite ends of the candle.</p> <p>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.</p> <p>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.</p> <p>CentOS vs. Debian: Architecture</p> <p>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?</p> <p>Both Debian and CentOS support AArch64/ARM64, armhf/armhfp, i386, ppc64el/ppc64le. (Note: armhf/armhfp and i386 are supported in CentOS 7 only.)</p> <p>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.</p> <p>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.</p>	

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.

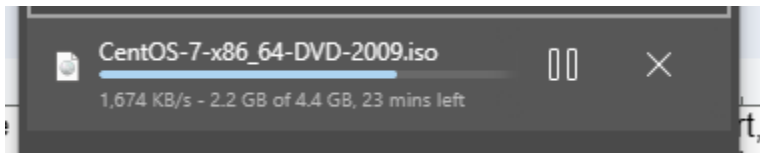
Debian uses the DEB package format and dpkg/APT as the package manager.

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 4 Gb RAM and 35 Gb HD.

Hardware

You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory: 4096 MB

Processors: 2

☐ Enable EFI (special OSes only)

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

☒ Create a Virtual Hard Disk Now

Disk Size: 35.00 GB

☐ Pre-allocate Full Size

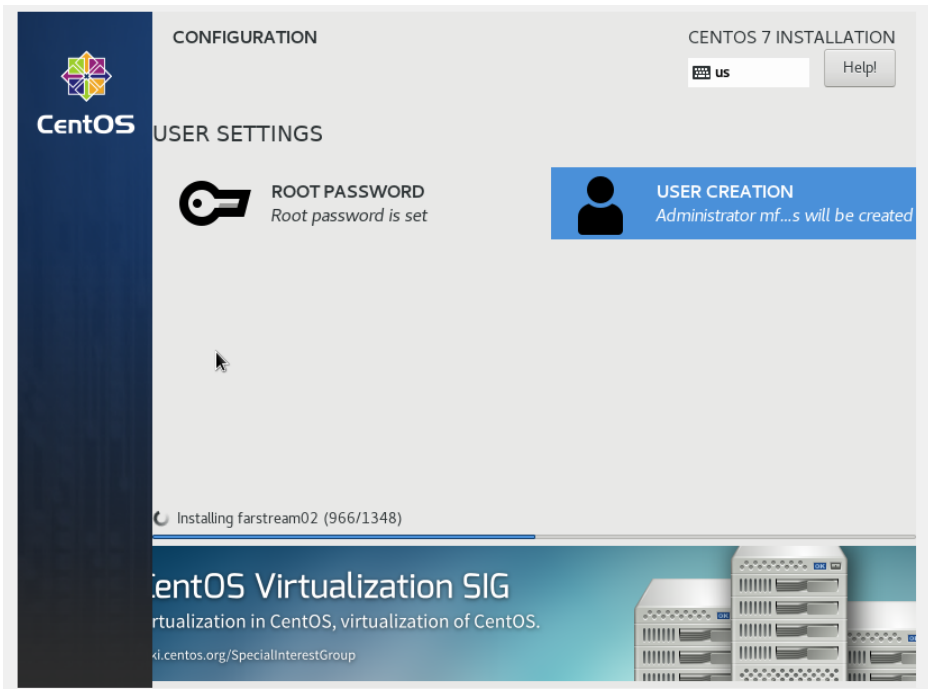
☐ Use an Existing Virtual Hard Disk File

CyberOps Workstation-disk002.vdi (Normal, 1.00 GB)

