How To Install an FTP Server on CentOS 7 With VSFTPD

phoenixnap.com/kb/how-to-setup-ftp-server-install-vsftpd-centos-7

February 28, 2019

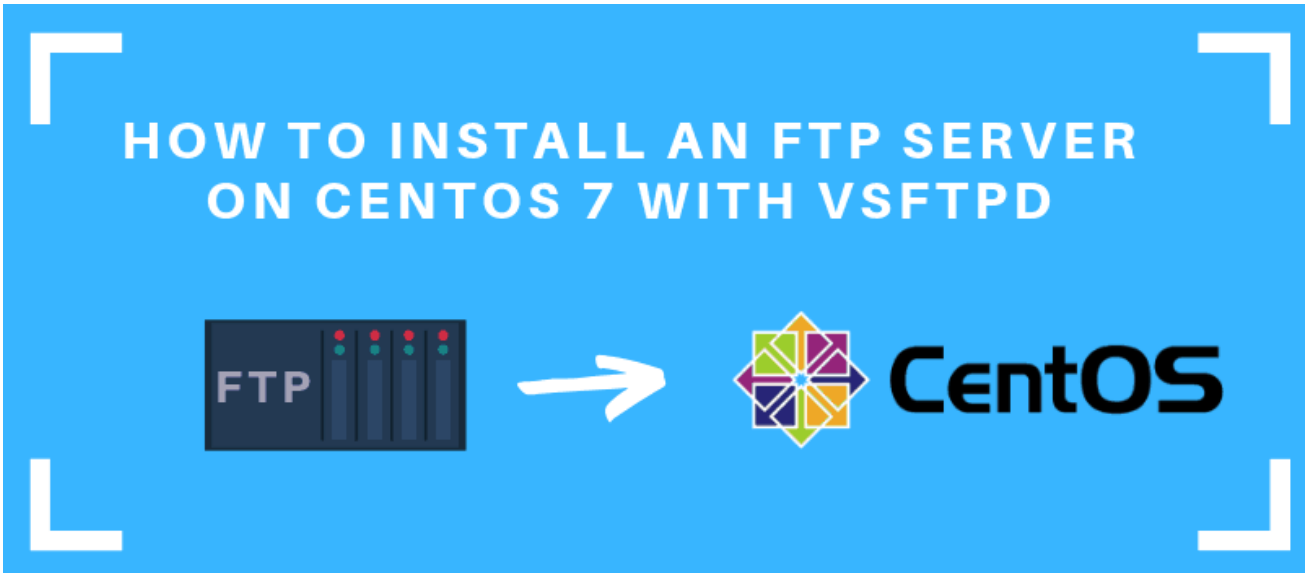
Introduction

If you are looking to install an FTP server, you can't beat the simplicity of VSFTPD.

FTP stands for File Transfer Protocol. It has been a standard method for transferring files between computers for decades.

Although security measures have been added, FTP is by nature an insecure method for transferring files. However, it can be useful when making files available to multiple users, or when working in a secure and private network.

This guide will show you how to configure and **install an FTP server using VSFTPD on CentOS 7.**



Prerequisites

- Access to a user account with sudo privileges
- The **yum** package manager, installed by default
- A text editor of your choice

Install FTP Server on CentOS 7

Step 1: Install FTP Service With VSFTPD

Step 1: Install FTP Service With VSFTPD

1. Start by updating the package manager:

```
sudo yum update
```

Allow the process to complete.

This guide uses the **vsftpd** (VSFTPD stands for “Very Secure FTP Daemon software package”). It’s a relatively easy software utility to use for creating an **FTP server**.

2. Install VSFTPD software with the following command:

```
sudo yum install vsftpd
```

Allow the operation to complete.

3. Start the service and set it to launch when the system boots with the following:

```
sudo systemctl start vsftpd
```

```
sudo systemctl enable vsftpd
```

4. Next, create a rule for your firewall to allow FTP traffic on Port 21:

```
sudo firewall-cmd --zone=public --permanent --add-port=21/tcp
```

```
sudo firewall-cmd --zone=public --permanent --add-service=ftp
```

```
sudo firewall-cmd –reload
```

Note: If you use a different firewall application, refer to the documentation to configure it correctly for Port 21. Also, some FTP clients use Port 20, so you may wish to include that rule as well. Simply copy the first line, and replace 21 with 20.

Step 2: Configuring VSFTPD

The behavior of the FTP service on your server is determined by the **/etc/vsftpd/vsftpd.conf** configuration file.

1. Before starting, create a copy of the default configuration file:

```
sudo cp /etc/vsftpd/vsftpd.conf /etc/vsftpd/vsftpd.conf.default
```

This ensures that you have a way to return to the default configuration, in case you change a setting that causes a problem.

2. Next, edit the configuration file with the following command:

2. Next, edit the configuration file with the following command:

```
sudo nano /etc/vsftpd/vsftpd.conf
```

3. Set your FTP server to disable anonymous users and allow local users.

Find the following entries in the configuration file, and edit them to match the following:

```
anonymous_enable=NO
```

```
local_enable=YES
```

This is an important step. Anonymous access is a risky – you should avoid it unless you understand the risks.

