Name: Cruz, Patrick Danielle C.	**Date Performed: Aug 21, 2025**
Course/Section: CPE212 - CPE31S4	**Date Submitted: Aug 21, 2025**
Instructor: Engr. Robin Valenzuela	**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.

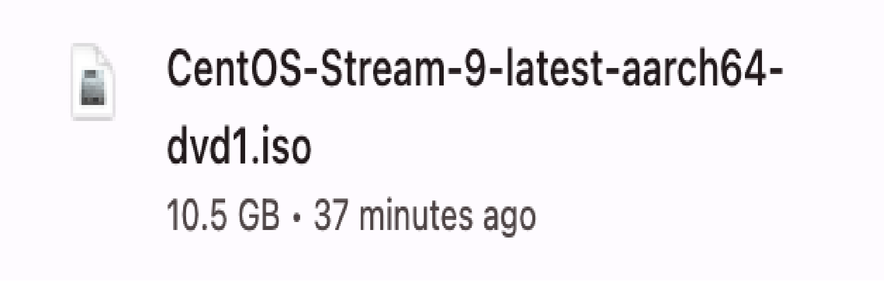
CentOS uses the RPM package format and YUM/DNF as the package manager.

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.

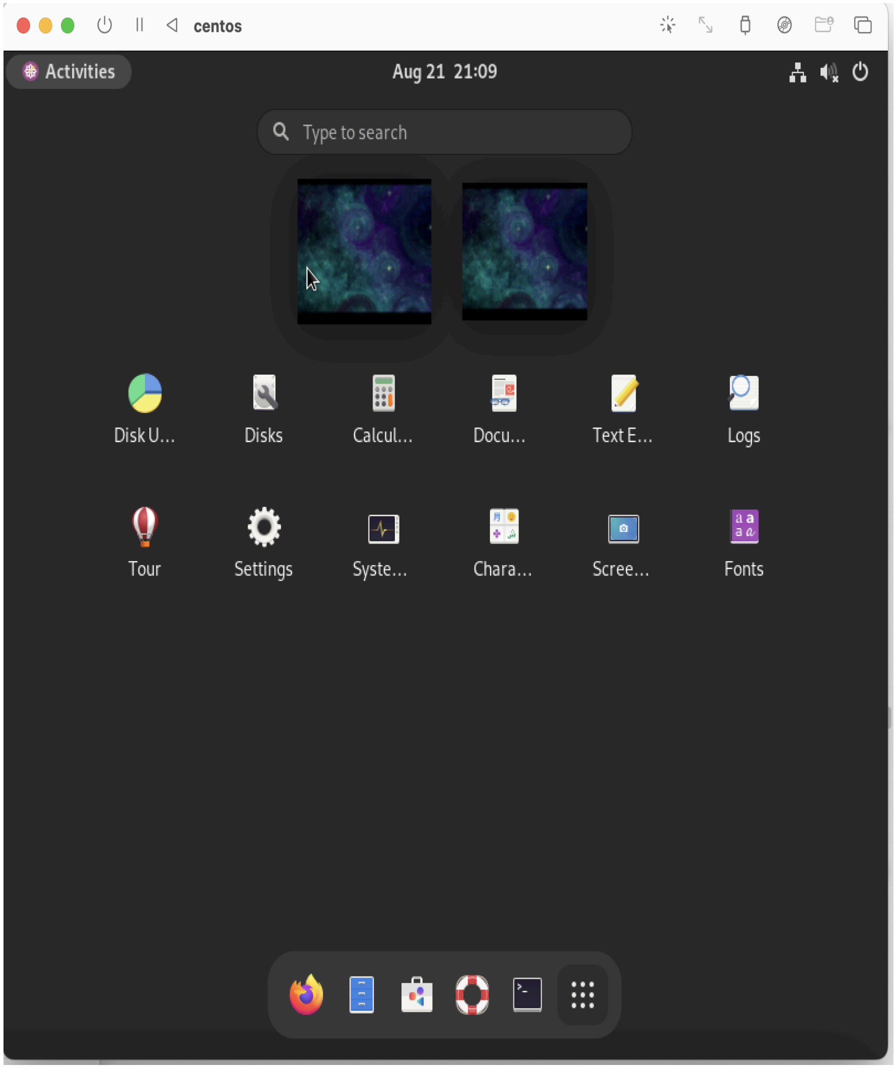
Task 1: Download the CentOS or RHEL-8 image (Create screenshots of the following)

