Contents

Introduction

Secure Shell (SSH) is a cryptographic protocol that allows a client to interact with a remote server in a secure environment.

High-level encryption protects the exchange of sensitive information and allows flie trans or issue commands on remote machines securely

Prerequisites

- CentOS 7 system to act as an SSH server
- A user with necessary permissions
- Access to a command line (Ctrl-Alt-T)
- **yum** utility (included by default)

Contents

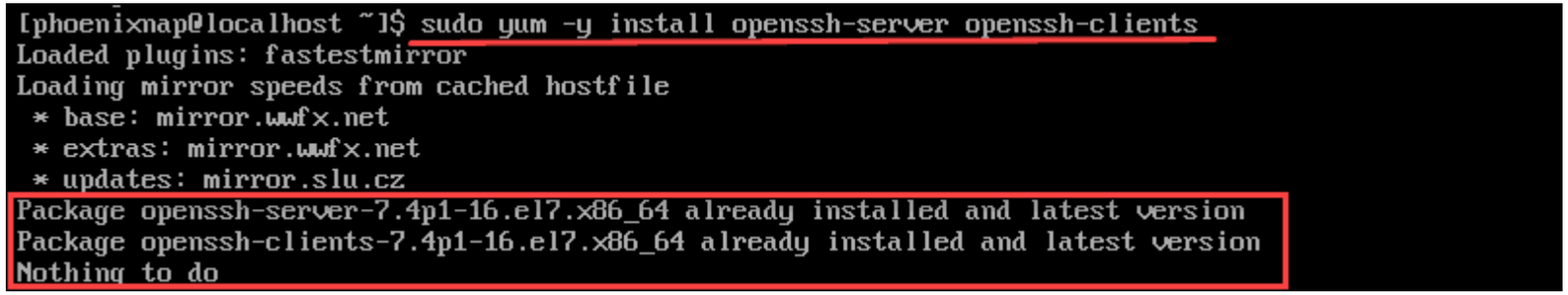
SSH software packages are included on CentOS by default. However, if these packages are not present on your system, easily install them by completing Step 1, outlined below.

Step 1: Install OpenSSH Server Software Package

Enter the following command from your terminal to start the installation process:

```
sudo yum -y install openssh-server openssh-clients
```

This command installs both the OpenSSH client applications, as well as the OpenSSH server daemon, **sshd**.



In this example, the system informs us that the latest version is already present.

Step 2: Starting SSH Service

Contents

```
sudo systemctl start sshd
```

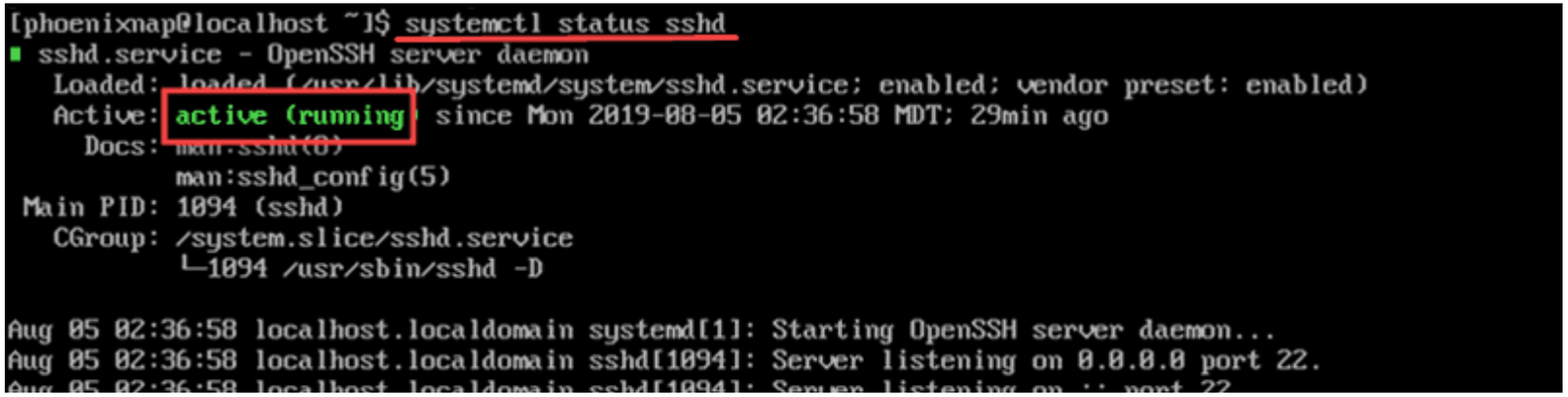
When active, **sshd** continuously listens for client connections from any of the client tools. When a connection request occurs, **sshd** sets up the correct connection.