4. Next, allow a logged-in user to upload files to your FTP server.

Find the following entry, and edit to match as follows:

```
write_enable-YES
```

Note: By default, this line starts with a # sign to indicate it's a comment. Commenting is a useful way to turn commands on and off. The # sign can also be used to make notes in the file without the system interpreting them as instructions.

5. Limit FTP users to their own home directory. This is often called "jail" or "chroot jail." Find and adjust the entry to match the following:

```
chroot_local_user=YES
```

```
allow_writeable_chroot=YES
```

Note: for test purposes, the **allow_writeable_chroot=YES** option will create a functioning FTP server that you can test and use. Some administrators advocate the use of the **user_sub_token** option for better security. Refer to the vsftpd documentation for more information on this option.

6.The **vsftpd** utility provides a way to create an approved user list. To manage users this way, find the **userlist_enable** entry, then edit the file to look as follows:

```
userlist_enable=YES
```

```
userlist_file=/etc/vsftpd/user_list
```

```
userlist_deny=NO
```

You can now edit the **/etc/vsftpd/user_list** file, and add your list of users. (List one per line.) The **userlist_deny** option lets you specify users to be included; setting it to **yes** would change the list to users that are blocked.

7. Once you're finished editing the configuration file, save your changes. Restart the **vsftpd** service to apply changes:

```
sudo systemctl restart vsftpd
```

Step 3: Create a New FTP user

1. To create a new FTP user enter the following:

```
sudo adduser testuser
```

```
sudo passwd testuser
```

The system should prompt you to enter and confirm a password for the new user.

2. Add the new user to the **userlist**:

```
echo "testuser" | sudo tee –a /etc/vsftpd/user_list
```

3. Create a directory for the new user, and adjust permissions:

```
sudo mkdir –p /home/testuser/ftp/upload
```

```
sudo chmod 550 /home/testuser/ftp
```

```
sudo chmod 750 /home/testuser/ftp/upload
```

```
sudo chown –R testuser: /home/testuser/ftp
```

Note: This creates a home/testuser directory for the new user, with a special directory for uploads. It sets permissions for uploads only to the /uploads directory.

4. Now, you can log in to your FTP server with the user you created:

```
ftp 192.168.01
```

Replace this IP address with the one from your system. You can find your IP address in Linux with the `ip addr` command.

The system should prompt you for a username – enter **testuser** (or whatever username you created earlier). Type the password, and the system should log you in.

Step 4: Test the FTP server

To Test the FTP Server Locally, use the command:

```
ftp localhost
```

```
Trying 127.0.0.1...
Connected to localhost (127.0.0.1).
220 (vsFTPd 2.2.2)
Name (localhost:root): ftpuser
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
***
```

To Test remotely, use the command:

`ftp your.ftp.server.com`

```
Connected to your.ftp.server.com.
220 (vsFTPd 2.2.2)
Name (your.ftp.server.com:yourname):
Name (localhost:root): ftpuser
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
***
```

Note: While some security measures have been included in this guide, it is strongly recommended that you familiarize yourself with the latest security protocols before implementing an FTP server in a production environment. This is especially important if

you're creating an FTP server that's open to the internet – many security breaches originate through the FTP protocol.

Conclusion

Now you know **how to set up and install an FTP server on Centos 7 with VSFTPD**. You should be able to login to your server via FTP and start transferring files.

Next, You Should Also Read:

Next you should also read

Web Servers,SysAdmin

How To Install an FTP Server on CentOS 7 With VSFTPD

February 28, 2019

In this updated Tutorial, learn how to Setup FTP Server with VSFTPD on CentOS 7. FTP stands for File Transfer...

Read More

SysAdmin

How to Find Files in Linux With the Find Command

December 19, 2018

The find command is a useful command-line tool in Linux. It allows you to use the terminal to search for a...

Read More

Web Servers,SysAdmin,Bare Metal Servers

How to Restart or Reboot Linux Server From the Command Line

October 22, 2018

There's a reason that tech support asks you if you've rebooted your Linux server. It's

There's a reason that tech support asks you if you've rebooted your Linux server. It's cliched but true -...

Read More

Security,SysAdmin

How to Reset or Change the Root Password in Linux

October 22, 2018

In Linux, root privileges (or root access) refers to a user account that has full access to all files,...

Read More

Web Servers,SysAdmin

How to Install phpMyAdmin on CentOS 7

October 22, 2018

This guide is for users who have already configured a CentOS server and installed the Apache HTTP services,...

Read More

Author

Dejan Tucakov

Dejan is the Technical Writing Team Lead at phoenixNAP with over 5 years of experience in Web publishing. Prior to joining phoenixNAP, he was Chief Editor of several websites striving to advocate for emerging technologies. He is dedicated to simplifying complex notions and providing meaningful insight into datacenter and cloud technology.