Name: Espiritu, Diego Angelo G.	**Date Performed: 07/09/2023**
Course/Section: CPE232/CPE31S6	**Date Submitted: 07/09/2023**
Instructor: Dr. Jonathan Vidal Taylar	**Semester and SY: 1st Semester**

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.

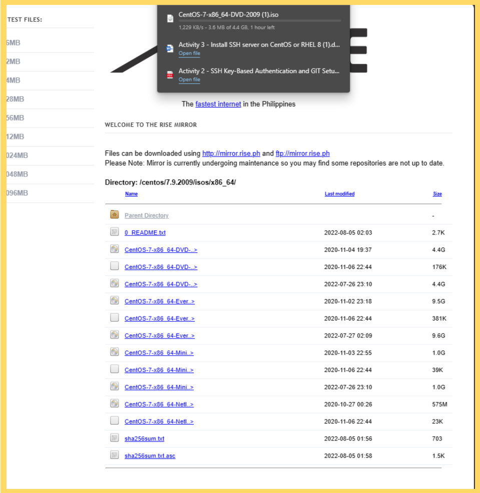
CentOS uses the RPM package format and YUM/DNF as the package manager.

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/



TEST FILES:

6MB
2MB
4MB
28MB
56MB
12MB
024MB
048MB
096MB

CentOS-7-x86_64-DVD-2009 (1).iso
1,229 KB/s - 3.6 MB of 4.4 GB, 1 hour left

Activity 3 - Install SSH server on CentOS or RHEL 8 (1).d...
Open file

Activity 2 - SSH Key-Based Authentication and GIT Setu...
Open file

The fastest internet in the Philippines

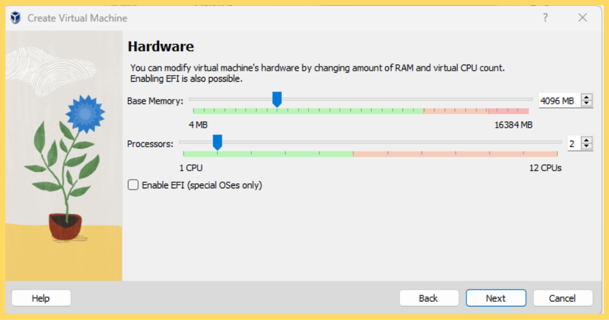
WELCOME TO THE RISE MIRROR

Files can be downloaded using http://mirror.rise.ph and ftp://mirror.rise.ph
Please Note: Mirror is currently undergoing maintenance so you may find some repositories are not up to date.

Directory: /centos/7.9.2009/isos/x86_64/

Name	Last modified	Size
Parent Directory		-
0_README.txt	2022-08-05 02:03	2.7K
CentOS-7-x86_64-DVD-..>	2020-11-04 19:37	4.4G
CentOS-7-x86_64-DVD-..>	2020-11-06 22:44	176K
CentOS-7-x86_64-DVD-..>	2022-07-26 23:10	4.4G
CentOS-7-x86_64-Ever..>	2020-11-02 23:18	9.5G
CentOS-7-x86_64-Ever..>	2020-11-06 22:44	381K
CentOS-7-x86_64-Ever..>	2022-07-27 02:09	9.6G
CentOS-7-x86_64-Mini..>	2020-11-03 22:55	1.0G
CentOS-7-x86_64-Mini..>	2020-11-06 22:44	39K
CentOS-7-x86_64-Mini..>	2022-07-26 23:10	1.0G
CentOS-7-x86_64-NetI..>	2020-10-27 00:26	575M
CentOS-7-x86_64-NetI..>	2020-11-06 22:44	23K
sha256sum.txt	2022-08-05 01:56	703
sha256sum.txt.asc	2022-08-05 01:58	1.5K

