

# Remote Terminal

SSH – or Secure Shell – allows us to connect to our server over our local network to run commands remotely.

This is very helpful for running an always-on server that doesn't have a display attached, often called a "headless" system. During the Debian install process, we opted to enable SSH by support.

When connecting to a new host, SSH will create a digital fingerprint for it. When we connect to this server in the future, it will compare the server's fingerprint to the one currently on file. In the event they don't match in the future, SSH will warn us that there is a mismatch and someone might be impersonating your server.

There are many different ways to connect to your server using SSH. We will highlight some common methods for connecting to our server and controlling it remotely.

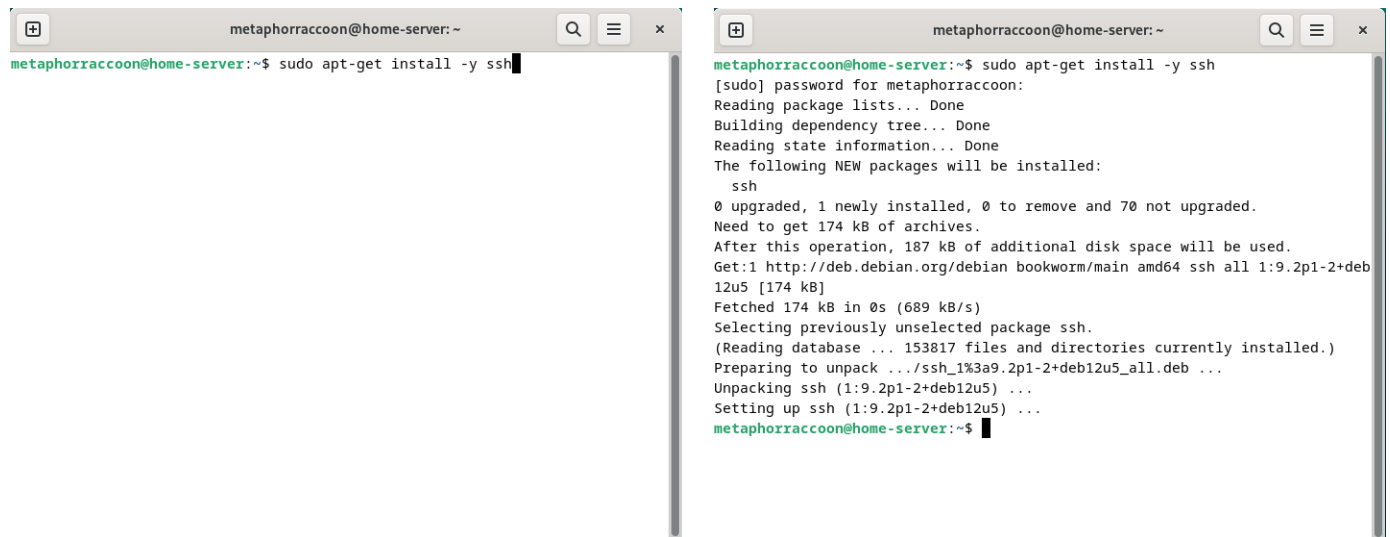
## Linux

You can use the terminal available on most Distributions of Linux – such as Debian, Ubuntu, and Linux Mint – to remotely connect to your server.

You will need to know your [username set during the installation process](#) and the IP Address you set during [network configuration](#).

You can open a terminal and ensure that SSH is installed by running the command:

```
sudo apt-get install -y ssh
```



The image shows two terminal windows from the user 'metaphorraccoon' on a machine named 'home-server'. The left window shows the command 'sudo apt-get install -y ssh' being entered. The right window shows the output of this command, which includes the password prompt, package list reading, dependency tree building, and the installation of the 'ssh' package. The output indicates that 174 kB of archives are needed and that the package is being fetched from the Debian repository.

```
metaphorraccoon@home-server: ~  
metaphorraccoon@home-server:~$ sudo apt-get install -y ssh  
  
metaphorraccoon@home-server:~$ sudo apt-get install -y ssh  
[sudo] password for metaphorraccoon:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following NEW packages will be installed:  
  ssh  
0 upgraded, 1 newly installed, 0 to remove and 70 not upgraded.  
Need to get 174 kB of archives.  
After this operation, 187 kB of additional disk space will be used.  
Get:1 http://deb.debian.org/debian bookworm/main amd64 ssh all 1:9.2p1-2+deb12u5 [174 kB]  
Fetched 174 kB in 0s (689 kB/s)  
Selecting previously unselected package ssh.  
(Reading database ... 153817 files and directories currently installed.)  
Preparing to unpack .../ssh_1%3a9.2p1-2+deb12u5_all.deb ...  
Unpacking ssh (1:9.2p1-2+deb12u5) ...  
Setting up ssh (1:9.2p1-2+deb12u5) ...  
metaphorraccoon@home-server:~$
```

