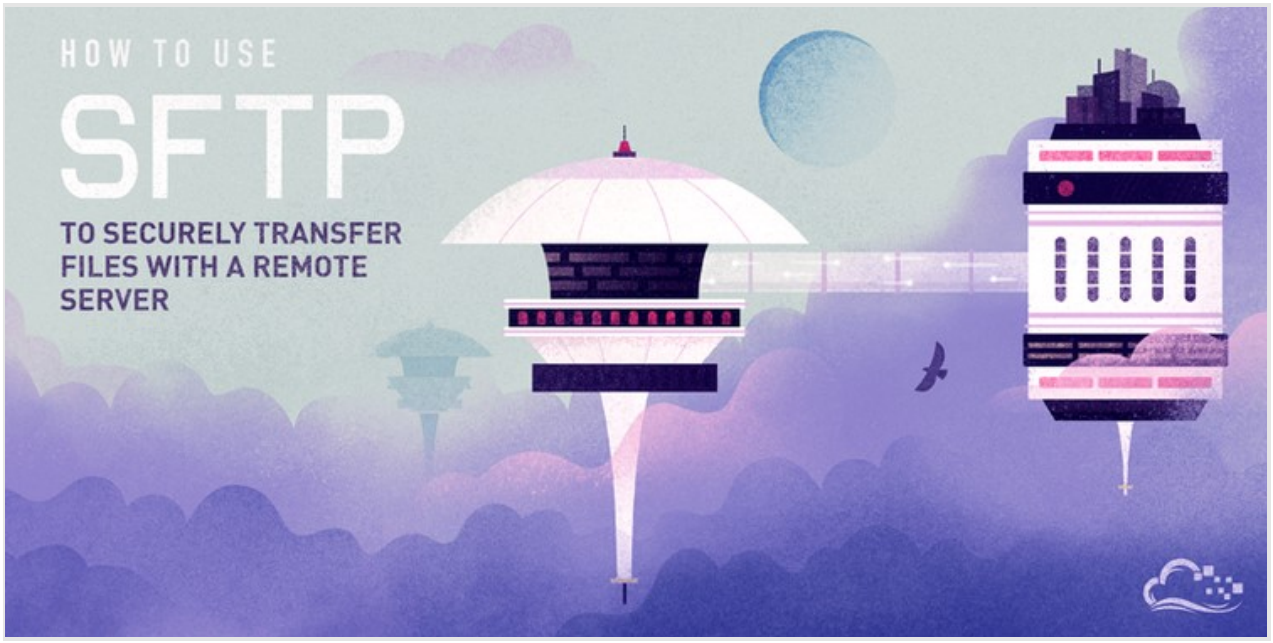
NEW Fresh features and 1-Click Apps for DigitalOcean Kubernetes ›



How To Use SFTP to Securely Transfer Files with a Remote Server

Posted August 13, 2013 2.8m LINUX BASICS

By Justin Ellingwood
Become an author

Introduction

FTP, or “File Transfer Protocol” is a popular method of transferring files between two remote systems.

SFTP, which stands for SSH File Transfer Protocol, or Secure File Transfer Protocol, is a separate protocol packaged with SSH that works in a similar way over a secure

SCROLL TO TOP

connection. The advantage is the ability to leverage a secure connection to transfer files and traverse the filesystem on both the local and remote system.

In almost all cases, SFTP is preferable to FTP because of its underlying security features and ability to piggy-back on an SSH connection. FTP is an insecure protocol that should only be used in limited cases or on networks you trust.

Although SFTP is integrated into many graphical tools, this guide will demonstrate how to use it through its interactive command line interface.

How to Connect with SFTP

By default, SFTP uses the SSH protocol to authenticate and establish a secure connection. Because of this, the same authentication methods are available that are present in SSH.

Although passwords are easy to use and set up by default, we recommend you create SSH keys and transfer your public key to any system that you need to access. This is much more secure and can save you time in the long run.

Please see this guide to set up SSH keys in order to access your server if you have not done so already.

If you can connect to the machine using SSH, then you have completed all of the necessary requirements necessary to use SFTP to manage files. Test SSH access with the following command:

```
ssh sammy@your_server_ip_or_remote_hostname
```

If that works, exit back out by typing:

```
exit
```

SCROLL TO TOP

We can establish an SSH connection and then open up an SFTP session using that connection by issuing the following command:

```
sftp sammy@your_server_ip_or_remote_hostname
```

You will connect the the remote system and your prompt will change to an SFTP prompt.