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



Create Virtual Machine

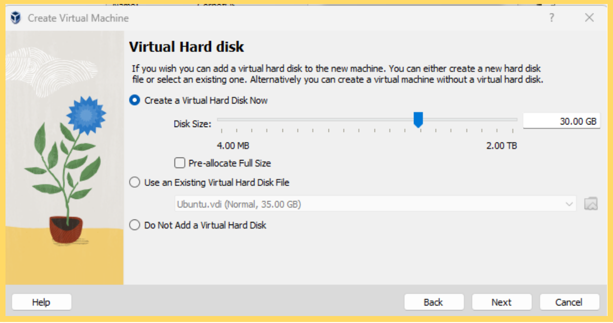
Hardware

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 (4 MB – 16384 MB)

Processors: 2 (1 CPU – 12 CPUs)

□ Enable EFI (special OSes only)



Create Virtual Machine

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: 30.00 GB (4.00 MB – 2.00 TB)

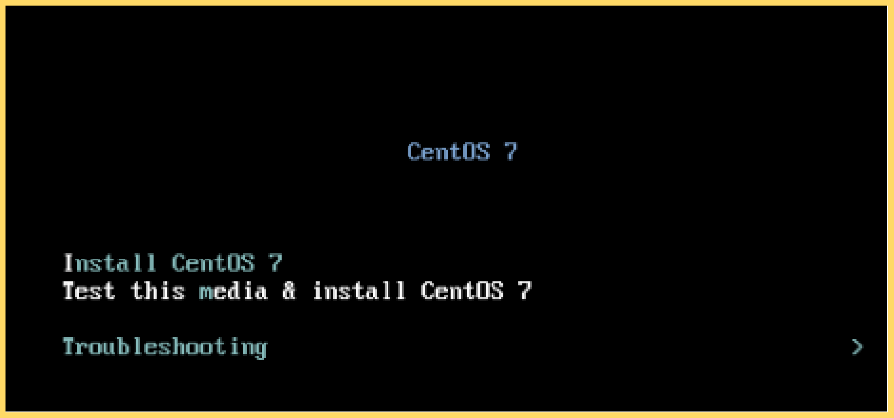
□ Pre-allocate Full Size

○ Use an Existing Virtual Hard Disk File

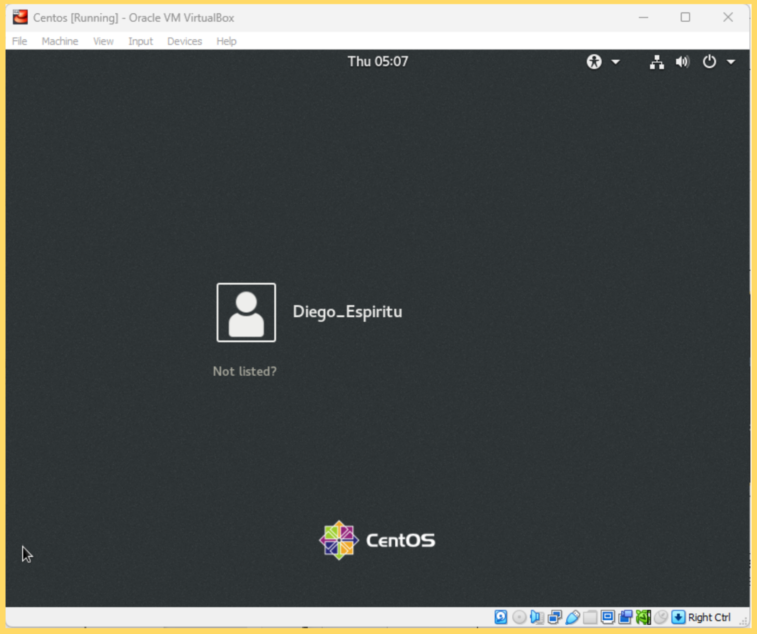
Ubuntu.vdi (Normal, 35.00 GB)

