# Introduction

VSFTPD stands for "Very Secure FTP Daemon" is a GPL licensed FTP server for UNIX systems. It is licensed under the GNU General Public License. It supports IPv6 and SSL. vsftpd supports explicit (since 2.0.0) and implicit (since 2.1.0) FTPS. vsftpd is the default FTP server in the Ubuntu, CentOS, Fedora, NimbleX, Slackware and RHEL Linux distributions. It is secure and extremely fast. It is stable. VSFTPD is a mature and trusted solution which supports virtual users with PAM (pluggable authentication modules). A virtual user is a user login which does not exist as a real login on the system in /etc/passwd and /etc/shadow file. Virtual users can therefore be more secure than real users, because a compromised account can only use the FTP server but cannot login to system to use other services such as SSH or SMTP.

In July 2011, it was discovered that VSFTPD version 2.3.4 downloadable from the master site had been compromised. Users logging into a compromised vsftpd-2.3.4 server may issue a ":)" smiley-face as the username and gain a command shell on port 6200. This was not an issue of a security hole in VSFTPD, instead, someone had uploaded a different version of VSFTPD which contained a backdoor. Since then, the site was moved to Google App Engine.

# Features

Despite being small for purposes of speed and security, many more complicated FTP setups are achievable with vsftpd! vsftpd can handle:

1. Virtual IP configurations
2. Virtual users
3. Standalone or inetd operation
4. Powerful per-user configurability
5. Bandwidth throttling
6. Per-source-IP configurability
7. Per-source-IP limits
8. IPv6
9. Encryption support through SSL integration...

# Configuration Instructions and Basic Setup

## Download

The latest vsftpd release is v3.0.2, currently at <https://security.appspot.com/downloads/vsftpd-3.0.2.tar.gz>

Or you can install via apt-get like sudo apt-get install vsftpd

Now you can configure it to either allow "local users" to be able to login via ftp, or "virtual users".

## To disable anonymous login and to enable local users login and give them write permissions:

Code:  
# No anonymous login   
anonymous\_enable=NO   
# Let local users login   
# If you connect from the internet with local users, you should enable TLS/SSL/FTPS   
local\_enable=YES  
# Write permissions   
write\_enable=YES

## To chroot users

To jail / chroot users (not the VSFTPD service), there are three choices. Search for "chroot\_local\_users" on the file and consider one of the following: Code:  
# 1. All users are jailed by default:  
chroot\_local\_user=YES  
chroot\_list\_enable=NO

# 2. Just some users are jailed:  
chroot\_local\_user=NO  
chroot\_list\_enable=YES  
# Create the file /etc/vsftpd.chroot\_list with a list of the jailed users.

# 3. Just some users are "free":  
chroot\_local\_user=YES  
chroot\_list\_enable=YES  
# Create the file /etc/vsftpd.chroot\_list with a list of the "free" users.

## To deny (or allow) just some users to login

To deny some users to login, add the following options in the end of the file: Code: userlist\_deny=YES  
userlist\_file=/etc/vsftpd.denied\_users  
In the file /etc/vsftpd.denied\_users add the username of the users that can't login. One username per line.

## To allow just some users to login:

Code: userlist\_deny=NO  
userlist\_enable=YES  
userlist\_file=/etc/vsftpd.allowed\_users  
In the file /etc/vsftpd.allowed\_users add the username of the users that can login.

The not allowed users will get an error that they can't login before they type their password.

## TLS/SSL/FTPS

NOTE: you definitely should use this if you connect from the Internet to your box, otherwise passwords will be sent in plaintext, etc.

To use vsftpd with encryption (it's safer), change or add the following options (some options aren't on the original config file, so add them): Code: ssl\_enable=YES  
allow\_anon\_ssl=NO  
force\_local\_data\_ssl=YES  
force\_local\_logins\_ssl=YES  
ssl\_tlsv1=YES  
ssl\_sslv2=NO  
ssl\_sslv3=NO  
# Filezilla uses port 21 if you don't set any port  
# in Servertype "FTPES - FTP over explicit TLS/SSL"  
# Port 990 is the default used for FTPS protocol.  
# Uncomment it if you want/have to use port 990.  
# listen\_port=990  
No need to create a certificate if openssl package is installed!

Install Filezilla (on the client side), and use the Servertype "FTPES - FTP over explicit TLS/SSL" option to connect to your server with TLS/SSL/FTPS.

