Name:John Edward Miles D. Espiritu	Date Performed:01/29/2024
Course/Section:CPE232-31S1	Date Submitted:01/29/2024
Instructor: Sir Jonathan Taylar	Semester and SY:2nd sem/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.

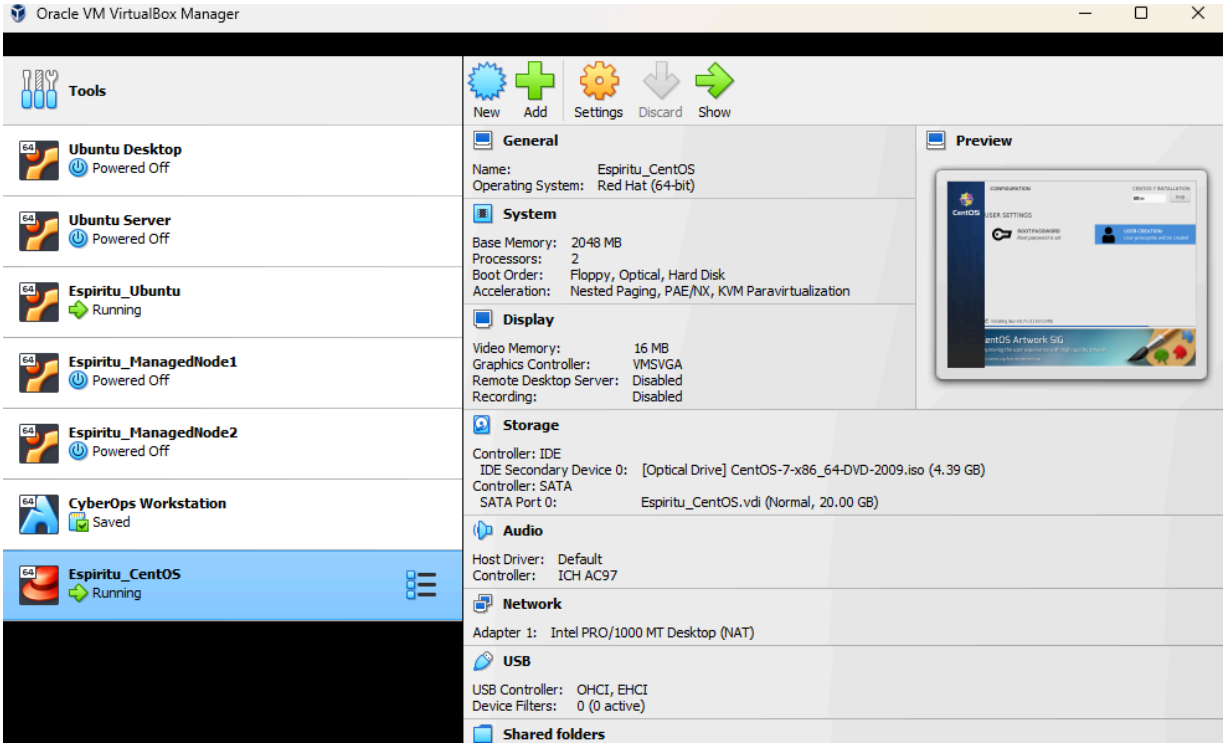
CentOS uses the RPM package format and YUM/DNF as the package manager.