○ Do Not Add a Virtual Hard Disk

3. Install the downloaded image.



4. Show evidence that the OS was installed already.

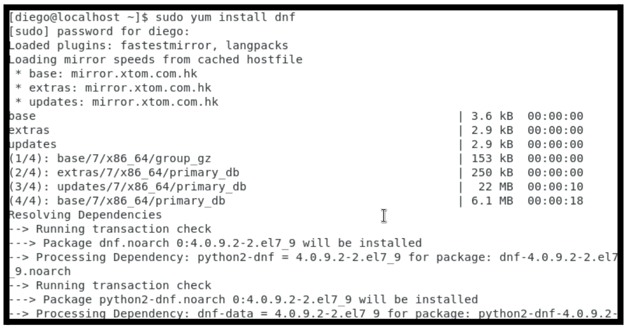


Task 2: Install the SSH server package *openssh*

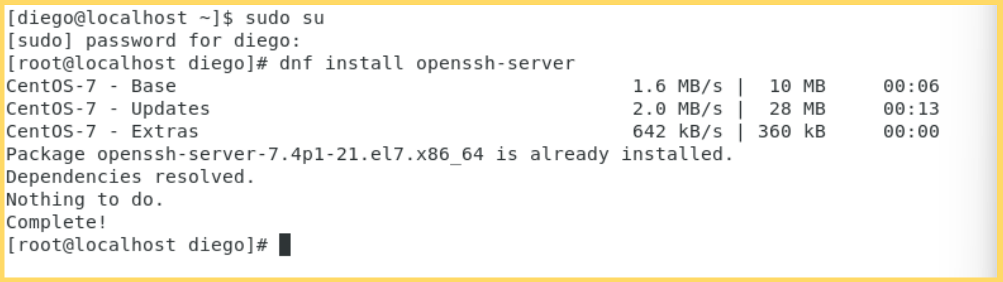
1. Install the ssh server package *openssh* by using the *dnf* command:

   $ dnf install openssh-server

   (If dnf doesn't work)

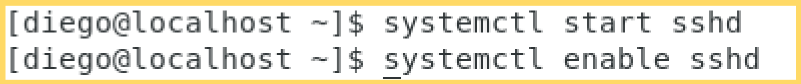


```
[diego@localhost ~]$ sudo yum install dnf
[sudo] password for diego:
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirror.xtom.com.hk
 * extras: mirror.xtom.com.hk
 * updates: mirror.xtom.com.hk
base                                                     | 3.6 kB  00:00:00
extras                                                   | 2.9 kB  00:00:00
updates                                                  | 2.9 kB  00:00:00
(1/4): base/7/x86_64/group_gz                            | 153 kB  00:00:00
(2/4): extras/7/x86_64/primary_db                        | 250 kB  00:00:00
(3/4): updates/7/x86_64/primary_db                       |  22 MB  00:00:10
(4/4): base/7/x86_64/primary_db                          | 6.1 MB  00:00:18
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-
```



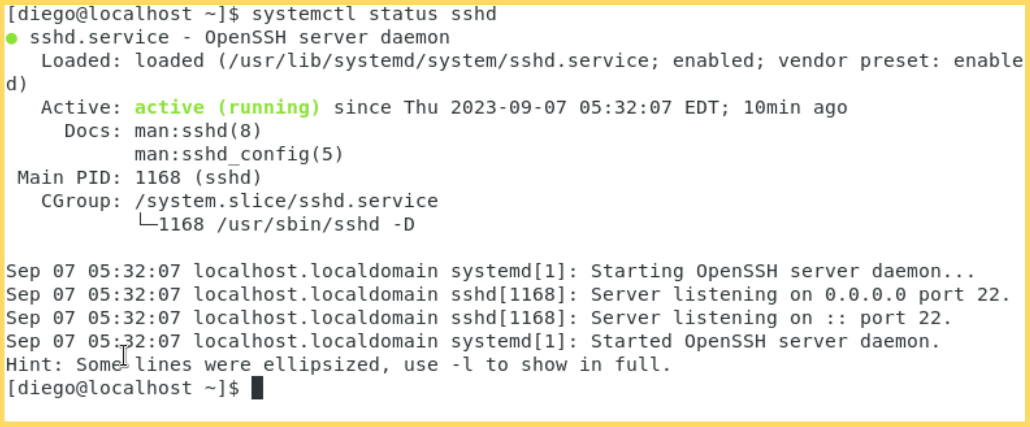
```
[diego@localhost ~]$ sudo su
[sudo] password for diego:
[root@localhost diego]# dnf install openssh-server
CentOS-7 - Base                                           1.6 MB/s |  10 MB     00:06
CentOS-7 - Updates                                        2.0 MB/s |  28 MB     00:13
CentOS-7 - Extras                                         642 kB/s | 360 kB     00:00
Package openssh-server-7.4p1-21.el7.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@localhost diego]# █
```