## Additional Options

Here are some other available options. The values are examples: Code: # Show hidden files and the "." and ".." folders.  
# Useful to not write over hidden files:  
force\_dot\_files=YES

# Hide the info about the owner (user and group) of the files.  
hide\_ids=YES

# Connection limit for each IP:  
max\_per\_ip=2

# Maximum number of clients:  
max\_clients=20

## Apply new configuration settings

Don't forget that to apply new configurations, you must restart the vsftpd service. Code:  
sudo /etc/init.d/vsftpd restart

## Webmin Module

For those who use webadmin, there is a module for VSFTPD here <http://www.webmin.com/third.html>.

## Set pasv\_min\_port and pasv\_max\_port in /etc/vsftpd.conf and allow outbound connections in the ports you set in your firewall.

Code:  
pasv\_min\_port=12000  
pasv\_max\_port=12100

## Virtual users with TLS/SSL/FTPS and a common upload directory - Complicated VSFTPD

Virtual users are users that do not exist on the system - they are not in /etc/passwd, do not have a home directory on the system, can not login but in vsftpd - or if they do exist, they can login in vsftpd with a non system password - security.

You can set different definitions to each virtual user, granting to each of these users different permissions. If TLS/SSL/FTPS and virtual users are enabled, the level of security of your vsftpd server is increased: encrypted passwords, with passwords that are not used on the system, and users that can't access directly to their home directory (if you want).

The following example is based and adapted on the example for virtual users in vsftpd site, on documentation and the very good examples in this forum that can be found here and here. Currently there is a restriction that with guest\_enable enabled, local users also get mapped to guest\_username. This is a polite way to say that if the default vsftpd PAM file is used, the system users will be guests too. To avoid confusions change the PAM file used by vsftpd to authenticate only virtual users, make all vsftpd users as virtual users and set their passwords, home and permissions based on this example.

# The workshop

## Create The Virtual Users Database

To create a "db4" format file to store usernames (another option here would be an apache htpasswd style file, not discussed), first create a plain text files with the usernames and password on alternating lines. For e.g. create user called "vivek" with password called "vivekpass" and sayali with password "sayalipass":

# mkdir /etc/vsftpd # if necessary  
# cd /etc/vsftpd  
# sudo gedit vusers.txt

## Sample output:

vivek  
vivekpass  
sayali  
sayalipass

Next, create the actual database file like this (may require the db\_util package to be installed first):

# db\_load -T -t hash -f vusers.txt vsftpd-virtual-user.db  
# chmod 600 vsftpd-virtual-user.db # make it not global readable  
# rm vusers.txt

## Configure VSFTPD for virtual user

Edit the vsftpd configuration file (/etc/vsftpd.conf). Add or correct the following configuration options, depending on if they're already listed somewhere in the file or not (or just add these all to the bottom):

anonymous\_enable=NO  
local\_enable=YES  
# Virtual users will use the same privileges as local users.  
# It will grant write access to virtual users. Virtual users will use the  
# same privileges as anonymous users, which tends to be more restrictive  
# (especially in terms of write access).  
virtual\_use\_local\_privs=YES  
write\_enable=YES

# Set the name of the PAM service vsftpd will use  
pam\_service\_name=vsftpd.virtual

# Activates virtual users  
guest\_enable=YES

# Automatically generate a home directory for each virtual user, based on a template.  
# For example, if the home directory of the real user specified via guest\_username is  
# /home/virtual/$USER, and user\_sub\_token is set to $USER, then when virtual user vivek  
# logs in, he will end up (usually chroot()'ed) in the directory /home/virtual/vivek.  
# This option also takes affect if local\_root contains user\_sub\_token.  
user\_sub\_token=$USER

# Usually this is mapped to Apache virtual hosting docroot, so that  
# Users can upload files  
local\_root=/home/vftp/$USER

# Chroot user and lock down to their home dirs  
chroot\_local\_user=YES

# Hide ids from user  
hide\_ids=YES

Save and close the file.