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

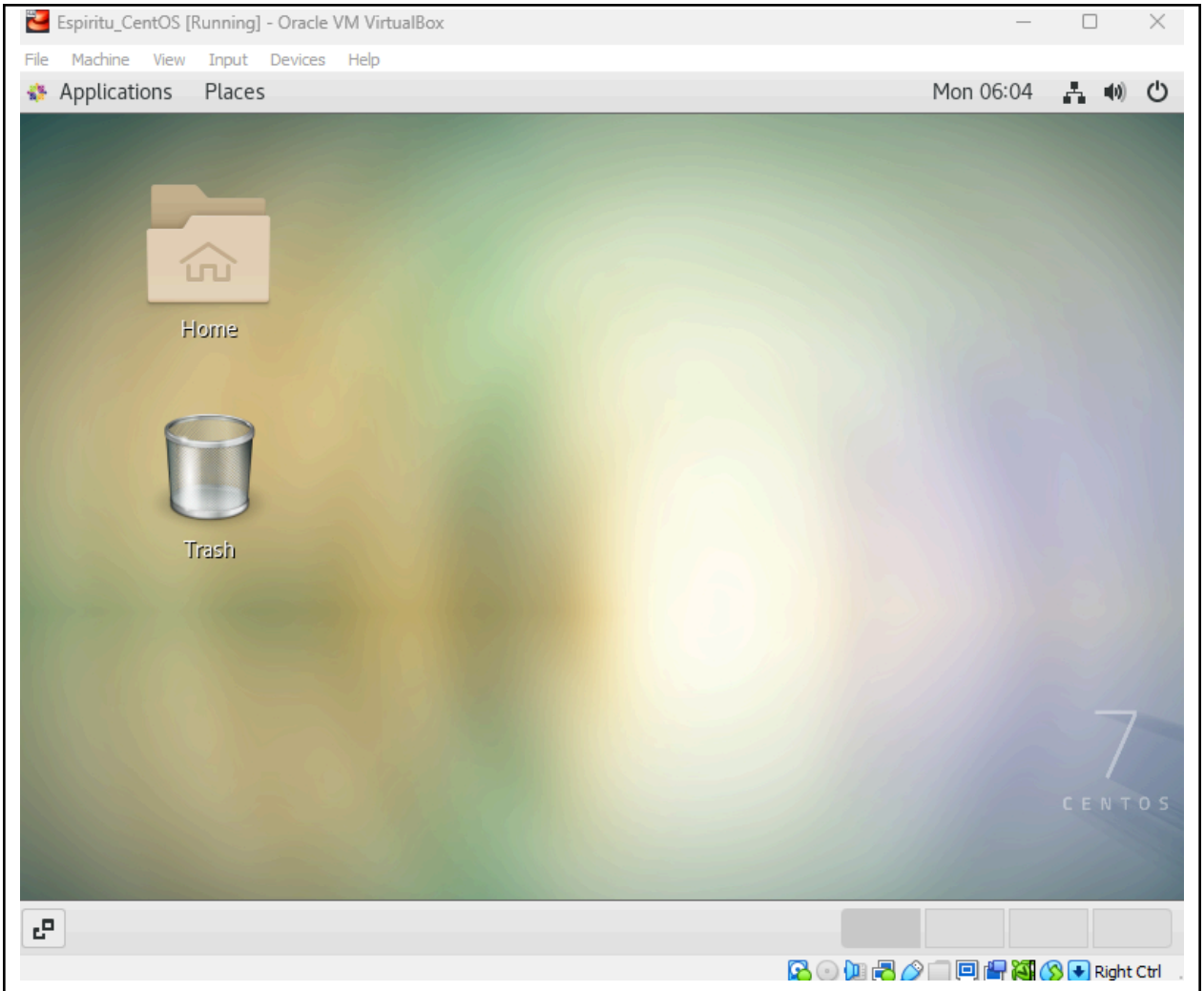
Both offer full-feature package management with network-based repository support, dependency checking and resolution, etc.. If you're familiar with one but not the other, you may have a little trouble switching over, but they're not overwhelmingly different. They both have similar features, just available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

```
[root@localhost jemespiritu]# yum install openssh-server
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirror.rise.ph
 * extras: mirror.rise.ph
 * updates: mirror.rise.ph

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
--> Running transaction check
---> Package openssh-clients.x86_64 0:7.4p1-21.el7 will be updated
---> Package openssh-clients.x86_64 0:7.4p1-23.el7_9 will be an update
--> Finished Dependency Resolution

Dependencies Resolved


Dependencies Resolved

================================================================================
 Package                 Arch           Version                   Repository       Size
================================================================================
Updating:
 openssh-server          x86_64         7.4p1-23.el7_9            updates         459 k
Updating for dependencies:
 openssh                 x86_64         7.4p1-23.el7_9            updates         510 k
 openssh-clients         x86_64         7.4p1-23.el7_9            updates         655 k

Transaction Summary
================================================================================
Upgrade  1 Package (+2 Dependent packages)

Total download size: 1.6 M
Is this ok [y/d/N]: Exiting on user command
Your transaction was saved, rerun it with:
 yum load-transaction /tmp/yum_save_tx.2024-01-29.06-31.UjtujO.yumtx
[root@localhost jemespiritu]#
```

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

   ```
   [root@localhost jemespiritu]# sudo systemctl start sshd
   ```

   $ systemctl enable sshd

   ```
   [root@localhost jemespiritu]# sudo systemctl enable sshd
   ```

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

```
[root@localhost jemespiritu]# sudo 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 Mon 2024-01-29 06:02:13 EST; 40min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
 Main PID: 1220 (sshd)
   CGroup: /system.slice/sshd.service
           └─1220 /usr/sbin/sshd -D

Jan 29 06:02:13 localhost.localdomain systemd[1]: Starting OpenSSH server daemon...
Jan 29 06:02:13 localhost.localdomain sshd[1220]: Server listening on 0.0.0.0 port 22.
Jan 29 06:02:13 localhost.localdomain sshd[1220]: Server listening on :: port 22.
Jan 29 06:02:13 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

```
[root@localhost jemespiritu]# sudo firewall-cmd --zone=public --permanent --add-service
=ssh
Warning: ALREADY_ENABLED: ssh
success
[root@localhost jemespiritu]#
```

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:

```
[root@localhost jemespiritu]# 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

```
[root@localhost jemespiritu]# sudo systemctl reload sshd
```

Task 3: Copy the Public Key to CentOS

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

```
jem@ManagedNode:~$ sudo apt install openssh-server
[sudo] password for jem:
Reading package lists... Done
Building dependency tree
Reading state information... Done
openssh-server is already the newest version (1:7.6p1-4ubuntu0.7).
The following package was automatically installed and is no longer required:
  libllvm7
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

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

```
jem@ManagedNode:~$ ssh-copy-id -i ~/.ssh/id_rsa jemespiritu@192.168.56.120
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/jem/.ssh/i
d_rsa.pub"
The authenticity of host '192.168.56.120 (192.168.56.120)' can't be established
.
ECDSA key fingerprint is SHA256:nZg7hmFffhOkeuzcyr3mQZRKa0eX9gUKv3szNCAR+ng.
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
jemespiritu@192.168.56.120's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'jemespiritu@192.168.56.120'"
and check to make sure that only the key(s) you wanted were added.
```

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

```
[jemespiritu@localhost ~]$ cd ~/.ssh
[jemespiritu@localhost .ssh]$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQCT3tnCFneUAxErhkFyQJzoTqdMqRhvVtpaonxr7CWFcW1oquh
UMFr+inrmXlQ5Nikc9vCBezTBbMtBx24+qq+xjyTBujdvJix7KFfyBPH/fyRbbBRwcWO6bh5k2lgtjWTTsmyi96
P96yTuNoN4mDjayAHE2MUr9eqXI2JjYHUa1qUWU7+dKcKEid4tM71AxnsX0MzgkGDJiRHFltT6um+1KZgqGLzWW
uH3h8RnxUprzzDUWeXsYSBGl0RmxJjhPCfPbp9imigMN6+EGGRXqbFSPdXMBqv/v+zXj0XwOd4bSTuAgZFe5TzH
J/ERlKi50sK5vChC+R24TFILa2S6yg+TCiR2OdUcbbmBDyfk3gVcMzYfSO37JmkVy4kz9fOPEjKRrLkg3nAzcUP
KcC13gQo+M2UNSc3NHW6JaE7YmKuX6fvu+RBsQv4bcHKFkcZmivDg7ZR7PDMgmJa+bxlMFPdrrymJfQxNMvOdqj
xRJczZcaRJ8ABwjLFREn2cUxfMZQTyOgMnHxsk/MpcOQ1LtgdclY7GXLORKiSrI88zb+k9QXB2CuoxKrvly6MMu
/G4gepzCjnR4rPNgn8RdND38mIL/EmD505MxHxWtvB/THbv0i+wYuS0VXyrxmLBOm9NggPsk14jgxpuEUzo5wHJ
zaL39iPIR09/OskiyVlMD5DAeQ== jem@ManagedNode
[jemespiritu@localhost .ssh]$
```

Task 4: Verify ssh remote connection

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

```
jem@ManagedNode:~$ ssh jemespiritu@192.168.56.120
Last login: Mon Jan 29 07:00:36 2024
[jemespiritu@localhost ~]$
```

2. Show evidence that you are connected.