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



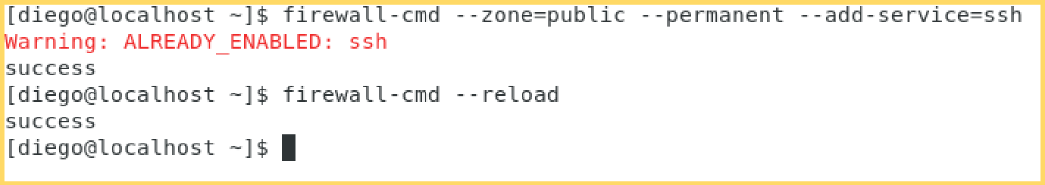
```
[diego@localhost ~]$ systemctl start sshd
[diego@localhost ~]$ systemctl enable sshd
```

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



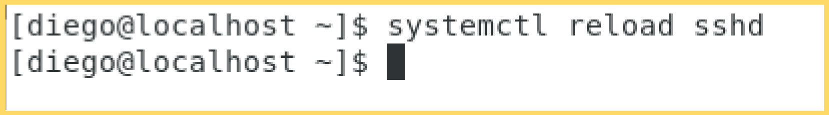
```
[diego@localhost ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enable
d)
   Active: active (running) since Thu 2023-09-07 05:32:07 EDT; 10min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
 Main PID: 1168 (sshd)
   CGroup: /system.slice/sshd.service
           └─1168 /usr/sbin/sshd -D

Sep 07 05:32:07 localhost.localdomain systemd[1]: Starting OpenSSH server daemon...
Sep 07 05:32:07 localhost.localdomain sshd[1168]: Server listening on 0.0.0.0 port 22.
Sep 07 05:32:07 localhost.localdomain sshd[1168]: Server listening on :: port 22.
Sep 07 05:32:07 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
Hint: Some lines were ellipsized, use -l to show in full.
[diego@localhost ~]$ █
```

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



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

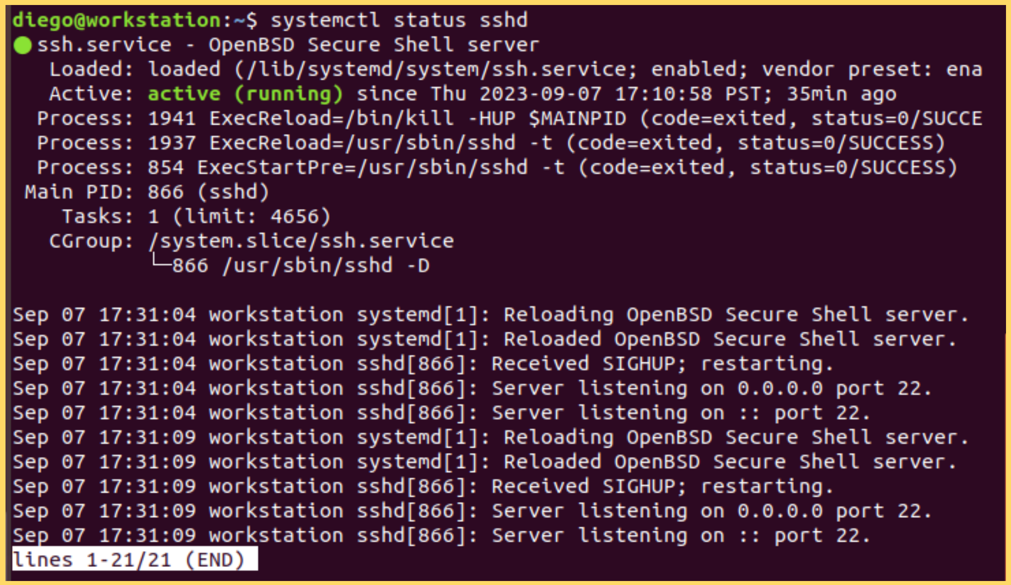
5. Locate the ssh server man config file */etc/ssh/sshd_config* and perform custom configuration. Every time you make any change to the */etc/ssh/sshd-config* configuration file reload the *sshd* service to apply changes:
   $ systemctl reload sshd



```
[diego@localhost ~]$ systemctl reload sshd
[diego@localhost ~]$ █
```