## Create a PAM File Which Uses Your New Database

The following PAM is used to authenticate users using your new database. Create /etc/pam.d/vsftpd.virtual: # sudo gedit /etc/pam.d/vsftpd.virtual

## Append (or create with) the following:

#%PAM-1.0  
auth required pam\_userdb.so db=/etc/vsftpd/vsftpd-virtual-user  
account required pam\_userdb.so db=/etc/vsftpd/vsftpd-virtual-user  
session required pam\_loginuid.so

Create The Location Of The Files

You need to set up the location of the files / dirs for the virtual users. Type the following command: # mkdir /home/vftp  
# mkdir -p /home/vftp/{vivek,sayali}  
# chown -R ftp:ftp /home/vftp

## Restart The FTP Server

Type the following command:  
# service vsftpd restart

## Test Your Setup

Open another shell session and type:  
$ ftp localhost

## Sample success output:

Connected to ftp.nixcraft.net.in.  
Name (localhost:root): vivek  
331 Please specify the password.[user now types in vivekpass]  
Password:  
230 Login successful.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp>

# Other docs

This article describes how to install and configure a vsFTPD server on CentOS®, Red Hat® Enterprise Linux® (RHEL), and the Ubuntu® operating system.

**Note:** You must be logged in through SSH as the root user to use the instructions in this article.

### Install vsFTPD

Use the following commands on the different Linux® distributions to install a vsFTPD server:

**CentOS and RHEL**

yum -y install vsftpd

**Ubuntu operating system**

apt-get install vsftpd

The installation process generates a configuration file. For CentOS and RHEL, the file is named  
**/etc/vsftpd/vsftpd.conf**, and for the Ubuntu operating system, the file is named **/etc/vsftpd.conf**. Use the instructions  
in the following sections to configure the settings in the vsFTPD configuration file.

### Configure vsFTPD

Open the vsFTPD configuration file in a file editor or by using vi, and replace the contents of the file  
with the following lines:

anonymous\_enable=NO  
local\_enable=YES  
write\_enable=YES  
local\_umask=022  
dirmessage\_enable=YES  
xferlog\_enable=YES  
connect\_from\_port\_20=YES  
xferlog\_std\_format=YES  
listen=YES  
pam\_service\_name=vsftpd  
userlist\_enable=YES  
tcp\_wrappers=YES  
pasv\_min\_port=60000  
pasv\_max\_port=65000

If you want to enable chroot jails, add the following lines at the bottom of the configuration file:

chroot\_local\_user=YES  
chroot\_list\_enable=YES  
chroot\_list\_file=/etc/vsftpd/vsftpd.chroot\_list

You must create a **vsftp.chroot\_list** file and put any users in it who are *not* chrooted. All users are chrooted by default. You must create the file even if you don't have any users to put in it.

**Note:** For the Ubuntu operating system, the line for the chroot list file is chroot\_list\_file=/etc/vsftpd.chroot\_list.

If you want to enable a user to use file access control lists (FACLs) or a set a group permission by default, add the following lines at the bottom of the configuration file:

file\_open\_mode=XXXX  
local\_umask=XXX

Here, you can change file\_open\_mode to 0775, 0664, and so on to meet your basic permission needs. You might not need to combine it with umask, depending on what you want to do.

Umask removes permissions from the files. For example, a file with 777 becomes 755 with a umask of 022 (the default). This restricts access for security purposes. Some people mistakenly set the umask to 000, thinking that the files will then show up as 777. This distinction is important. While file\_open\_mode tells vsFTPD the default permissions to use, umask only takes away permissions, it can never grant them.

#### Restart and enable vsFTPD

After you edit the configuation file, you must restart the vsFTPD service for the changes to take effect. Use  
the following command to restart vsFTPD on CentOS, RHEL, and the Ubuntu operating system:

systemctl restart vsftpd

After vsFTPD restarts, you should also configure it to start when the server boots. Use the following command to enable  
vsFTPD to start at boot on CentOS, RHEL, and the Ubuntu operating system:

systemctl enable vsftpd

#### Allow vsFTPD through the firewall

The final step is to allow vsFTPD through your server firewall by using the following commands on the different Linux distributions:

**CentOS and RHEL**

iptables -I RH-Firewall-1-INPUT -p tcp --dport 21 -m comment --comment "FTP" -j ACCEPT  
iptables -I RH-Firewall-1-INPUT -p tcp -m multiport --dports 60000:65000 -m comment --comment "FTP passive mode ports" -j ACCEPT  
/etc/init.d/iptables save

**The Ubuntu operating system**

ufw allow 21  
ufw allow proto tcp from any to any port 60000:65000