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



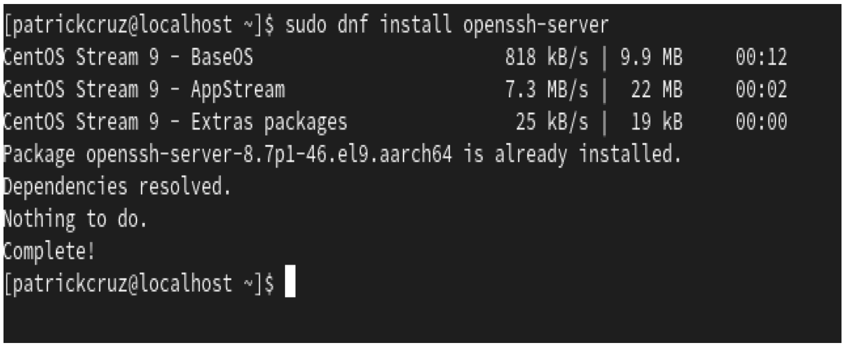
CentOS-Stream-9-latest-aarch64-dvd1.iso

10.5 GB • 37 minutes ago


centos
Activities
Aug 21 21:09
Type to search
Disk U...
Disks
Calcul...
Docu...
Text E...
Logs
Tour
Settings
Syste...
Chara...
Scree...
Fonts

Task 2: Install the SSH server package *openssh*

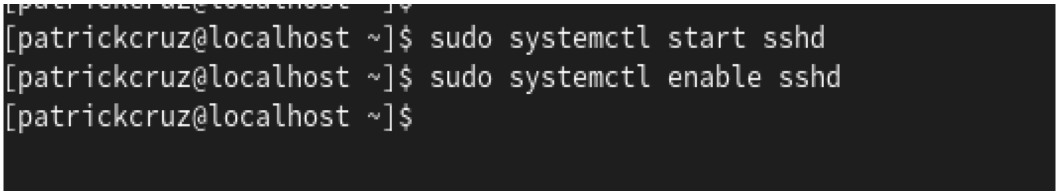
1. Install the ssh server package *openssh* by using the *dnf* command:
   $ dnf install openssh-server



```
[patrickcruz@localhost ~]$ sudo dnf install openssh-server
CentOS Stream 9 - BaseOS                      818 kB/s | 9.9 MB     00:12
CentOS Stream 9 - AppStream                   7.3 MB/s |  22 MB     00:02
CentOS Stream 9 - Extras packages              25 kB/s |  19 kB     00:00
Package openssh-server-8.7p1-46.el9.aarch64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[patrickcruz@localhost ~]$
```

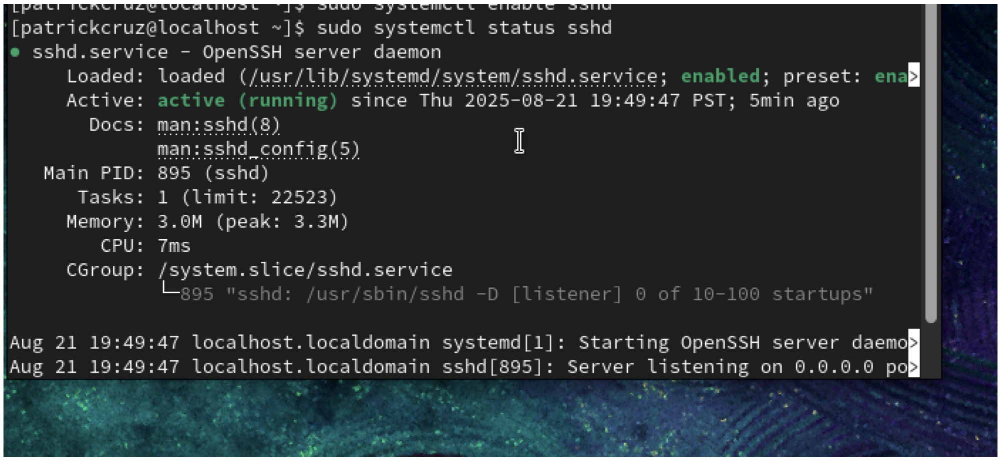
2. Start the *sshd* daemon and set to start after reboot:
   $ systemctl start sshd