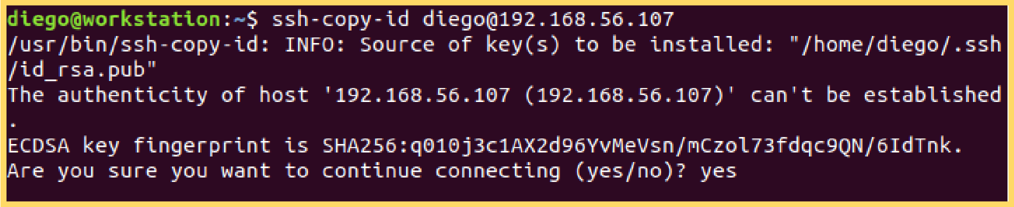
Task 3: Copy the Public Key to CentOS

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

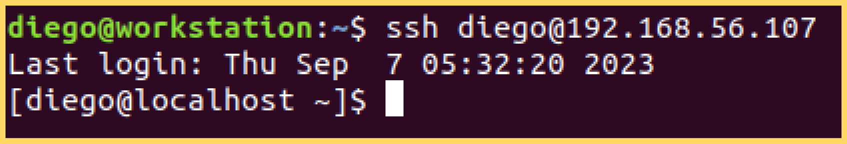


```
diego@workstation:~$ systemctl status sshd
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: ena
   Active: active (running) since Thu 2023-09-07 17:10:58 PST; 35min ago
  Process: 1941 ExecReload=/bin/kill -HUP $MAINPID (code=exited, status=0/SUCCE
  Process: 1937 ExecReload=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
  Process: 854 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
 Main PID: 866 (sshd)
    Tasks: 1 (limit: 4656)
   CGroup: /system.slice/ssh.service
           └─866 /usr/sbin/sshd -D

Sep 07 17:31:04 workstation systemd[1]: Reloading OpenBSD Secure Shell server.
Sep 07 17:31:04 workstation systemd[1]: Reloaded OpenBSD Secure Shell server.
Sep 07 17:31:04 workstation sshd[866]: Received SIGHUP; restarting.
Sep 07 17:31:04 workstation sshd[866]: Server listening on 0.0.0.0 port 22.
Sep 07 17:31:04 workstation sshd[866]: Server listening on :: port 22.
Sep 07 17:31:09 workstation systemd[1]: Reloading OpenBSD Secure Shell server.
Sep 07 17:31:09 workstation systemd[1]: Reloaded OpenBSD Secure Shell server.
Sep 07 17:31:09 workstation sshd[866]: Received SIGHUP; restarting.
Sep 07 17:31:09 workstation sshd[866]: Server listening on 0.0.0.0 port 22.
Sep 07 17:31:09 workstation sshd[866]: Server listening on :: port 22.
lines 1-21/21 (END)
```

