

# How to mount remote directory in Linux with SSHFS

From an end-user perspective, SSHFS is a simple and very easy to use solution. SSHFS is also extremely secure because it depends on the proven OpenSSH server.

On the Internet, there is definitely no shortage of file synchronization solutions between the two devices. Google Drive, Microsoft OneDrive, NextCloud, etc . may not give you the official Linux client. But perhaps you do not trust a large corporation in storing your important files. Or, maybe, you want a higher level of security than what NextCloud provides, as there are likely to be unexplored security vulnerabilities in existence.

## Why should I use SSHFS?

From an end-user perspective, SSHFS is a simple and very easy to use solution. SSHFS is also extremely secure because it depends on the proven OpenSSH server. SSHFS encryption is also very high, so you can rest assured that no one can steal files during the transfer process. An updated server is correctly configured, only listening to SSH connections, often impossible to crack, unless an attacker comes from NSA or a security specialist, etc. (but maybe they I also don't want to take your files for anything.

## Why don't you want to use SSHFS?

If you want the best transfer speed possible, you can use other solutions like NFS. SSHFS is not the slowest option, if your Internet connection is of high quality, it is not the fastest way. And if you want to move folders containing hundreds of small files, this task becomes really terrible. Also, if you want to tweak file sharing settings, based on individual users or other factors, you may want to use another software.

In short, if all you need is an easy, safe, and quick way to synchronize remote folders with a local folder, you'll be happy to use SSHFS.

## Install SSHFS

On Arch Linux-based installations, use the following command:

```
sudo pacman -S sshfs
```

If you use a Fedora-based distribution, use the following command:

```
sudo dnf install sshfs
```

On Debian and Ubuntu, use the command:

```
sudo apt install sshfs
```

For OpenSUSE users, enter the following command:

```
sudo zypper install sshfs
```

## Server configuration

If you rent a server or VPS, the OpenSSH server daemon has been configured. Follow the steps suggested by the cloud service provider to configure the regular user (not the root). Some providers allow you to do this directly from the web control panel and even allow you to enter a public key to access SSH. In this case, create local key pairs using the `ssh-keygen` command. Then enter the public key from **/home/your\_username/.ssh/id\_rsa.pub**.

If the cloud provider does not provide a tool to easily enter public SSH keys, do this manually. At the very least, do not allow root login and disable password login. Instead, use only SSH keys because they cannot be brute-force attacks, as is the case with passwords.

If you want to synchronize files between two home computers, just consider one as the server (install the **openssh-server** package and configure) and consider the other computer as a client, then apply the steps the same, similar.

## Mount a local remote directory with SSHFS

First, create a folder that will be synchronized with the remote device.

```
mkdir $HOME/sshfs
```

Next, mount the local remote directory via SSHFS. Replace the **user** with the actual user name created on the server and **203.0.113.1** by the actual IP address name of the remote device.

```
sshfs user@203.0.113.1:/home/user $HOME/sshfs
```

Of course, if you don't want to synchronize the entire user's home directory on the server side, simply replace **/home / user** with **/home / user / some\_other\_directory** after you create it on the server.

When you want to unmount, use this command:

```
cd && fusermount -u $HOME/sshfs
```

If you want a directory to synchronize permanently with the remote device, add a command like **sshfs user@203.0.113.1: /home / user \$HOME / sshfs** in the autostart manager. Each graphics manager has a different autostart configuration manager, so refer to the help guide for the desktop environment you are using. Some sources recommend adding **/etc / fstab**, but you should avoid that, because failing to mount the directory may prevent your system from booting completely.

Hope this article is useful for you. If necessary, you can refer to the following link:

<https://linux.die.net/man/1/sshfs>

Hope you are succesful.

You finished reading the article "**How to mount remote directory in Linux with SSHFS**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.

---