Step 3: Check sshd status

Check the status of the SSH daemon:

```
sudo systemctl status sshd
```

As we have previously started the service, the output confirms that it is active.

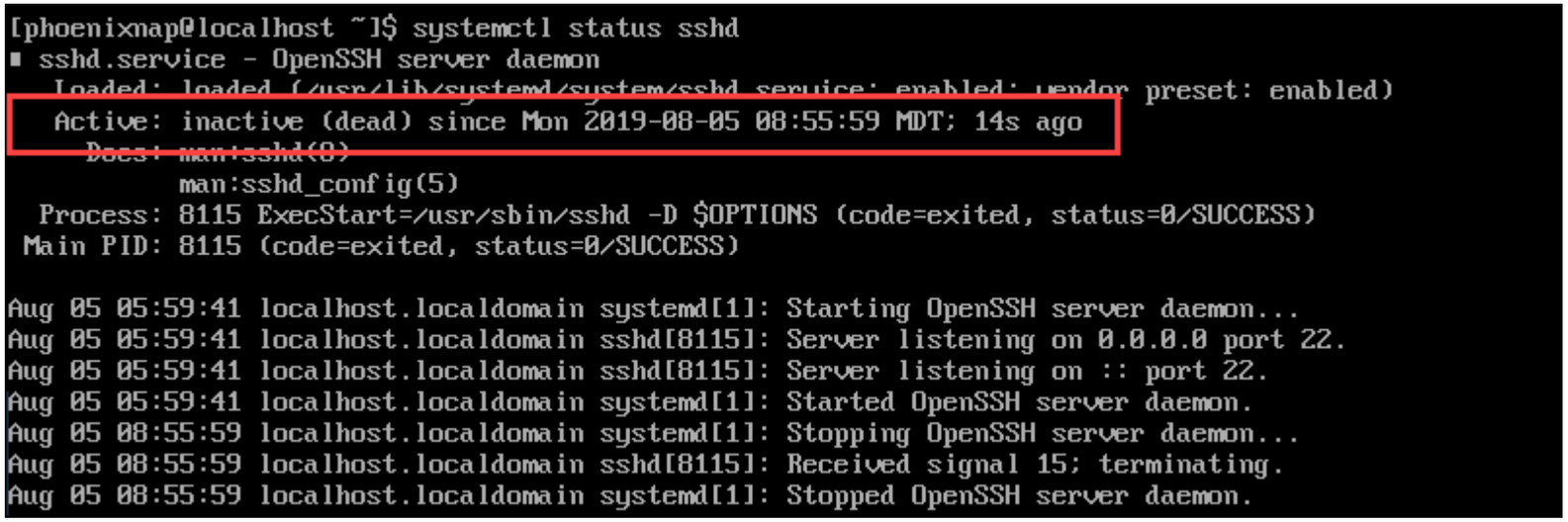


Contents

To stop the SSH daemon enter:

```
systemctl stop sshd
```

We can check if the service has stopped by verifying the status. The output shows that the service is inactive and the time and date when the status last changed.



Step 4: Enable OpenSSH Service

Enable SSH to start automatically after each system reboot by using the **systemctl** command:

Contents

To disable SSH after reboot enter:

```
sudo systemctl disable sshd
```

OpenSSH Server Configuration

Properly configuring the **sshd** configuration file hardens server security. The most common settings to enhance security are changing the port number, disabling root logins, and limiting access to only certain users.

To edit these settings access the **/etc/ssh/sshd_config** file:

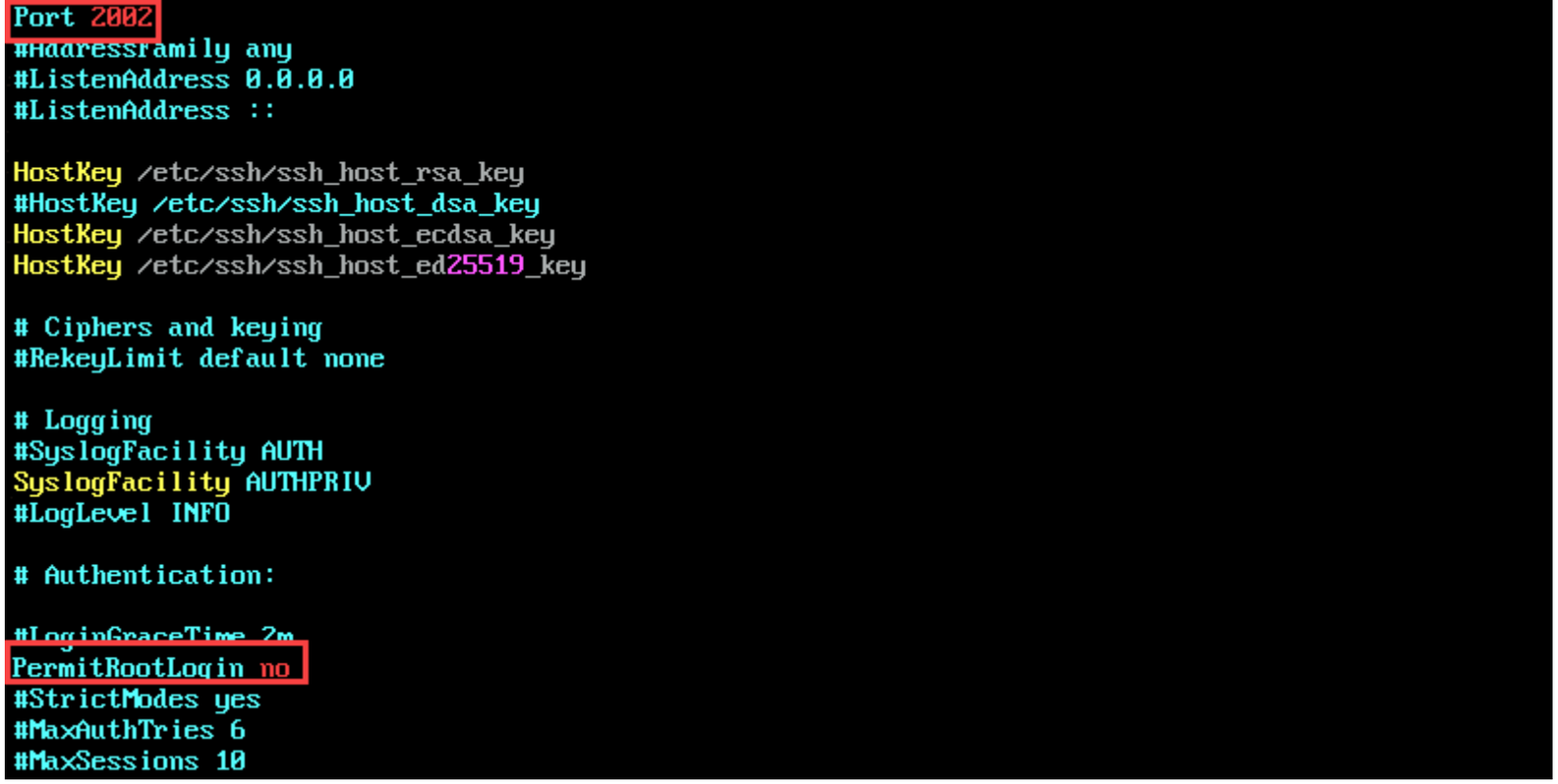
```
sudo vim /etc/ssh/sshd_config
```

Once you access the file by using a text editor (in this example we used **vim**), you can disable root logins and edit the default port number:

- To disable root login:

PermitRootLogin no

Contents



Remember to uncomment the lines that you edit by removing the hashtag.

Save and close the file. Restart **sshd**:

```
service sshd restart
```

Contents

After successfully enabling SSH and configuring the **sshd** file, adjust the firewall settings to make sure there are no compatibility issues.

It is also possible to restrict IP access to make the connection even more secure.

To restrict IP access, edit the **iptables** file by typing:

```
sudo vim /etc/sysconfig/iptables
```

To allow access using the port defined in the sshd config file, add the following line to the iptables file:

```
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 2002 -j ACCEPT
```

To restrict access to a specific IP, for example 133.123.40.166, edit the line as follows:

```
-A RH-Firewall-1-INPUT -s 133.123.40.166 -m state --state NEW -p tcp --dport 2002 -j ACC
EPT
```

Contents

```
# sample configuration for iptables service
# you can edit this manually or use system-config-firewall
# please do not ask us to add additional ports/services to this default configuration
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m tcp --dport 2002 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibitedi
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
-A RH-Firewall-1-INPUT -s 133.123.40.166 -m state --state NEW -m tcp -p tcp --dport 2002 -j ACCEPT
COMMIT
~
```

If your site uses IPv6, and you are editing ip6tables, use the line:

```
-A RH-Firewall-1-INPUT -m tcp -p tcp --dport 2002 -j ACCEPT
```

Save and exit the file by pressing Escape (Esc) on your keyboard and typing:

:X

Press Enter to confirm.

Restart iptables to apply the changes: