




trusty (5) vsftpd.conf.5.gz

Provided by: vsftpd\_3.0.2-1ubuntu2\_i386 

## NAME

vsftpd.conf - config file for vsftpd

## DESCRIPTION

vsftpd.conf may be used to control various aspects of vsftpd's behaviour. By default, vsftpd looks for this file at the location **/etc/vsftpd.conf**. However, you may override this by specifying a command line argument to vsftpd. The command line argument is the pathname of the configuration file for vsftpd. This behaviour is useful because you may wish to use an advanced inetd such as **xinetd** to launch vsftpd with different configuration files on a per virtual host basis.

## FORMAT

The format of vsftpd.conf is very simple. Each line is either a comment or a directive. Comment lines start with a **#** and are ignored. A directive line has the format:

option=value

It is important to note that it is an error to put any space between the option, = and value.

Each setting has a compiled in default which may be modified in the configuration file.

## BOOLEAN OPTIONS

Below is a list of boolean options. The value for a boolean option may be set to **YES** or **NO**.

### allow\_anon\_ssl

Only applies if **ssl\_enable** is active. If set to **YES**, anonymous users will be allowed to use secured SSL connections.

Default: NO

### anon\_mkdir\_write\_enable

If set to **YES**, anonymous users will be permitted to create new directories under certain conditions. For this to work, the option **write\_enable** must be activated, and the anonymous ftp user must have write permission on the parent directory.

Default: NO

### anon\_other\_write\_enable

If set to **YES**, anonymous users will be permitted to perform write operations other than upload and create directory, such as deletion and renaming. This is generally not recommended but included for completeness.

Default: NO

### anon\_upload\_enable

If set to **YES**, anonymous users will be permitted to upload files under certain conditions. For this to work, the option **write\_enable** must be activated, and the anonymous ftp user must have write permission on desired upload locations. This setting is also required for virtual users to upload; by default, virtual users are treated with anonymous (i.e. maximally restricted) privilege.

Default: NO

### anon\_world\_readable\_only

When enabled, anonymous users will only be allowed to download files which are world readable. This is recognising that the ftp user may own files, especially in the presence of uploads.

Default: YES

### anonymous\_enable

Controls whether anonymous logins are permitted or not. If enabled, both the usernames **ftp** and **anonymous** are recognised as anonymous logins.

Default: YES

#### **ascii\_download\_enable**

When enabled, ASCII mode data transfers will be honoured on downloads.

Default: NO

#### **ascii\_upload\_enable**

When enabled, ASCII mode data transfers will be honoured on uploads.

Default: NO

#### **async\_abor\_enable**

When enabled, a special FTP command known as "async ABOR" will be enabled. Only ill advised FTP clients will use this feature. Additionally, this feature is awkward to handle, so it is disabled by default. Unfortunately, some FTP clients will hang when cancelling a transfer unless this feature is available, so you may wish to enable it.

Default: NO

#### **background**

When enabled, and vsftpd is started in "listen" mode, vsftpd will background the listener process. i.e. control will immediately be returned to the shell which launched vsftpd.

Default: NO

#### **check\_shell**

Note! This option only has an effect for non-PAM builds of vsftpd. If disabled, vsftpd will not check `/etc/shells` for a valid user shell for local logins.

Default: YES

#### **chmod\_enable**

When enabled, allows use of the SITE CHMOD command. NOTE! This only applies to local users. Anonymous users never get to use SITE CHMOD.

Default: YES

#### **chown\_uploads**

If enabled, all anonymously uploaded files will have the ownership changed to the user specified in the setting **chown\_username**. This is useful from an administrative, and perhaps security, standpoint.

Default: NO

#### **chroot\_list\_enable**

If activated, you may provide a list of local users who are placed in a chroot() jail in their home directory upon login. The meaning is slightly different if chroot\_local\_user is set to YES. In this case, the list becomes a list of users which are NOT to be placed in a chroot() jail. By default, the file containing this list is `/etc/vsftpd.chroot_list`, but you may override this with the **chroot\_list\_file** setting.