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

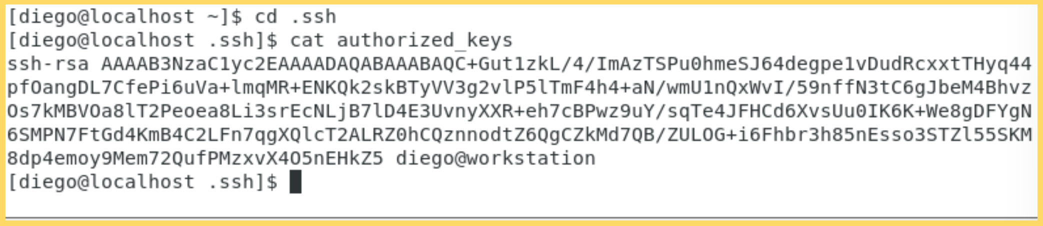


```
diego@workstation:~$ ssh-copy-id diego@192.168.56.107
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/diego/.ssh
/id_rsa.pub"
The authenticity of host '192.168.56.107 (192.168.56.107)' can't be established
.
ECDSA key fingerprint is SHA256:q010j3c1AX2d96YvMeVsn/mCzol73fdqc9QN/6IdTnk.
Are you sure you want to continue connecting (yes/no)? yes
```



```
diego@workstation:~$ ssh diego@192.168.56.107
Last login: Thu Sep  7 05:32:20 2023
[diego@localhost ~]$ 
```

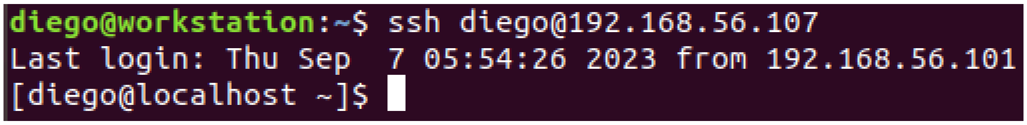
3. On CentOS, verify that you have the *authorized_keys*.



```
[diego@localhost ~]$ cd .ssh
[diego@localhost .ssh]$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC+Gut1zkL/4/ImAzTSPu0hmeSJ64degpe1vDudRcxxtTHyq44
pfOangDL7CfePi6uVa+lmqMR+ENKQk2skBTyVV3g2vlP5lTmF4h4+aN/wmU1nQxWvI/59nffN3tC6gJbeM4Bhvz
Os7kMBVOa8lT2Peoea8Li3srEcNLjB7lD4E3UvnyXXR+eh7cBPwz9uY/sqTe4JFHCd6XvsUu0IK6K+We8gDFYgN
6SMPN7FtGd4KmB4C2LFn7qgXQlcT2ALRZ0hCQznnodtZ6QgCZkMd7QB/ZULOG+i6Fhbr3h85nEsso3STZl55SKM
8dp4emoy9Mem72QufPMzxvX405nEHkZ5 diego@workstation
[diego@localhost .ssh]$ 
```

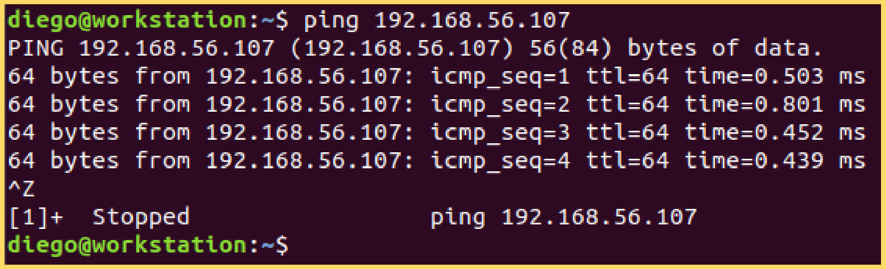
Task 4: Verify ssh remote connection

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

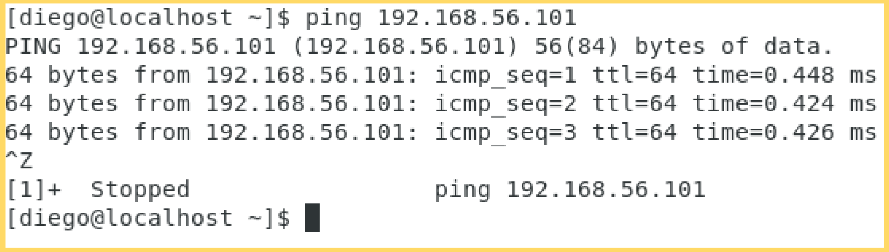


```
diego@workstation:~$ ssh diego@192.168.56.107
Last login: Thu Sep  7 05:54:26 2023 from 192.168.56.101
[diego@localhost ~]$ 
```