If you are working on a custom SSH port (not the default port 22), then you can open an SFTP session as follows:

```
sftp -oPort=custom_port sammy@your_server_ip_or_remote_hostname
```

This will connect you to the remote system by way of your specified port.

Getting Help in SFTP

The most useful command to learn first is the help command. This gives you access to a summary of the SFTP help. You can call it by typing either of these in the prompt:

```
help
```

```
?
```

This will display a list of the available commands:

```
Available commands:
bye                                Quit sftp
cd path                            Change remote directory to 'path'
chgrp grp path                     Change group of file 'path' to 'grp'
chmod mode path                    Change permissions of file 'path' to 'mode'
chown own path                     Change owner of file 'path' to 'own'
```

SCROLL TO TOP

```
df [-hi] [path]                    Display statistics for current directory or
                                   filesystem containing 'path'
exit                               Quit sftp
get [-Ppr] remote [local]          Download file
help                               Display this help text
lcd path                           Change local directory to 'path'
. . .
```

We will explore some of the commands you see in the following sections.

Navigating with SFTP

We can navigate through the remote system's file hierarchy using a number of commands that function similarly to their shell counterparts.

First, let's orient ourselves by finding out which directory we are in currently on the remote system. Just like in a typical shell session, we can type the following to get the current directory:

```
pwd
```

```
Remote working directory: /home/demouser
```

We can view the contents of the current directory of the remote system with another familiar command:

```
ls
```

```
Summary.txt     info.html       temp.txt        testDirectory
```

Note that the commands within the SFTP interface are not the normal shell commands and are not as feature-rich, but they do implement some of the more important optional flags:

SCROLL TO TOP

```
ls -la
```

```
drwxr-xr-x    5 demouser   demouser       4096 Aug 13 15:11 .
drwxr-xr-x    3 root     root         4096 Aug 13 15:02 ..
-rw-------    1 demouser   demouser          5 Aug 13 15:04 .bash_history
-rw-r--r--    1 demouser   demouser        220 Aug 13 15:02 .bash_logout
-rw-r--r--    1 demouser   demouser       3486 Aug 13 15:02 .bashrc
drwx------    2 demouser   demouser       4096 Aug 13 15:04 .cache
-rw-r--r--    1 demouser   demouser        675 Aug 13 15:02 .profile
. . .
```

To get to another directory, we can issue this command:

```
cd testDirectory
```

We can now traverse the remote file system, but what if we need to access our local file system? We can direct commands towards the local file system by preceding them with an “l” for local.

All of the commands discussed so far have local equivalents. We can print the local working directory:

```
lpwd
```

```
Local working directory: /Users/demouser
```

We can list the contents of the current directory on the local machine:

```
lls
```

```
Desktop         local.txt       test.html
Documents       analysis.rtf        zebra.html
```

SCROLL TO TOP

We can also change the directory we wish to interact with on the local system:

```
lcd Desktop
```

Transferring Files with SFTP

Navigating the remote and local filesystems is of limited usefulness without being able to transfer files between the two.

Transferring Remote Files to the Local System

If we would like download files from our remote host, we can do so by issuing the following command:

```
get remoteFile
```

```
Fetching /home/demouser/remoteFile to remoteFile
/home/demouser/remoteFile                       100%   37KB  36.8KB/s   00:01
```

As you can see, by default, the “get” command downloads a remote file to a file with the same name on the local file system.

We can copy the remote file to a different name by specifying the name afterwards:

```
get remoteFile localFile
```

The “get” command also takes some option flags. For instance, we can copy a directory and all of its contents by specifying the recursive option:

```
get -r someDirectory
```

We can tell SFTP to maintain the appropriate permissions and access times by using the “-P” or “-p” flag:

```
get -Pr someDirectory
```

Transferring Local Files to the Remote System

Transferring files to the remote system is just as easily accomplished by using the appropriately named “put” command:

```
put localFile
```

```
Uploading localFile to /home/demouser/localFile
localFile                                     100% 7607     7.4KB/s   00:00
```

The same flags that work with “get” apply to “put”. So to copy an entire local directory, you can issue:

```
put -r localDirectory
```

Note: There is currently a bug in the versions of OpenSSH shipped with current Ubuntu releases (at least 14.04 to 15.10) that prevents the above command from operating correctly. Upon issuing the command above to transfer content to a server using the buggy version of OpenSSH, the following error will be given: `Couldn't canonicalise: No such file or directory`.

To work around this issue, create the destination directory on the remote end first by typing `mkdir localDirectory`. Afterwards, the above command should complete without error.