Default: NO

#### **chroot\_local\_user**

If set to YES, local users will be (by default) placed in a chroot() jail in their home directory after login. **Warning:** This option has security implications, especially if the users have upload permission, or shell access. Only enable if you know what you are doing. Note that these security implications are not vsftpd specific. They apply to all FTP daemons which offer to put local users in chroot() jails.

Default: NO

#### **connect\_from\_port\_20**

This controls whether PORT style data connections use port 20 (ftp-data) on the server machine. For security reasons, some clients may insist that this is the case. Conversely, disabling this option enables vsftpd to run with slightly less privilege.

Default: NO (but the sample config file enables it)

#### **debug\_ssl**

If true, OpenSSL connection diagnostics are dumped to the vsftpd

log file. (Added in v2.0.6).

Default: NO

#### **delete\_failed\_uploads**

If true, any failed upload files are deleted. (Added in v2.0.7).

Default: NO

#### **deny\_email\_enable**

If activated, you may provide a list of anonymous password e-mail responses which cause login to be denied. By default, the file containing this list is `/etc/vsftpd.banned_emails`, but you may override this with the **banned\_email\_file** setting.

Default: NO

#### **dirlist\_enable**

If set to NO, all directory list commands will give permission denied.

Default: YES

#### **dirmessage\_enable**

If enabled, users of the FTP server can be shown messages when they first enter a new directory. By default, a directory is scanned for the file `.message`, but that may be overridden with the configuration setting **message\_file**.

Default: NO (but the sample config file enables it)

#### **download\_enable**

If set to NO, all download requests will give permission denied.

Default: YES

#### **dual\_log\_enable**

If enabled, two log files are generated in parallel, going by default to `/var/log/xferlog` and `/var/log/vsftpd.log`. The former is a wu-ftp style transfer log, parseable by standard tools. The latter is vsftpd's own style log.

Default: NO

#### **force\_dot\_files**

If activated, files and directories starting with `.` will be shown in directory listings even if the "a" flag was not used by the client. This override excludes the `."` and `.."` entries.

Default: NO

#### **force\_anon\_data\_ssl**

Only applies if **ssl\_enable** is activated. If activated, all anonymous logins are forced to use a secure SSL connection in order to send and receive data on data connections.

Default: NO

#### **force\_anon\_logins\_ssl**

Only applies if **ssl\_enable** is activated. If activated, all anonymous logins are forced to use a secure SSL connection in order to send the password.

Default: NO

#### **force\_local\_data\_ssl**

Only applies if **ssl\_enable** is activated. If activated, all non-anonymous logins are forced to use a secure SSL connection in order to send and receive data on data connections.

Default: YES

#### **force\_local\_logins\_ssl**

Only applies if **ssl\_enable** is activated. If activated, all non-anonymous logins are forced to use a secure SSL connection in order to send the password.

Default: YES

#### **guest\_enable**

If enabled, all non-anonymous logins are classed as "guest" logins. A guest login is remapped to the user specified in the **guest\_username** setting.

Default: NO

Default: NO

#### **hide\_ids**

If enabled, all user and group information in directory listings will be displayed as "ftp".

Default: NO

#### **implicit\_ssl**

If enabled, an SSL handshake is the first thing expected on all connections (the FTPS protocol). To support explicit SSL and/or plain text too, a separate vsftpd listener process should be run.

Default: NO

**listen** If enabled, vsftpd will run in standalone mode. This means that vsftpd must not be run from an inetd of some kind. Instead, the vsftpd executable is run once directly. vsftpd itself will then take care of listening for and handling incoming connections.

Default: NO

#### **listen\_ipv6**

Like the listen parameter, except vsftpd will listen on an IPv6 socket instead of an IPv4 one. This parameter and the listen parameter are mutually exclusive.

