Name: Dean Lenard D Perez	Date Performed:
Course/Section: CPE31S21	Date Submitted: 13/09/2024
Instructor:	Semester and SY: 2024-2025

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.

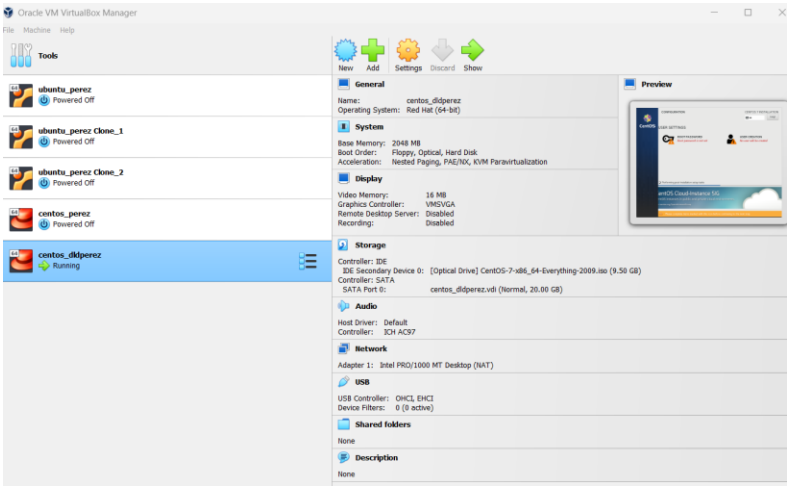
Both offer full-feature package management with network-based repository support, dependency checking and resolution, etc.. If you're familiar with one but not the other, you may have a little trouble switching over, but they're not overwhelmingly different. They both have similar features, just available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

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

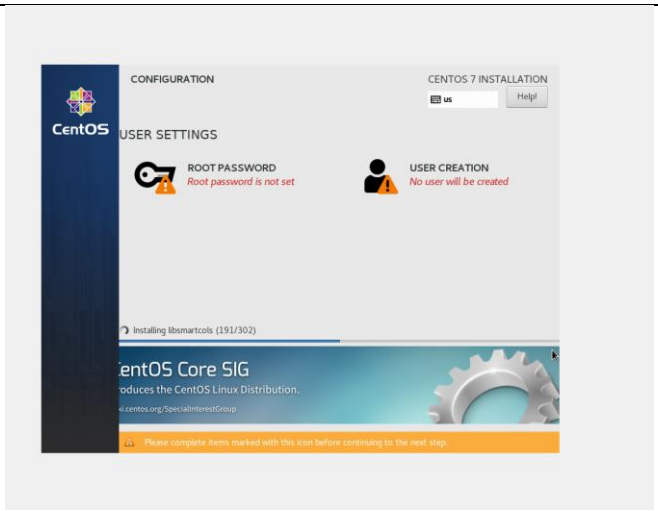
1. Download the image of the CentOS here: http://mirror.rise.ph/centos/7.9.2009/isos/x86_64/

CentOS-7-x86_64-Everything-2009

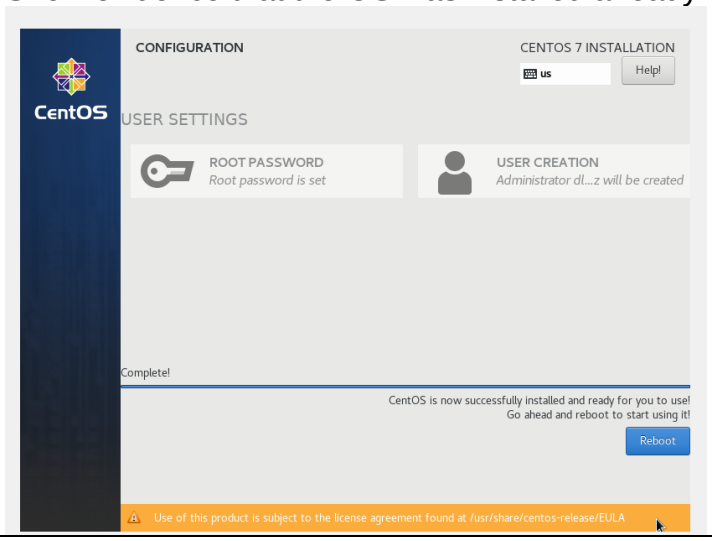
2. Create a VM machine with 2 Gb RAM and 20 Gb HD.



3. Install the downloaded image.



4. Show evidence that the OS was installed already.



```
CentOS Linux 7 (Core)
Kernel 3.10.0-1160.el7.x86_64 on an x86_64

localhost login: dldperez
Password:
Login incorrect

localhost login: dldperez
Password:
Last failed login: Thu Sep 12 20:40:31 PST 2024 on tty1
There was 1 failed login attempt since the last successful login.
[dldperez@localhost ~]$
```

Task 2: Install the SSH server package *openssh*

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

```
[dldperez@localhost ~]$ rpm -qa | grep openssh-server
openssh-server-7.4p1-21.el7.x86_64
[dldperez@localhost ~]$
```

Opensssh-server is already installed.