   $ systemctl enable sshd



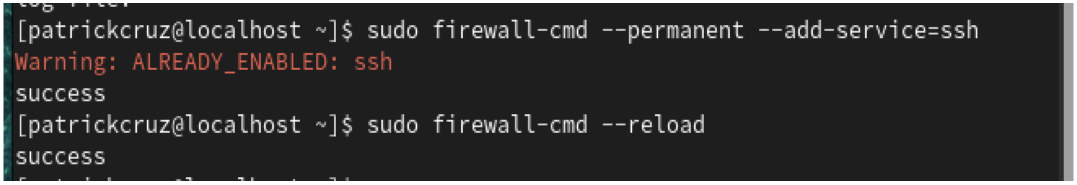
```
[patrickcruz@localhost ~]$
[patrickcruz@localhost ~]$ sudo systemctl start sshd
[patrickcruz@localhost ~]$ sudo systemctl enable sshd
[patrickcruz@localhost ~]$
```

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



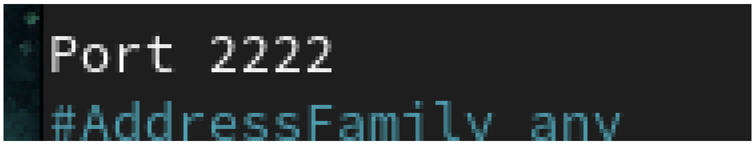
```
[patrickcruz@localhost ~]$ sudo systemctl enable sshd
[patrickcruz@localhost ~]$ sudo systemctl status sshd
● sshd.service - OpenSSH server daemon
     Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: ena>
     Active: active (running) since Thu 2025-08-21 19:49:47 PST; 5min ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 895 (sshd)
      Tasks: 1 (limit: 22523)
     Memory: 3.0M (peak: 3.3M)
        CPU: 7ms
     CGroup: /system.slice/sshd.service
             └─895 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Aug 21 19:49:47 localhost.localdomain systemd[1]: Starting OpenSSH server daemo>
Aug 21 19:49:47 localhost.localdomain sshd[895]: Server listening on 0.0.0.0 po>
```

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

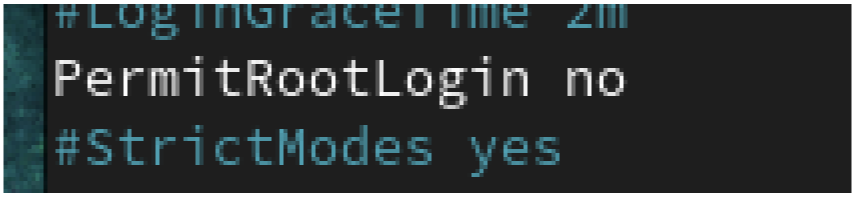


```
[patrickcruz@localhost ~]$ sudo firewall-cmd --permanent --add-service=ssh
Warning: ALREADY_ENABLED: ssh
success
[patrickcruz@localhost ~]$ sudo firewall-cmd --reload
success
```

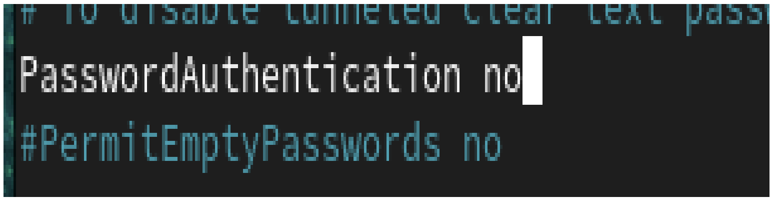
5. Locate the ssh server man config file /*etc/ssh/sshd_config* and perform custom configuration. Every time you make any change to the /*etc/ssh/sshd-config* configuration file reload the *sshd* service to apply changes:
   $ systemctl reload sshd