Default: NO

#### **local\_enable**

Controls whether local logins are permitted or not. If enabled, normal user accounts in `/etc/passwd` (or wherever your PAM config references) may be used to log in. This must be enabled for any non-anonymous login to work, including virtual users.

Default: NO

#### **lock\_upload\_files**

When enabled, all uploads proceed with a write lock on the upload file. All downloads proceed with a shared read lock on the download file. WARNING! Before enabling this, be aware that malicious readers could starve a writer wanting to e.g. append a file.

Default: YES

#### **log\_ftp\_protocol**

When enabled, all FTP requests and responses are logged, providing the option `xferlog_std_format` is not enabled. Useful for debugging.

Default: NO

#### **ls\_recurse\_enable**

When enabled, this setting will allow the use of "ls -R". This is a minor security risk, because a ls -R at the top level of a large site may consume a lot of resources.

Default: NO

#### **mdtm\_write**

When enabled, this setting will allow MDTM to set file modification times (subject to the usual access checks).

Default: YES

#### **no\_anon\_password**

When enabled, this prevents vsftpd from asking for an anonymous password - the anonymous user will log straight in.

Default: NO

#### **no\_log\_lock**

When enabled, this prevents vsftpd from taking a file lock when writing to log files. This option should generally not be enabled. It exists to work around operating system bugs such as the Solaris / Veritas filesystem combination which has been observed to sometimes exhibit hangs trying to lock log files.

Default: NO

#### **one\_process\_model**

If you have a Linux 2.4 kernel, it is possible to use a different security model which only uses one process per connection. It is a less pure security model, but gives you

connection. It is a less pure security model, but gains you performance. You really don't want to enable this unless you know what you are doing, and your site supports huge numbers of simultaneously connected users.

Default: NO

**passwd\_chroot\_enable**

If enabled, along with **chroot\_local\_user**, then a chroot() jail location may be specified on a per-user basis. Each user's jail is derived from their home directory string in /etc/passwd. The occurrence of **./** in the home directory string denotes that the jail is at that particular location in the path.

Default: NO

**pasv\_addr\_resolve**

Set to YES if you want to use a hostname (as opposed to IP address) in the **pasv\_address** option.

Default: NO

**pasv\_enable**

Set to NO if you want to disallow the PASV method of obtaining a data connection.

Default: YES

**pasv\_promiscuous**

Set to YES if you want to disable the PASV security check that ensures the data connection originates from the same IP address as the control connection. Only enable if you know what you are doing! The only legitimate use for this is in some form of secure tunnelling scheme, or perhaps to facilitate FXP support.

Default: NO

**port\_enable**

Set to NO if you want to disallow the PORT method of obtaining a data connection.

Default: YES

**port\_promiscuous**

Set to YES if you want to disable the PORT security check that ensures that outgoing data connections can only connect to the client. Only enable if you know what you are doing!

Default: NO

**require\_cert**

If set to yes, all SSL client connections are required to present a client certificate. The degree of validation applied to this certificate is controlled by **validate\_cert** (Added in v2.0.6).

Default: NO

**require\_ssl\_reuse**

If set to yes, all SSL data connections are required to exhibit SSL session reuse (which proves that they know the same master secret as the control channel). Although this is a secure default, it may break many FTP clients, so you may want to disable it. For a discussion of the consequences, see <http://scarybeastsecurity.blogspot.com/2009/02/vsftpd-210-released.html> (Added in v2.1.0).

Default: YES

**run\_as\_launching\_user**

Set to YES if you want vsftpd to run as the user which launched vsftpd. This is useful where root access is not available. MASSIVE WARNING! Do NOT enable this option unless you totally know what you are doing, as naive use of this option can create massive security problems. Specifically, vsftpd does not / cannot use chroot technology to restrict file access when this option is set (even if launched by root). A poor substitute could be to use a **deny\_file** setting such as **{/\*,\*.\*}**, but the reliability of this cannot compare to chroot, and should not be relied on. If using this option, many restrictions on other options apply. For example, options requiring privilege such as non-anonymous logins, upload ownership changing, connecting from port 20 and listen ports less than 1024 are not expected to work. Other options may be impacted.