One familiar tool that is useful when downloading and uploading files is the “df” command, which works similar to the command line version. Using this, you can check that you have enough space to complete the transfers you are interested in:

```
df -h
```

SCROLL TO TOP

```
    Size     Used    Avail   (root)    %Capacity
  19.9GB   1016MB   17.9GB   18.9GB           4%
```

Please note, that there is no local variation of this command, but we can get around that by issuing the "!" command.

The "!" command drops us into a local shell, where we can run any command available on our local system. We can check disk usage by typing:

```
!
df -h
```

```
Filesystem      Size   Used  Avail Capacity  Mounted on
/dev/disk0s2   595Gi   52Gi  544Gi     9%    /
devfs          181Ki  181Ki    0Bi   100%    /dev
map -hosts       0Bi    0Bi    0Bi   100%    /net
map auto_home    0Bi    0Bi    0Bi   100%    /home
```

Any other local command will work as expected. To return to your SFTP session, type:

```
exit
```

You should now see the SFTP prompt return.

Simple File Manipulations with SFTP

SFTP allows you to perform the type of basic file maintenance that is useful when working with file hierarchies.

For instance, you can change the owner of a file on the remote system with:

SCROLL TO TOP

```
chown userID file
```

Notice how, unlike the system “chmod” command, the SFTP command does not accept usernames, but instead uses UIDs. Unfortunately, there is no easy way to know the appropriate UID from within the SFTP interface.

An involved work around could be accomplished with:

```
get /etc/passwd
!less passwd
```

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
. . .
```

Notice how instead of giving the “!” command by itself, we’ve used it as a prefix for a local shell command. This works to run any command available on our local machine and could have been used with the local “df” command earlier.

The UID will be in the third column of the file, as delineated by colon characters.

Similarly, we can change the group owner of a file with:

```
chgrp groupID file
```

Again, there is no easy way to get a listing of the remote system’s groups. We can work around it with the following command:

```
get /etc/group
!less group


root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:
tty:x:5:
disk:x:6:
lp:x:7:
. . .
```

The third column holds the ID of the group associated with name in the first column. This is what we are looking for.

Thankfully, the “chmod” command works as expected on the remote file system:

```
chmod 777 publicFile


Changing mode on /home/demouser/publicFile
```

There is no command for manipulating local file permissions, but you can set the local umask, so that any files copied to the local system will have the appropriate permissions.

That can be done with the “lumask” command:

```
lumask 022


Local umask: 022
```

Now all regular files downloaded (as long as the “-p” flag is not used) will have 644 permissions.

SCROLL TO TOP

SFTP allows you to create directories on both local and remote systems with “lmkdir” and “mkdir” respectively. These work as expected.

The rest of the file commands target only the remote filesystem:

```
ln
rm
rmdir
```

These commands replicate the basic behavior of the shell versions. If you need to perform these actions on the local file system, remember that you can drop into a shell by issuing this command:

```
!
```

Or execute a single command on the local system by prefacing the command with “!” like so:

```
!chmod 644 somefile
```

When you are finished with your SFTP session, use “exit” or “bye” to close the connection.