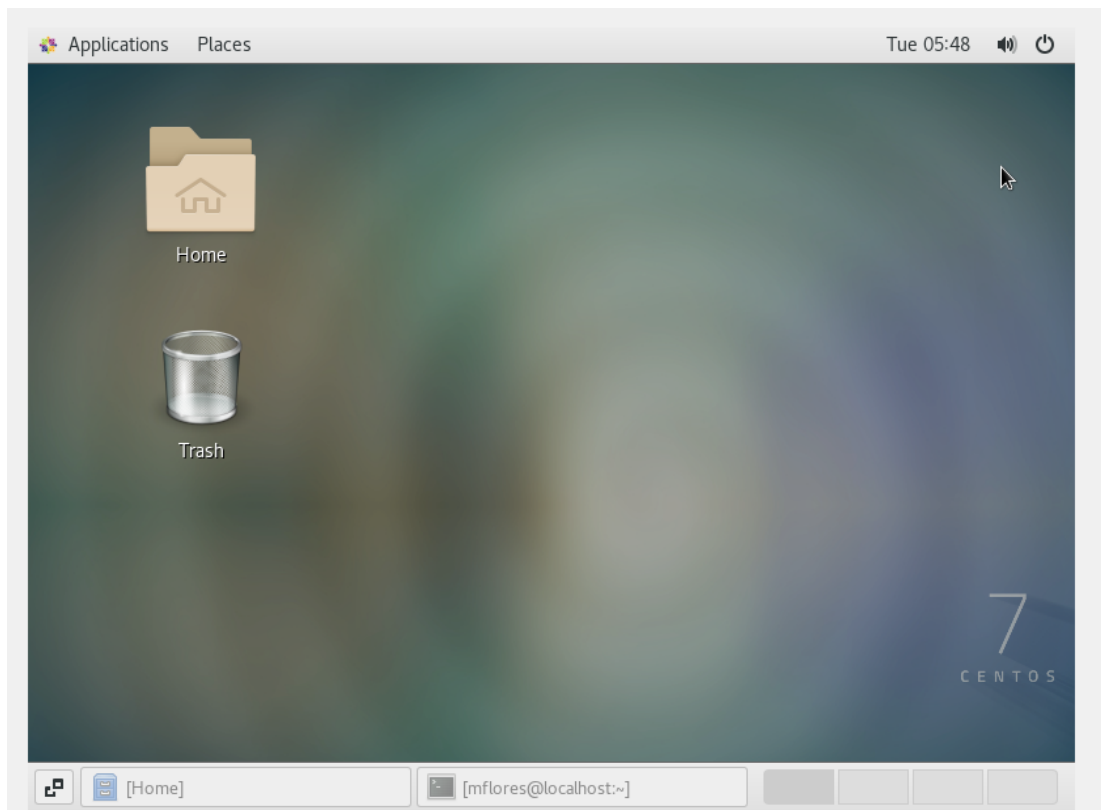
☐ Do Not Add a Virtual Hard Disk

INSTALL GUEST ADDITIONS	
	Hardware
Base Memory	4096
Processor(s)	2
EFI Enable	false
	Disk
Disk Size	35.00 GB
Pre-allocate Full Size	false

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:

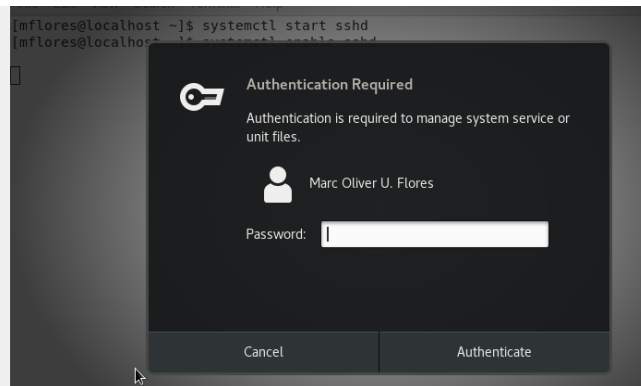
\$ dnf install openssh-server

```
[mflores@localhost ~]$ su -
Password:
[root@localhost ~]# yum install openssh-server
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
* base: mirror.ups.edu.my
* extras: mirror.ups.edu.my
* updates: mirror.ups.edu.my
Package openssh-server-7.4p1-23.el7_9.x86_64 already installed and latest version
Nothing to do
[root@localhost ~]#
```

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

\$ systemctl start sshd

\$ systemctl enable sshd



File Edit View Search Terminal Help

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

[mflores@localhost ~]$
[mflores@localhost ~]$
```

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

\$ systemctl status sshd

```
[mflores@localhost ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; vendor preset: enable
d)
   Active: active (running) since Tue 2023-08-29 05:45:00 EDT; 11min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
    Main PID: 1136 (sshd)
    CGroup: /system.slice/ssh.service
            └─1136 /usr/sbin/sshd -D