Default: NO

Default: NO

#### **secure\_email\_list\_enable**

Set to YES if you want only a specified list of e-mail passwords for anonymous logins to be accepted. This is useful as a low-hassle way of restricting access to low-security content without needing virtual users. When enabled, anonymous logins are prevented unless the password provided is listed in the file specified by the **email\_password\_file** setting. The file format is one password per line, no extra whitespace. The default filename is /etc/vsftpd.email\_passwords.

Default: NO

#### **session\_support**

This controls whether vsftpd attempts to maintain sessions for logins. If vsftpd is maintaining sessions, it will try and update utmp and wtmp. It will also open a pam\_session if using PAM to authenticate, and only close this upon logout. You may wish to disable this if you do not need session logging, and you wish to give vsftpd more opportunity to run with less processes and / or less privilege. NOTE - utmp and wtmp support is only provided with PAM enabled builds.

Default: NO

#### **setproctitle\_enable**

If enabled, vsftpd will try and show session status information in the system process listing. In other words, the reported name of the process will change to reflect what a vsftpd session is doing (idle, downloading etc). You probably want to leave this off for security purposes.

Default: NO

#### **ssl\_enable**

If enabled, and vsftpd was compiled against OpenSSL, vsftpd will support secure connections via SSL. This applies to the control connection (including login) and also data connections. You'll need a client with SSL support too. NOTE!! Beware enabling this option. Only enable it if you need it. vsftpd can make no guarantees about the security of the OpenSSL libraries. By enabling this option, you are declaring that you trust the security of your installed OpenSSL library.

Default: NO

#### **ssl\_request\_cert**

If enabled, vsftpd will request (but not necessarily require; see **require\_cert**) a certificate on incoming SSL connections. Normally this should not cause any trouble at all, but IBM zOS seems to have issues. (New in v2.0.7).

Default: YES

#### **ssl\_sslv2**

Only applies if **ssl\_enable** is activated. If enabled, this option will permit SSL v2 protocol connections. TLS v1 connections are preferred.

Default: NO

#### **ssl\_sslv3**

Only applies if **ssl\_enable** is activated. If enabled, this option will permit SSL v3 protocol connections. TLS v1 connections are preferred.

Default: NO

#### **ssl\_tlsv1**

Only applies if **ssl\_enable** is activated. If enabled, this option will permit TLS v1 protocol connections. TLS v1 connections are preferred.

Default: YES

#### **strict\_ssl\_read\_eof**

If enabled, SSL data uploads are required to terminate via SSL, not an EOF on the socket. This option is required to be sure that an attacker did not terminate an upload prematurely with a faked TCP FIN. Unfortunately, it is not enabled by default because so few clients get it right. (New in v2.0.7).

Default: NO

**strict\_ssl\_write\_shutdown**

If enabled, SSL data downloads are required to terminate via SSL, not an EOF on the socket. This is off by default as I was unable to find a single FTP client that does this. It is minor. All it affects is our ability to tell whether the client confirmed full receipt of the file. Even without this option, the client is able to check the integrity of the download. (New in v2.0.7).

Default: NO

**syslog\_enable**

If enabled, then any log output which would have gone to /var/log/vsftpd.log goes to the system log instead. Logging is done under the FTPD facility.

Default: NO

**tcp\_wrappers**

If enabled, and vsftpd was compiled with tcp\_wrappers support, incoming connections will be fed through tcp\_wrappers access control. Furthermore, there is a mechanism for per-IP based configuration. If tcp\_wrappers sets the VSFTPD\_LOAD\_CONF environment variable, then the vsftpd session will try and load the vsftpd configuration file specified in this variable.