```
bye
```

Conclusion

Although SFTP is a simple tool, it is very useful for administrating servers and transferring files between them.

For example, you can use SFTP to enable particular users to transfer files without SSH access. For more information on this process, check out our tutorial on How To

Enable SFTP Without Shell Access on Ubuntu 16.04 and on How To Enable SFTP Without Shell Access on CentOS 7.

If you are used to using FTP or SCP to accomplish your transfers, SFTP is a good way to leverage the strengths of both. While it is not appropriate for every situation, it is a flexible tool to have in your repertoire.

By Justin Ellingwood

Was this helpful? Yes No

Report an issue

Related

TUTORIAL

How to Add and Delete Users on Ubuntu 16.04

Learning how to manage users effectively is an essential skill for any Linux system administrator. In ...

TUTORIAL

How to Add and Delete Users on Ubuntu 18.04

Learning how to manage users effectively is an essential skill for any Linux system administrator. In ...

TUTORIAL

TUTORIAL

SCROLL TO TOP

Still looking for an answer?

Ask a question

Search for more help

55 Comments

Leave a comment...

SCROLL TO TOP

Sign In to Comment

yelinaung *August 18, 2013*

0 Awesome tuts. Thanks.

theguycalledsam *August 22, 2013*

0 Easier to use FilaZilla - hopefully my tutorial will be here soon.

pablo *October 23, 2013*

2 @Samuel, Sorry to steal your thunder: How To Use Filezilla to Transfer and Manage Files Securely on your VPS.

> **How To Use Filezilla to Transfer and Manage Files Securely on your VPS**
> by Pablo Carranza
>
> This article will teach you how to use Filezilla to transfer and manage files securely on your VPS.

mcmaster97330 *March 21, 2019*

1 Filezilla (and Cyberduck and others) are fine if you're on a personal computer, but if you want to transfer between two servers (i.e., two droplets) you'll be happy to have these instructions. Sure beats downloading files from the source server to your computer and then uploading them to the destination server.

sauarav23 *March 7, 2014*

0 -r works with put ?? i am trying , put -r localfile , and it is saying , invalid flag -r

SCROLL TO TOP

larrylanden *March 26, 2014*

0 Please help. Using "put -r localDirectory" as a template (I want to upload all the files and folders from a folder on my local machine) I ran: sftp> lpwd Local working directory: /Users/Larry/Documents/Website sftp> put -r . But the results had errors: Uploading ./ to /var/www/html/. remote open("/var/www/html/.DS_Store"): Permission denied Uploading of file ./.DS_Store to /var/www/html/.DS_Store failed! remote open("/var/www/html/.htaccess"): Permission denied Uploading of file ./.htaccess to /var/www/html/.htaccess failed! remote open("/var/www/html/index.php"): Permission denied Uploading of file ./index.php to /var/www/html/index.php failed! Not sure what is wrong, or how to fix it. Perhaps locally I have to be one directory above the desired folder to copy? sftp> lcd .. sftp> lpwd Local working directory: /Users/Larry/Documents sftp> put -r Website Uploading Website/ to /var/www/html/Website Couldn't canonicalise: No such file or directory Unable to canonicalise path "/var/www/html/Website" sftp>

jellingwood *March 26, 2014*

0 Larry: It looks like you're trying to upload files into a directory on the remote server that you do not have permission to write to. There are a few ways around this. You could upload them to a directory on the remote server that you do have access to, like your home directory, and then sign in through SSH and move the files over to the correct location (using sudo or by signing in with root). Another alternative is to log in as the root user when connecting through SFTP by giving a command like `sftp root@your_server_ip`. You would then have adequate permissions to transfer the files to the web root as you are attempting to do. Please write back if you have more questions.

bing *January 10, 2016*

0 Hi Jellingwood,

I got stuck at the same place. The problem is a bit different from Larry's. So when I followed the `mkdir localdirectory` step, I ran:

```
mkdir /Desktop/MyWebsite
```

it shows:

Couldn't create directory: No such file or directory.

Why and how to fix? :/

SCROLL TO TOP

EDIT:
Found a way to fix this.
`lcd` to the upper level of the local directory to upload. e.g. Desktop
then run the `mkdir` e.g. `mkdir MyWebsite`
And do the put -r . there.
But still I don't know why `mkdir /Desktop/MyWebsite` does not work.

ossie *May 10, 2014*

0 I had changed my ssh port when i configured my server so i use
for example ssh -p 4444 username@server_ip_addr but how can i do the same for sftp i tried sftp -p 4444 username@server_ip_addr but i did not work
connection closing

catherinefawcett *May 20, 2014*

0 You need to give the argument -oPort sftp -oPort 4444
username@server_ip_addr

sndr *November 3, 2016*

0 I had to add a `=` between `-oPort` and the port number to make it work:

```
sftp -oPort=4444 username@server_ip_addr
```

alishaaukani+digoc *June 21, 2014*

0 Hey, I can ssh onto my droplet, but if I type "put", it says
"No command 'put' found". It does the same for commands like "lpwd" and "lcd".
Any idea about what's happening?

jellingwood *June 21, 2014*

0 alishaaukani+digoc: You need to use the `sftp`
command instead of `ssh` when you wish to use the SFTP
functionality. This will take you into an SFTP session instead of a normal

SCROLL TO TOP

SSH session, and allow you to use the commands you mention and transfer files. Let me know if you have any additional questions.

webghostdeveloper *August 20, 2014*

3 For changed ports

if
sftp -oPort port*number username@server*ip_addr

doesn't work, this should:

sftp -oPort=port*number username@server*ip_addr

> beconomist *March 18, 2015*
>
> 0 sftp -oPort=portnumber username@serverip_addr
>
> Works for me. Thanks

Okidoki *September 7, 2014*

0 Help, I'm stuck in the first step. When I type `ssh username@remote_hostname_or_IP` I get `Permission denied (publickey)`. I get same answer when typing `sftp username@remote_hostname_or_IP`. Of course, I changed remote*hostname*or_IP to the appropriate IPv4 address.

The SSH key works great on Putty program, though. I logged in without problem.

> danielemm *September 7, 2014*
>
> 1 Okidoki, you need to use your root's account password no the password of the server (the one you have received from digitalocean and you use to connect via ssh).
> It works for me.
>
> Okidoki *September 7, 2014*
>
> 0 Hi danielemm, I followed this tutorial all the way to the end to create SSH by editing /etc/ssh/sshd_config to

SCROLL TO TOP