Once we are sure this is installed, we can run connect to our server through the terminal using the `ssh` command. We will be connecting to our user account what is at our server's IP address:

```
ssh username@192.168.68.100
```



The image shows a terminal window from the user 'metaphorraccoon' on a machine named 'home-server'. The command 'ssh metaphorraccoon@192.168.68.100' is entered at the prompt.

```
metaphorraccoon@home-server: ~  
metaphorraccoon@home-server:~$ ssh metaphorraccoon@192.168.68.100
```

You will be alerted that the authenticity of the host server could not be verified. This happens because we have never connected to the server before and it can't ensure the server's fingerprint.

```
metaphorraccoon@home-server: ~  
metaphorraccoon@home-server:~$ ssh metaphorraccoon@192.168.68.100  
The authenticity of host '192.168.68.100 (192.168.68.100)' can't be established.  
ED25519 key fingerprint is SHA256:kWh1o1ifp1FwFvNHiWW/FueuUk/2BZEPooUHBqLd0g.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

We can type 'yes' then hit 'enter' to accept the newly generated host fingerprint and remember it for future use.

```
metaphorraccoon@home-server: ~  
metaphorraccoon@home-server:~$ ssh metaphorraccoon@192.168.68.100  
The authenticity of host '192.168.68.100 (192.168.68.100)' can't be established.  
ED25519 key fingerprint is SHA256:kWh1o1ifp1FwFvNHiWW/FueuUk/2BZEPooUHBqLd0g.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '192.168.68.100' (ED25519) to the list of known hosts.  
metaphorraccoon@192.168.68.100's password: █
```

Now, you will be prompted for your password. After hitting 'enter', you will be connected to your server and can run commands.

```
metaphorraccoon@development: ~  
metaphorraccoon@home-server:~$ ssh metaphorraccoon@192.168.68.100  
The authenticity of host '192.168.68.100 (192.168.68.100)' can't be established.  
ED25519 key fingerprint is SHA256:kWh1o1ifplFwFvNHiWW/FueuUk/2BZEPooUHBQLd0g.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '192.168.68.100' (ED25519) to the list of known hosts.  
metaphorraccoon@192.168.68.100's password:  
Linux development 6.1.0-32-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.129-1 (2025-03-06) x86_64  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Sun Mar 30 13:27:17 2025 from ::ffff:127.0.0.1  
metaphorraccoon@development:~$ █
```

You can disconnect from the SSH connection with the following command:

```
exit
```

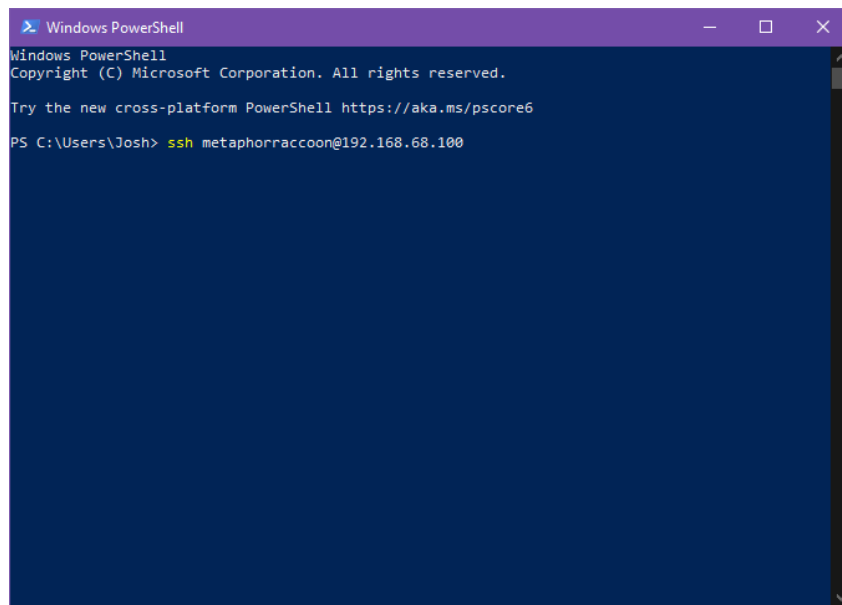
# Windows

Windows 10 and Windows 11 come with a client pre-installed so you can use SSH to connect to your server remotely.

You will need to know your [username set during the installation process](#) and the IP Address you set during [network configuration](#).

You can open a terminal or PowerShell and run the following command:

```
ssh username@192.168.68.100
```

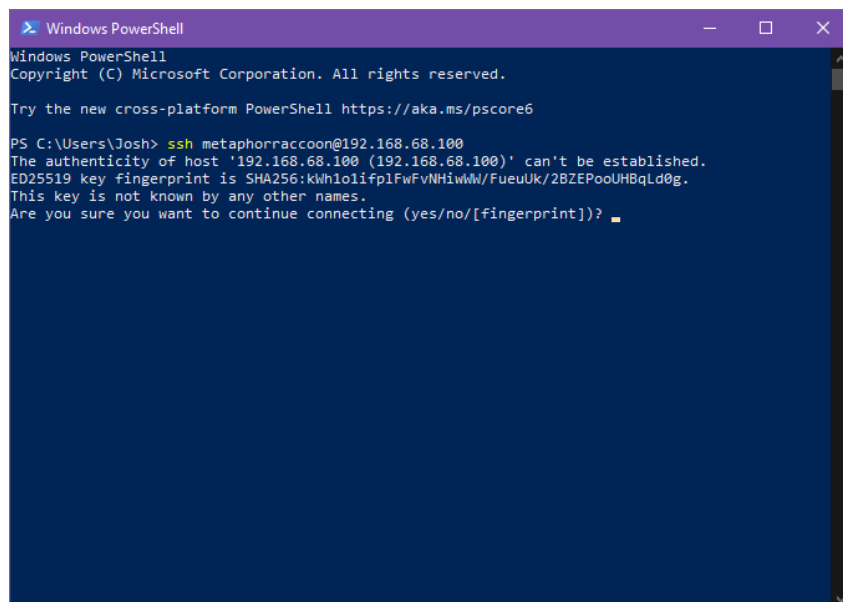


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Josh> ssh metaphorraccoon@192.168.68.100
```

You will be alerted that the authenticity of the host server could not be verified. This happens because we have never connected to the server before and it can't ensure the server's fingerprint.



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Josh> ssh metaphorraccoon@192.168.68.100
The authenticity of host '192.168.68.100 (192.168.68.100)' can't be established.
ED25519 key fingerprint is SHA256:KWh1o1fplFwFvNHiwWn/FueuUk/2BZEPOoUHBqld0g.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? 
```

We can type 'yes' then hit 'enter' to accept the newly generated host fingerprint and remember it for future use.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Josh> ssh metaphorraccoon@192.168.68.100
The authenticity of host '192.168.68.100 (192.168.68.100)' can't be established.
ED25519 key fingerprint is SHA256:kWh1o1fp1FwFvNH1wM/FueuUk/2BZEPOoUHBqLd0g.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.68.100' (ED25519) to the list of known hosts.
metaphorraccoon@192.168.68.100's password: █
```

Now, you will be prompted for your password. After hitting 'enter', you will be connected to your server and can run commands.

```
metaphorraccoon@development: ~
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Josh> ssh metaphorraccoon@192.168.68.100
The authenticity of host '192.168.68.100 (192.168.68.100)' can't be established.
ED25519 key fingerprint is SHA256:kWh1o1fp1FwFvNH1wM/FueuUk/2BZEPOoUHBqLd0g.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.68.100' (ED25519) to the list of known hosts.
metaphorraccoon@192.168.68.100's password: █
Linux development 6.1.0-32-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.129-1 (2025-03-06) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Apr 10 16:10:59 2025 from 192.168.68.67
metaphorraccoon@development: $ █
```

You can disconnect from the SSH connection with the following command:

```
exit
```

# MacOS

MacOS, and the legacy OS X, come with SSH already installed so you can remotely connect to your server.

You will need to know your [username set during the installation process](#) and the IP Address you set during [network configuration](#).

We can connect to our server through the terminal using the `ssh` command. We will be connecting to our user account what is at our server's IP address:

```
ssh username@192.168.68.100
```

You will be alerted that the authenticity of the host server could not be verified. This happens because we have never connected to the server before and it can't ensure the server's fingerprint.

We can type 'yes' then hit 'enter' to accept the newly generated host fingerprint and remember it for future use.

Now, you will be prompted for your password. After hitting 'enter', you will be connected to your server and can run commands.

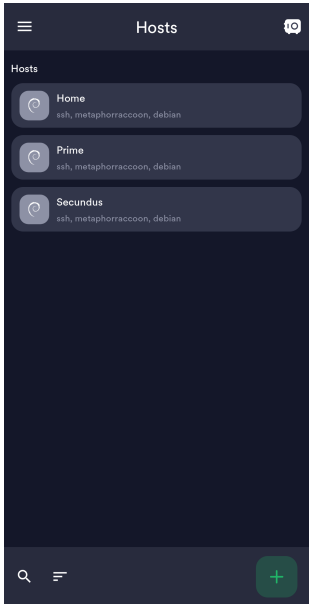
You can disconnect from the SSH connection with the following command:

```
exit
```

## Android & iOS

There are apps available through the Google Play Store and Apple App Store that allow you to remotely connect to your server using SSH.

We recommend [Termius](#), a freeware application for Android 7 and iOS 16 or newer. You can create a profile for your server with the credentials pre-saved for quick connection. As a premium paid feature, you can also sync these hosts between the mobile and desktop clients.



# Web Access

SSH is a great option for connecting to your server over your local network, but is not available through the world wide web. The easiest way to accomplish this is to use our home server to host Guacamole, a web application that facilitates remote connection to RDP, VNC and SSH through your web browser.

