

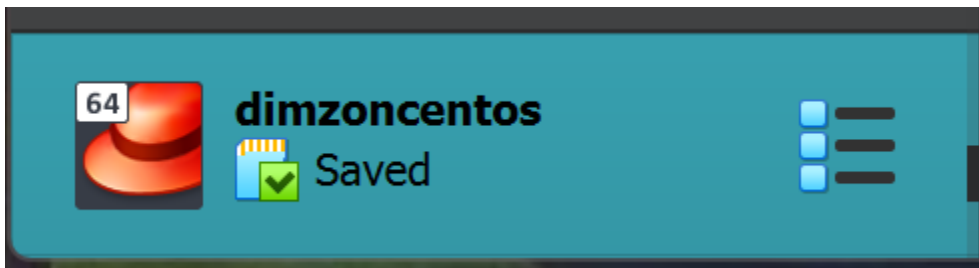
sudo

<b>Name: MARK ALLEN RHOY DIMZON</b>	<b>Date Performed: 09/19/23</b>
<b>Course/Section: CPE 232 CPE31S4</b>	<b>Date Submitted: 09/19/23</b>
<b>Instructor: DR. JONATHAN V. TAYLAR</b>	<b>Semester and SY: 1ST SEM 2023-2024</b>
<b>Activity 3: Install SSH server on CentOS or RHEL 8</b>	
<b>1. Objectives:</b> 1.1 Install Community Enterprise OS or Red Hat Linux OS 1.2 Configure remote SSH connection from remote computer to CentOS/RHEL-8	
<b>2. Discussion:</b>  <b>CentOS vs. Debian: Overview</b>  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.  <b>CentOS vs. Debian: Architecture</b>  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.  <b>CentOS vs. Debian: Package Management</b>  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/](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:

```

root@localhost ~]# sudo yum install dnf
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
Resolving Dependencies
--> Running transaction check
--> Package dnf.noarch 0:4.0.9.2-2.el7_9 will be installed
--> Processing Dependency: python2-dnf = 4.0.9.2-2.el7_9 for package: dnf-4.0.9.2-2.el7_9.noarch
--> Running transaction check
--> Package python2-dnf.noarch 0:4.0.9.2-2.el7_9 will be installed
--> Processing Dependency: dnf-data = 4.0.9.2-2.el7_9 for package: python2-dnf-4.0.9.2-2.el7_9.noarch
--> Processing Dependency: python2-libdnf >= 0.22.5 for package: python2-dnf-4.0.9.2-2.el7_9.noarch
--> Processing Dependency: python2-libcomps >= 0.1.8 for package: python2-dnf-4.0.9.2-2.el7_9.noarch
--> Processing Dependency: python2-hawkey >= 0.22.5 for package: python2-dnf-4.0.9.2-2.el7_9.noarch
libdnf.x86_64 0:0.22.5-2.el7_9
librepo.x86_64 0:1.8.1-8.el7_9
python-enum34.noarch 0:1.0.4-1.el7
python2-hawkey.x86_64 0:0.22.5-2.el7_9
python2-libdnf.x86_64 0:0.22.5-2.el7_9
libmodulemd.x86_64 0:1.6.3-1.el7
libsolv.x86_64 0:0.6.34-4.el7
python2-dnf.noarch 0:4.0.9.2-2.el7_9
python2-libcomps.x86_64 0:0.1.8-14.el7

Complete!
```

```
$ dnf install openssh-server
```

```

Complete!
[root@localhost ~]# sudo dnf install openssh-server
CentOS-7 - Base                               6.0 MB/s | 10 MB      00:01
CentOS-7 - Updates                             10 MB/s | 28 MB      00:02
CentOS-7 - Extras                             655 kB/s | 360 kB     00:00
Package openssh-server-7.4p1-21.el7.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!

```

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

```
$ systemctl start sshd
```

```
$ systemctl enable sshd
```

```

Complete!
[root@localhost ~]# systemctl start sshd
[root@localhost ~]# systemctl enable sshd

```

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

```
$ systemctl status sshd
```

```

[root@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 Mon 2023-09-18 14:11:56 EDT; 11min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
  Main PID: 1245 (sshd)
    CGroup: /system.slice/ssh.service
            └─1245 /usr/sbin/sshd -D

```

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

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

```
$ firewall-cmd --reload
```

```

Note: Some lines were truncated, use -t to show all rules.
[root@localhost ~]# firewall-cmd --zone=public --permanent --add-service=ssh
Warning: ALREADY_ENABLED: ssh
success
[root@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
```

```

success
[root@localhost ~]# systemctl reload sshd
[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*.

```
[sudo] password for dimzon:
dimzon@localmachine:~$ ssh dimzon@centos
The authenticity of host 'centos (192.168.51.10)' can't be established.
ECDSA key fingerprint is SHA256:XUK96DSmRNvND6KEPiw/YgrbdQKcmT1r2BEMAHTtn2Q.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'centos,192.168.51.10' (ECDSA) to the list of known hosts.
dimzon@centos's password:
Last login: Mon Sep 18 01:10:31 2023
[dimzon@localhost ~]$ logout
Connection to centos closed.
dimzon@localmachine:~$ ssh-copy-id -i ~/.ssh/id_rsa dimzon@centos
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/dimzon/.ssh/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 prompted now it is to install the new keys
dimzon@centos's password:

Number of key(s) added: 1

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

#### Task 4: Verify ssh remote connection

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

```
dimzon@localmachine: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hosts

127.0.0.1    localhost
127.0.0.1    dimzon
192.168.56.105 cs1
192.168.56.106 cs2
192.168.56.107 centos

# The following lines are desirable for IPv6 capable hosts
::1        ip6-localhost ip6-loopback
fe00::0    ip6-localnet
ff00::0    ip6-mcastprefix
ff02::1    ip6-allnodes
ff02::2    ip6-allrouters
```

2. Show evidence that you are connected.

```
dimzon@localmachine:~$ sudo nano /etc/hosts
dimzon@localmachine:~$ ssh dimzon@192.168.56.107
Last login: Mon Sep 18 14:34:44 2023 from 192.168.56.104
dimzon@centos:~$
```

**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?
  - For me it's the debian distribution because its more commonly used and can easily navigate through its ui and is more optimized.
2. What are the main differences between Debian and Red Hat Linux distributions?
  - Debian is an open-source, community-driven Linux distribution renowned for reliability. The enterprise-focused distribution Red Hat, on the other hand, is known for its security.