```
[...]
PasswordAuthentication no
[...]
UsePAM no
[...]
```

When I commented back PasswordAuthentication and changed UsePAM to yes I was able to use the root's password as you said so. However, is there a way to disable username/password logins to achieve better security while allowing sftp access at the same time?

> **How To Create SSH Keys With PuTTY to Connect to a VPS**
> by Pablo Carranza
>
> This tutorial runs through creating SSH keys with PuTTY to connect to your virtual server.

didit *September 18, 2014*

0

When I am using SFTP and upload my site folder, I got :

mysite.com/ is not regular file.

I have tried upload using both "User" and "root"

What does it mean ? 'Not regular File' ?

ralpheiligan *September 20, 2014*

0

Perfect!

praveen187 *September 24, 2014*

0

i have used sftp username@remote*hostname*or_IP command and it takes me to sftp prompt. BUt i directly want the file to be transfered to the remote location without prompting to SFTP prompt.

Is there any solution to this

SCROLL TO TOP

simerng *November 21, 2014*

0 It's even a lot easier to connect via coreFTP as opposed to Filezilla, Filezilla kept asking to type password for each file i wanted to upload.

Dayandnightpers *January 13, 2015*

0 I do not understand. You state it has to be this method. What exactly is the username to be used? My login name for digitalocean is an email address so is it mydetails@domain.com@ipaddress ??

> jellingwood *January 13, 2015*
>
> 0 @Dayandnightpers: In this case, you would not be using your username for your DigitalOcean *account*, you would need to use the username for your server.
>
> By default, most of the distributions use the `root` user account as the default account for your server. If you have completed some of the other guides on this site, you may have configured another account. So you need to use whichever account you use to log into your server.
>
> If you did not include SSH keys when you created your server, you would have received an email with the login credentials for your new server. These are the details you need.

retador *April 23, 2015*

0 I want to have a shared hosting server, with multiple domains and obviously different content on each. Do I have to create a folder for each domain in my home or root directory?

> kamaln7 MOD *April 27, 2015*
>
> 0 Yes, that is correct. Where you create the directories depends on how your server is structured. This is all explained in this tutorial: How To Set Up Apache Virtual Hosts on Ubuntu 14.04 LTS.
>
> How To Set Up Apache Virtual Hosts on Ubuntu 14.04 LTS
> by Justin Ellingwood
> SCROLL TO TOP

The Apache web server is the most popular way to serve web content on the internet. Apache has the ability to serve multiple domains from a single server by using a mechanism called "virtual hosts". If a virtual host is configured

Akshay11 *April 25, 2015*

0 How to set local directory path??

lpwd and pwd showing same path. plz help.

mefav *September 10, 2015*

0 make sure you're using your local machine cmd or gitbash but not putty that already connected inside the vps. Then you will see your local path.

Load More Comments

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

Featured on Community Intro to Kubernetes Learn Python 3 Machine Learning in Python Getting started with Go Migrate Node.js to Kubernetes

DigitalOcean Products Droplets Managed Databases Managed Kubernetes Spaces Object Storage Marketplace

Welcome to the developer cloud

DigitalOcean makes it simple to launch in the cloud and scale up as you grow – whether you're running one virtual machine or ten thousand.

Learn More

SCROLL TO TOP

© 2019 DigitalOcean, LLC. All rights reserved.

Company

- About
- Leadership
- Blog
- Careers
- Partners
- Referral Program
- Press
- Legal & Security

Products

- Products Overview
- Pricing
- Droplets
- Kubernetes
- Managed Databases
- Spaces
- Marketplace
- Load Balancers
- Block Storage
- Tools & Integrations
- API
- Documentation
- Release Notes

Community

- Tutorials
- Q&A
- Tools and Integrations
- Tags
- Product Ideas
- Meetups
- Write for DOnations
- Droplets for Demos
- Hatch Startup Program
- Shop Swag
- Research Program
- Currents Research

Contact

- Support
- Sales
- Report Abuse
- System Status

SCROLL TO TOP

SCROLL TO TOP