Aug 29 05:45:00 localhost.localdomain systemd[1]: Starting OpenSSH server daemon...
Aug 29 05:45:00 localhost.localdomain sshd[1136]: Server listening on 0.0.0.0 port 22.
Aug 29 05:45:00 localhost.localdomain sshd[1136]: Server listening on :: port 22.
Aug 29 05:45:00 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
Hint: Some lines were ellipsized, use -l to show in full.
[mflores@localhost ~]$
```

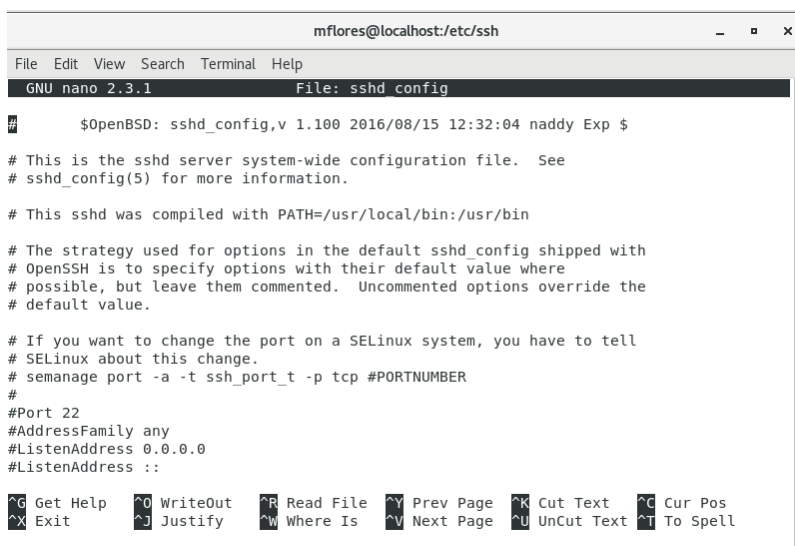
4. Open the SSH port 22 to allow incoming traffic:

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

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

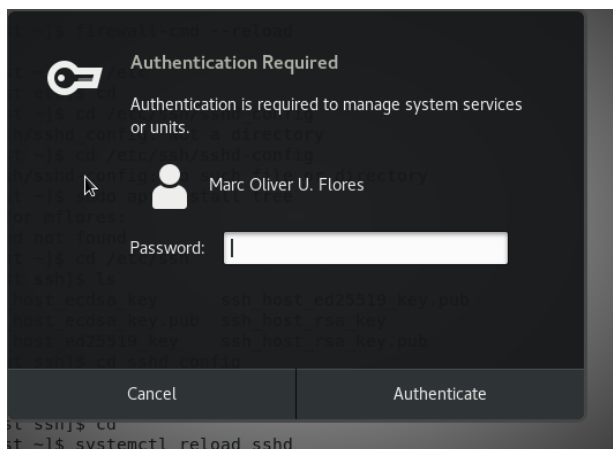
5. Locate the ssh server man config file */etc/ssh/sshd_config* and perform custom configuration. Every time you make any change to the */etc/ssh/sshd-config* configuration file reload the *sshd* service to apply changes:

```
$ systemctl reload sshd
```



```
mflores@localhost:/etc/ssh  
File Edit View Search Terminal Help  
GNU nano 2.3.1 File: sshd config  
# $OpenBSD: sshd_config,v 1.100 2016/08/15 12:32:04 naddy Exp $  
  
# This is the sshd server system-wide configuration file. See  
# sshd_config(5) for more information.  
  
# This sshd was compiled with PATH=/usr/local/bin:/usr/bin  
  
# The strategy used for options in the default sshd_config shipped with  
# OpenSSH is to specify options with their default value where  
# possible, but leave them commented. Uncommented options override the  
# default value.  
  
# If you want to change the port on a SELinux system, you have to tell  
# SELinux about this change.  
# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER  
#  
#Port 22  
#AddressFamily any  
#ListenAddress 0.0.0.0  
#ListenAddress ::  
  
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos  
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

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



Task 3: Copy the Public Key to CentOS

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

```
[mflores@localhost ~]$ ssh
usage: ssh [-1246AaCfGgKkMNNqsTtVvXxYy] [-b bind_address] [-c cipher_spec]
          [-D [bind_address:]port] [-E log_file] [-e escape_char]
          [-F configfile] [-I pkcs11] [-i identity_file]
          [-J [user@]host[:port]] [-L address] [-l login_name] [-m mac_spec]
          [-O ctl_cmd] [-o option] [-p port] [-Q query_option] [-R address]
          [-S ctl_path] [-W host:port] [-w local_tun[:remote_tun]]
          [user@]hostname [command]
```