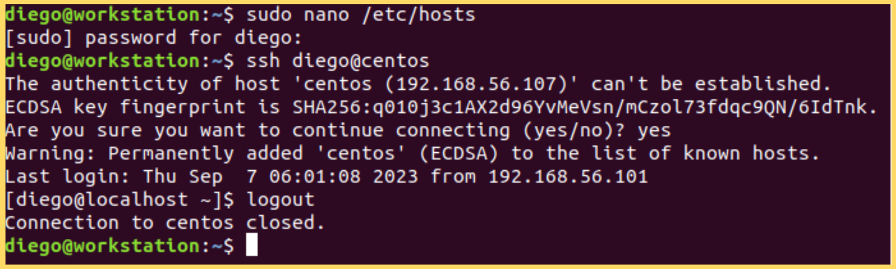
2. Show evidence that you are connected.



```
diego@workstation:~$ ping 192.168.56.107
PING 192.168.56.107 (192.168.56.107) 56(84) bytes of data.
64 bytes from 192.168.56.107: icmp_seq=1 ttl=64 time=0.503 ms
64 bytes from 192.168.56.107: icmp_seq=2 ttl=64 time=0.801 ms
64 bytes from 192.168.56.107: icmp_seq=3 ttl=64 time=0.452 ms
64 bytes from 192.168.56.107: icmp_seq=4 ttl=64 time=0.439 ms
^Z
[1]+  Stopped                 ping 192.168.56.107
diego@workstation:~$
```



```
[diego@localhost ~]$ ping 192.168.56.101
PING 192.168.56.101 (192.168.56.101) 56(84) bytes of data.
64 bytes from 192.168.56.101: icmp_seq=1 ttl=64 time=0.448 ms
64 bytes from 192.168.56.101: icmp_seq=2 ttl=64 time=0.424 ms
64 bytes from 192.168.56.101: icmp_seq=3 ttl=64 time=0.426 ms
^Z
[1]+  Stopped                 ping 192.168.56.101
[diego@localhost ~]$
```



```
diego@workstation:~$ sudo nano /etc/hosts
[sudo] password for diego:
diego@workstation:~$ ssh diego@centos
The authenticity of host 'centos (192.168.56.107)' can't be established.
ECDSA key fingerprint is SHA256:q010j3c1AX2d96YvMeVsn/mCzol73fdqc9QN/6IdTnk.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'centos' (ECDSA) to the list of known hosts.
Last login: Thu Sep  7 06:01:08 2023 from 192.168.56.101
[diego@localhost ~]$ logout
Connection to centos closed.
diego@workstation:~$
```

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?
   -To choose between Debian and Red Hat Linux distributions, you should consider your needs and preferences. Debian is a community-driven distribution that prioritizes user autonomy, open-source software, and personalization. It supports any architecture or platform that runs Linux, FreeBSD kernel, and any GNU tool sets such as GCC. On the other hand, Red Hat Enterprise Linux or RHEL is the most popular commercially supported Linux distribution

2. What are the main differences between Debian and Red Hat Linux distributions?
   -The main differences between Debian and Red Hat Linux distributions are their package management systems and tools. Debian uses the Debian (.deb) package format, while Red-hat uses the RPM (.rpm) package format. Arch and Slackware use their own packaging formats and tools.

Conclusion:
In this activity we have learned how to install CentOS and connect it to the workstation in ubuntu by the command (ssh-copy-id). The installation of an SSH server on CentOS is a simple and straightforward process that can greatly enhance the security and accessibility of a system. By successfully setting up SSH, users can securely connect to their CentOS server remotely, transfer files, and manage their system from anywhere in the world. It is crucial to ensure that proper security measures are implemented, such as strong passwords and disabling root login, to prevent unauthorized access. Overall, installing an SSH server on CentOS is a crucial step in ensuring a secure and efficient remote server management experience.