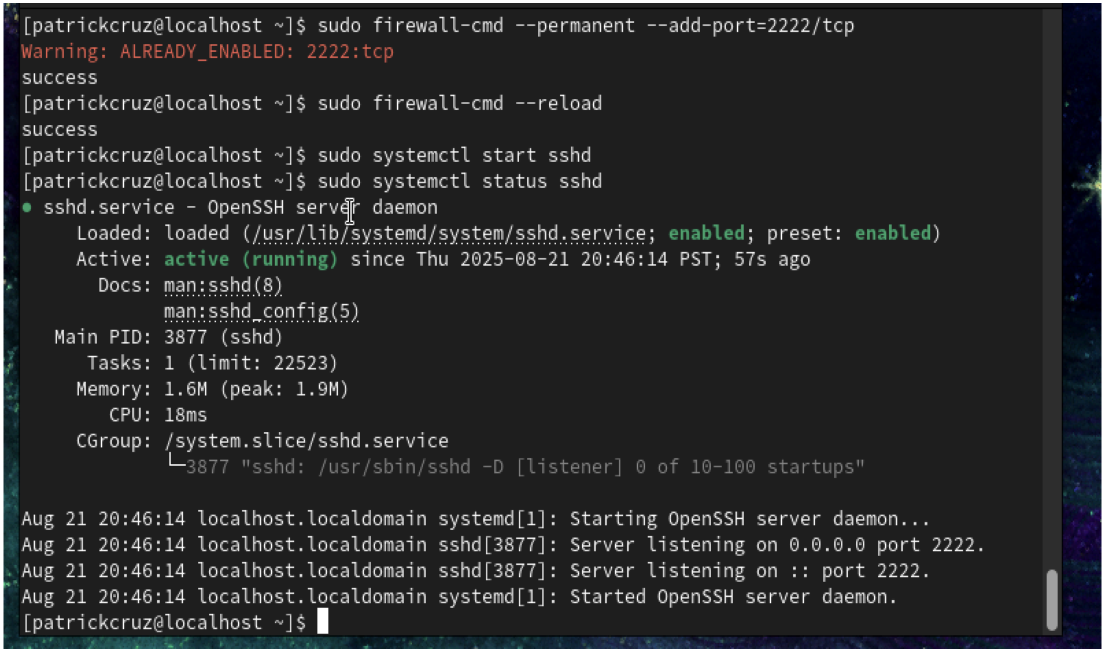
```
Port 2222
#AddressFamily any
```



```
#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
```



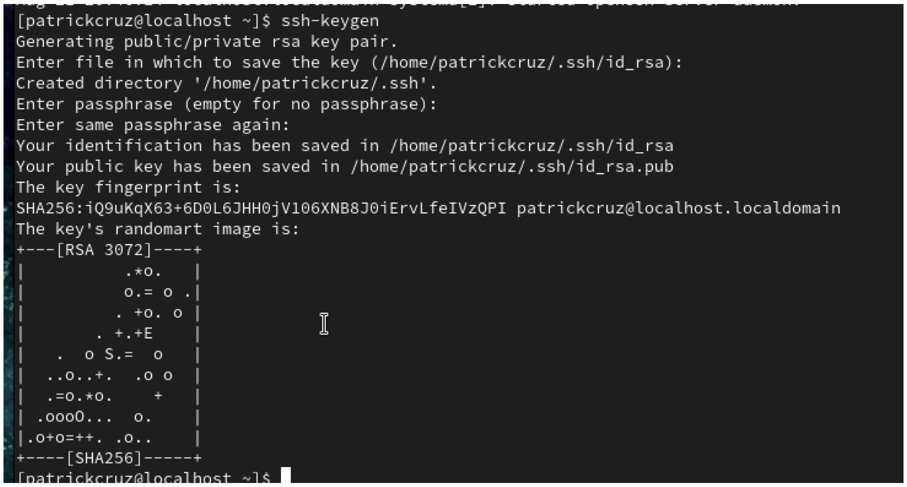
```
# To disable tunneled clear text passw
PasswordAuthentication no
#PermitEmptyPasswords no
```



```
[patrickcruz@localhost ~]$ sudo firewall-cmd --permanent --add-port=2222/tcp
Warning: ALREADY_ENABLED: 2222:tcp
success
[patrickcruz@localhost ~]$ sudo firewall-cmd --reload
success
[patrickcruz@localhost ~]$ sudo systemctl start sshd
[patrickcruz@localhost ~]$ sudo systemctl status sshd
● sshd.service - OpenSSH server daemon
     Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: enabled)
     Active: active (running) since Thu 2025-08-21 20:46:14 PST; 57s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 3877 (sshd)
      Tasks: 1 (limit: 22523)
     Memory: 1.6M (peak: 1.9M)
        CPU: 18ms
     CGroup: /system.slice/sshd.service
             └─3877 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Aug 21 20:46:14 localhost.localdomain systemd[1]: Starting OpenSSH server daemon...
Aug 21 20:46:14 localhost.localdomain sshd[3877]: Server listening on 0.0.0.0 port 2222.
Aug 21 20:46:14 localhost.localdomain sshd[3877]: Server listening on :: port 2222.
Aug 21 20:46:14 localhost.localdomain systemd[1]: Started OpenSSH server daemon.
[patrickcruz@localhost ~]$
```