```
jem@ManagedNode:~$ ssh jemespiritu@192.168.56.120
Last login: Mon Jan 29 07:17:58 2024 from 192.168.56.117
[jemespiritu@localhost ~]$ ls -a
.               .bash_profile  .dbus       .esd_auth      Music      Templates
..              .bashrc        Desktop     .ICEauthority  Pictures   Videos
.bash_history   .cache         Documents   .local         Public
.bash_logout    .config        Downloads   .mozilla       .ssh
[jemespiritu@localhost ~]$ ls ~/.ssh
authorized_keys  id_rsa  id_rsa.pub
[jemespiritu@localhost ~]$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQCT3tnCFneUAxErhkFyQJzoTqdMqRhvVtpaonxr7CW
FcW1oquhUMFr+inrmXlQ5Nikc9vCBezTBbMtBx24+qq+xjyTBujdvJix7KFfyBPH/fyRbbBRwcWO6bh
5k2lgtjWTTsmyi96P96yTuNoN4mDjayAHE2MUr9eqXI2JjYHUa1qUWU7+dKcKEid4tM71AxnsX0Mzgk
GDJiRHFltT6um+1KZgqGLzWWuH3h8RnxUprzzDUWeXsYSBGl0RmxJjhPCfPbp9imigMN6+EGGRXqbFS
PdXMBqv/v+zXj0XwOd4bSTuAgZFe5TzHJ/ERlKi50sK5vChC+R24TFILa2S6yg+TCiR2OdUcbbmBDyf
k3gVcMzYfSO37JmkVy4kz9fOPEjKRrLkg3nAzcUPKcC13gQo+M2UNSc3NHW6JaE7YmKuX6fvu+RBsQv
4bcHKFkcZmivDg7ZR7PDMgmJa+bxlMFPdrrymJfQxNMvOdqjxRJczZcaRJ8ABwjLFREn2cUxfMZQTyO
gMnHxsk/MpcOQ1LtgdclY7GXLORKiSrI88zb+k9QXB2CuoxKrvly6MMu/G4gepzCjnR4rPNgn8RdND3
8mIL/EmD505MxHxWtvB/THbv0i+wYuS0VXyrxmLBOm9NggPsk14jgxpuEUzo5wHJzaL39iPIR09/Osk
iyVlMD5DAeQ== jem@ManagedNode
[jemespiritu@localhost ~]$ logot
bash: logot: command not found...
[jemespiritu@localhost ~]$ logout
Connection to 192.168.56.120 closed.
jem@ManagedNode:~$
```

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?**
   - There are a number of things to take into account while deciding between the Red Hat and Debian Linux distributions. Software licensing: Debian is a free and open-source distribution that does not require a subscription, whereas Red Hat Enterprise Linux (RHEL) is a commercial distribution that does. Software Packages / Feature Updates: RHEL offers 14,000 software packages in its repository, whereas Debian has around 59,000 software packages.

2. **What are the main diffence between Debian and Red Hat Linux distributions?**
   - Two well-known Linux distributions with different features are Debian and Red Hat. A community of volunteers created Debian, which prioritizes stability and uses a cautious release cycle. Installing and removing software is made easier by its package management system, APT. Debian is renowned for its dedication to open-source and free software. On the other hand, Red Hat, a for-profit organization, manages the development of Red Hat Enterprise Linux (RHEL), emphasizing enterprise settings. Fedora, the version backed by the community, releases updates more quickly than RHEL, which has a longer lifecycle and offers more support. Red Hat's default init system is systemd, and it manages packages via YUM/DNF. Red Hat focuses on mission-critical systems and

offers commercial support and a subscription-based strategy, whereas Debian is flexible and extensively utilized.