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

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

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

```
[dldperez@localhost ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2024-09-12 20:40:10 PST; 11min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
 Main PID: 1021 (sshd)
   CGroup: /system.slice/sshd.service
           └─1021 /usr/sbin/sshd -D

Sep 12 20:40:09 localhost.localdomain systemd[1]: Starting OpenSSH server daemon...
Sep 12 20:40:10 localhost.localdomain sshd[1021]: Server listening on 0.0.0.0 port 22.
Sep 12 20:40:10 localhost.localdomain sshd[1021]: Server listening on :: port 22.
Sep 12 20:40:10 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
[dldperez@localhost ~]$ _
```

It is active and running.

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

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

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

```
File  Machine  View  Input  Devices  Help
#       $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 IS NEW ISNERT YHOHOHOHOHOH
# 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 ::

HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_dsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
SyslogFacility AUTHPRIV
#LogLevel INFO

# Authentication:
```

The command that was used is: sudo vi /etc/ssh/sshd_config
i to insert a change, esc to exit insert mode.
:wq to save and exit.

```
[root@localhost dldperez]# ssh localhost
root@localhost's password:
Last login: Thu Sep 12 21:36:54 2024
[root@localhost ~]#
```

Ssh is working

Task 3: Copy the Public Key to CentOS

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

```
[root@localhost dldperez]# ssh -v
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]
[root@localhost dldperez]# _
```

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

```
[dldperez@localhost ~]$ ssh-keygen -t rsa -b 2048
Generating public/private rsa key pair.
Enter file in which to save the key (/home/dldperez/.ssh/id_rsa):
Created directory '/home/dldperez/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/dldperez/.ssh/id_rsa.
Your public key has been saved in /home/dldperez/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:ypJDH1UC+JCahm5xtovWu3Saw2rEynteer6BUXW4yCc dldperez@localhost.localdomain
The key's randomart image is:
+---[RSA 2048]----+
|     o.oo..      |
|    + ...o       |
| . o.+. o        |
|..+o.E.+         |
|o.+.o + S        |
| = oo+ o         |
|= +o=++          |
|.=.***.          |
|oo=B*o.          |
+----[SHA256]-----+
[dldperez@localhost ~]$
+----[SHA256]-----+
[dldperez@localhost ~]$ ls -l ~/.ssh
total 8
-rw-------. 1 dldperez dldperez 1679 Sep 12 21:53 id_rsa
-rw-r--r--. 1 dldperez dldperez  412 Sep 12 21:53 id_rsa.pub
[dldperez@localhost ~]$
[dldperez@localhost ~]$ ssh-copy-id dldperez@localhost.localdomain
/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/dldperez/.ssh/id_rsa.pub"
The authenticity of host 'localhost.localdomain (::1)' can't be established.
ECDSA key fingerprint is SHA256:t4KzhYRNrCjFjKIWgwrOpKgVCwG+GTgwF+EmLKodTks.
ECDSA key fingerprint is MD5:f3:10:b4:f5:89:9f:6c:22:65:f5:25:c2:24:91:0d:fe.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already
 installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install
the new keys
dldperez@localhost.localdomain's password:

Number of key(s) added: 1

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

[dldperez@localhost ~]$
```

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

```
[dldperez@localhost ~]$ ls -ld ~/.ssh
drwx------. 2 dldperez dldperez 80 Sep 12 21:56 /home/dldperez/.ssh
[dldperez@localhost ~]$ ls -l ~/.ssh/authorized_keys
-rw-------. 1 dldperez dldperez 412 Sep 12 21:56 /home/dldperez/.ssh/authorized_keys
[dldperez@localhost ~]$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDQQtb1VkczGPrKl4rcMUxj6XpFSxNMTT2kPtKwSGKBs1klOpSvUn2dm083WBMw
vbh6CxI4IfZAlsfLUnT9V8Log0U/K2GeMhhhaX2rh3t5teg2g5LWYj//XxIOB0W5Ht5JqG4f7P/bYnon+V88xtBQBBhVZMa2odet
uN1vXhnKwGeQsqmCwwDP2kaEiFdsf+7DkIN4KHWhQ69pXwJ8DEcH5v1dF63BffJR9TtiM+lC+crUbQXywoHar+fz+wlw/vWu0Mf0
18C0HM+qxY38uurbmPWRaWWGM5J7fjzP7yoD1bmnk2JWV4ZPHPmZ7wXTPF7/LWb7+7GOhOZ5xNWOW8gp dldperez@localhost.
localdomain
[dldperez@localhost ~]$ _
```

Task 4: Verify ssh remote connection

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

```
[dldperez@localhost ~]$ ssh dldperez@192.168.56.107
The authenticity of host '192.168.56.107 (192.168.56.107)' can't be established.
ECDSA key fingerprint is SHA256:t4KzhYRNrCjFjKIWgwrOpKgVCwG+GTgwF+EmLKodTks.
ECDSA key fingerprint is MD5:f3:10:b4:f5:89:9f:6c:22:65:f5:25:c2:24:91:0d:fe.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.56.107' (ECDSA) to the list of known hosts.
Last login: Thu Sep 12 22:14:45 2024
```

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?

   I think we should look for the package management and system administration tools because it is the one that you will use a lot. If you don't know the commands, you might struggle a little. It really depends on what your needs are or what you are capable of buying. You should use the one that you are most comfortable.

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

   The main difference between Debian and Red Hat Linux distributions is they focus on different audience. Debian is based on the community that needs free and reliable distribution while Red had focuses more on enterprises and they offer supports professionally.