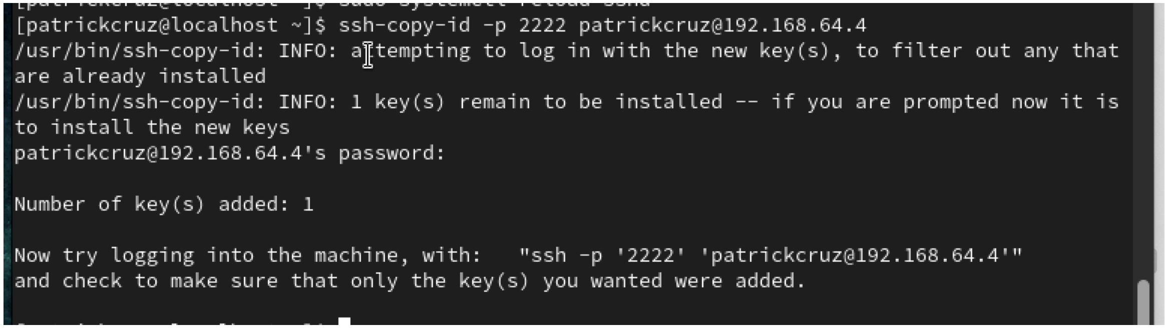
Task 3: Copy the Public Key to CentOS

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



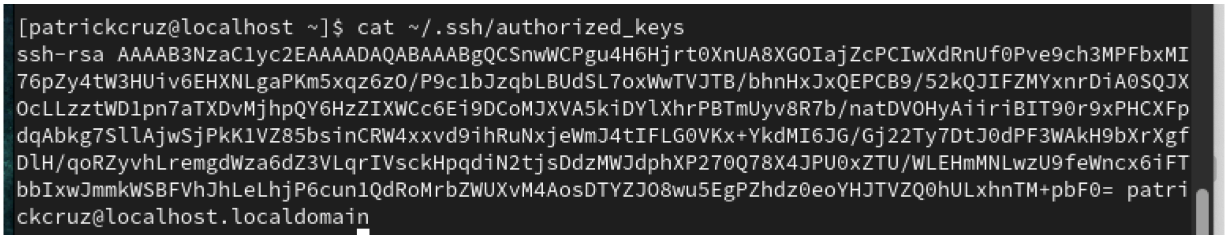
```
[patrickcruz@localhost ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/patrickcruz/.ssh/id_rsa):
Created directory '/home/patrickcruz/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/patrickcruz/.ssh/id_rsa
Your public key has been saved in /home/patrickcruz/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:iQ9uKqX63+6D0L6JHH0jV106XNB8J0iErvLfeIVzQPI patrickcruz@localhost.localdomain
The key's randomart image is:
+---[RSA 3072]----+
|          .*o.   |
|          o.= o .|
|         . +o. o |
|       . +.+E    |
|   .  o S.=  o   |
|  ..o..+.  .o o  |
|  .=o.*o.    +   |
| .ooO...  o.     |
|.o+o=++. .o..    |
+----[SHA256]-----+
[patrickcruz@localhost ~]$
```