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

```
[mflores@mflores ~]$ ssh-copy-id workstation
The authenticity of host 'workstation (192.168.164.7)' can't be established.
ECDSA key fingerprint is SHA256:Th0KsfXrZPx0RXAmIxmSoucffnq3/cpSRvo3j3BX760.
ECDSA key fingerprint is MD5:a1:33:c0:83:10:ff:a4:61:ca:5b:91:6b:c7:53:4c:73.
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
that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are promp
it is to install the new keys
mflores@workstation's password:

Number of key(s) added: 1

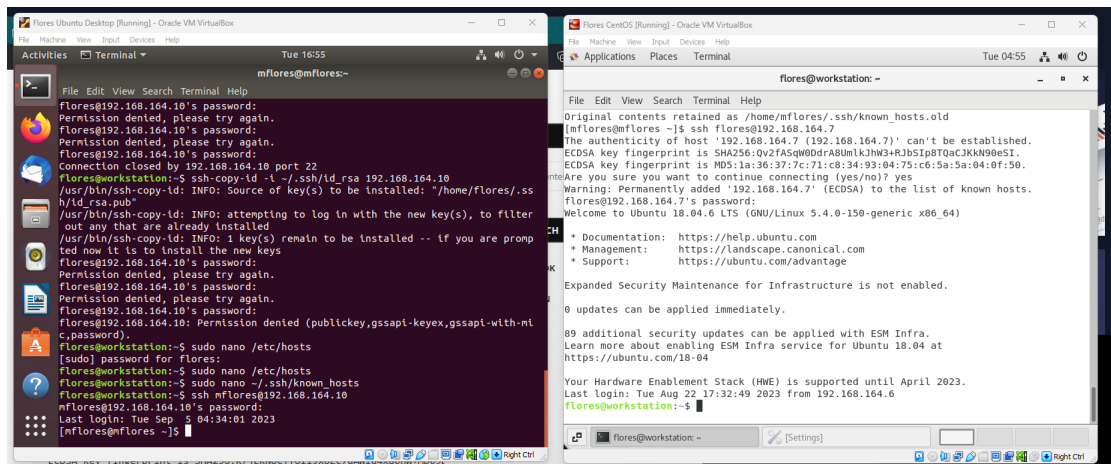
Now try logging into the machine, with:  "ssh 'workstation'"
and check to make sure that only the key(s) you wanted were added.
```

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

```
ssh: connect to host workstation port 22: No route to host
[mflores@mflores ~]$ ls .ssh
authorized_keys  id_rsa  id_rsa.pub  known_hosts
[mflores@mflores ~]$
```

Task 4: Verify ssh remote connection

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



The screenshot shows two terminal windows. The left window is titled 'Flores Ubuntu Desktop (Running) - Oracle VM VirtualBox' and shows a user 'mflores' at 'mflores@mflores:~'. The user runs 'ssh-copy-id -i ~/.ssh/ld_rsa 192.168.164.10', which copies the local public key to the remote host. The right window is titled 'Flores CentOS (Running) - Oracle VM VirtualBox' and shows the user 'mflores' at 'flores@workstation:~'. The user runs 'ssh 192.168.164.7', which successfully establishes an SSH connection to the CentOS machine. The CentOS terminal shows the user's login history and the system's uptime.

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?
 - On choosing the best distribution between the Debian and Red Hat Linux distributions it really depends on what is your needs and preference. Debian is a community-driven distribution that prioritizes user autonomy, personalization, and open-source software. It also supports any architecture or platform that can be run by FreeBSD kernel, Linux, and any GNU tool sets such as GCC. While the Red Hat Enterprise Linux or RHEL is the most popular commercially supported Linux distribution. It uses the .rpm packages and a package manager called dnf, along with its own ecosystem of tools.
 - So for me if you are looking for a free open-source operating system that is very easy to use and has a very large and active community of developers and users Debian is a very good choice. On the other hand if you are looking for a commercially supported operating system that is more stable and very secure than Debian you should go for Red Hat Enterprise Linux (RHEL) is a very good choice.

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

- The difference between Debian and Red Hat Linux distributions are their package management systems and tools. In debian it uses the debian package format (.deb). while Red Hat Linux uses the RPM (.rpm) package format. Also based on my answer in question 1 the difference between this 2 distributions is that Debian is a community-driven distribution that prioritizes user autonomy, personalization, and open-source software. While the Red Hat Linux distribution is a commercially supported operating system that is more stable and secure than the debian.

Conclusion:

- In this activity I was able to install a different type of OS and also configure the remote SSH from a remote computer to a CentOS. Also I was able to understand the difference between Debian and the Red Hat Linux distribution.