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



```
[patrickcruz@localhost ~]$ ssh-copy-id -p 2222 patrickcruz@192.168.64.4
/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
patrickcruz@192.168.64.4's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh -p '2222' 'patrickcruz@192.168.64.4'"
and check to make sure that only the key(s) you wanted were added.
```

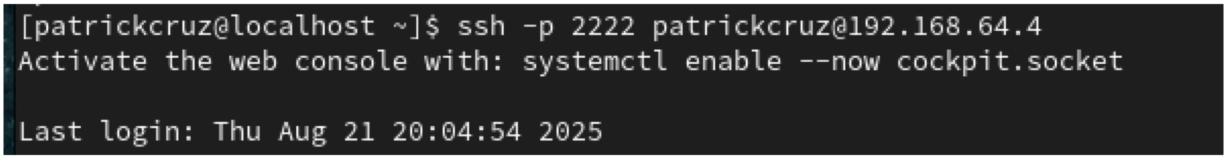
3. On CentOS, verify that you have the *authorized_keys*.



```
[patrickcruz@localhost ~]$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCSnwWCPgu4H6Hjrt0XnUA8XGOIajZcPCIwXdRnUf0Pve9ch3MPFbxMI
76pZy4tW3HUiv6EHXNLgaPKm5xqz6zO/P9c1bJzqbLBUdSL7oxWwTVJTB/bhnHxJxQEPCB9/52kQJIFZMYxnrDiA0SQJX
OcLLzztWD1pn7aTXDvMjhpQY6HzZIXWCc6Ei9DCoMJXVA5kiDYlXhrPBTmUyv8R7b/natDVOHyAiiriBIT90r9xPHCXFp
dqAbkg7SllAjwSjPkK1VZ85bsinCRW4xxvd9ihRuNxjeWmJ4tIFLG0VKx+YkdMI6JG/Gj22Ty7DtJ0dPF3WAkH9bXrXgf
DlH/qoRZyvhLremgdWza6dZ3VLqrIVsckHpqdiN2tjsDdzMWJdphXP270Q78X4JPU0xZTU/WLEHmMNLwzU9feWncx6iFT
bbIxwJmmkWSBFVhJhLeLhjP6cun1QdRoMrbZWUXvM4AosDTYZJO8wu5EgPZhdz0eoYHJTVZQ0hULxhnTM+pbF0= patri
ckcruz@localhost.localdomain
```

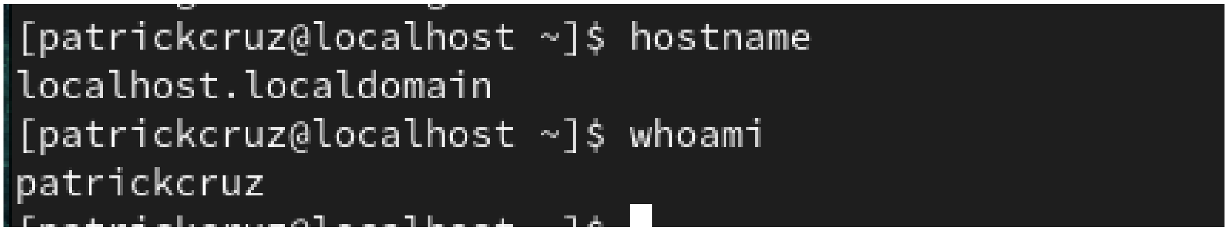
Task 4: Verify ssh remote connection

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



```
[patrickcruz@localhost ~]$ ssh -p 2222 patrickcruz@192.168.64.4
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Thu Aug 21 20:04:54 2025
```

2. Show evidence that you are connected.



```
[patrickcruz@localhost ~]$ hostname
localhost.localdomain
[patrickcruz@localhost ~]$ whoami
patrickcruz
```

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?
   - In choosing between Debian and Red Hat Linux distributions, we should consider factors such as stability, package management system, community

vs. enterprise support, security update policies, ease of administration, hardware compatibility, and long-term maintenance.

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

    - The main difference is that Debian is community-driven, primarily focusing on stability and free software with the APT (dpkg) package manager, while Red Hat (RHEL) is enterprise-focused, offering professional support, certifications, and uses the YUM/DNF (RPM) package manager. Debian is often preferred for flexibility and servers, while Red Hat is favored in enterprise environments requiring official support and certifications.