Default: NO

**text\_userdb\_names**

By default, numeric IDs are shown in the user and group fields of directory listings. You can get textual names by enabling this parameter. It is off by default for performance reasons.

Default: NO

**tilde\_user\_enable**

If enabled, vsftpd will try and resolve pathnames such as ~chris/pics, i.e. a tilde followed by a username. Note that vsftpd will always resolve the pathnames ~ and ~/something (in this case the ~ resolves to the initial login directory). Note that ~user paths will only resolve if the file /etc/passwd may be found within the \_current\_ chroot() jail.

Default: NO

**use\_localtime**

If enabled, vsftpd will display directory listings with the time in your local time zone. The default is to display GMT. The times returned by the MDTM FTP command are also affected by this option.

Default: NO

**use\_sendfile**

An internal setting used for testing the relative benefit of using the sendfile() system call on your platform.

Default: YES

**userlist\_deny**

This option is examined if **userlist\_enable** is activated. If you set this setting to NO, then users will be denied login unless they are explicitly listed in the file specified by **userlist\_file**. When login is denied, the denial is issued before the user is asked for a password.

Default: YES

**userlist\_enable**

If enabled, vsftpd will load a list of usernames, from the filename given by **userlist\_file**. If a user tries to log in using a name in this file, they will be denied before they are asked for a password. This may be useful in preventing cleartext passwords being transmitted. See also **userlist\_deny**.

Default: NO

**validate\_cert**

If set to yes, all SSL client certificates received must validate OK. Self-signed certs do not constitute OK validation. (New in v2.0.6).

Default: NO

**virtual\_use\_local\_privs**

If enabled, virtual users will use the same privileges as local users. By default, virtual users will use the same privileges as anonymous users, which tends to be more restrictive (especially in terms of write access).

Default: NO

#### **write\_enable**

This controls whether any FTP commands which change the filesystem are allowed or not. These commands are: STOR, DELE, RNFR, RNT0, MKD, RMD, APPE and SITE.

Default: NO

#### **xferlog\_enable**

If enabled, a log file will be maintained detailing uploads and downloads. By default, this file will be placed at /var/log/vsftpd.log, but this location may be overridden using the configuration setting **vsftpd\_log\_file**.

Default: NO (but the sample config file enables it)

#### **xferlog\_std\_format**

If enabled, the transfer log file will be written in standard xferlog format, as used by wu-ftp. This is useful because you can reuse existing transfer statistics generators. The default format is more readable, however. The default location for this style of log file is /var/log/xferlog, but you may change it with the setting **xferlog\_file**.

Default: NO

### **NUMERIC OPTIONS**

Below is a list of numeric options. A numeric option must be set to a non-negative integer. Octal numbers are supported, for convenience of the umask options. To specify an octal number, use 0 as the first digit of the number.

#### **accept\_timeout**

The timeout, in seconds, for a remote client to establish connection with a PASV style data connection.

Default: 60

#### **anon\_max\_rate**

The maximum data transfer rate permitted, in bytes per second, for anonymous clients.

Default: 0 (unlimited)

#### **anon\_umask**

The value that the umask for file creation is set to for anonymous users. NOTE! If you want to specify octal values, remember the "0" prefix otherwise the value will be treated as a base 10 integer!

Default: 077

#### **chown\_upload\_mode**

The file mode to force for chown()ed anonymous uploads. (Added in v2.0.6).

Default: 0600

#### **connect\_timeout**

The timeout, in seconds, for a remote client to respond to our PORT style data connection.

Default: 60

#### **data\_connection\_timeout**

The timeout, in seconds, which is roughly the maximum time we permit data transfers to stall for with no progress. If the timeout triggers, the remote client is kicked off.

Default: 300

#### **delay\_failed\_login**

The number of seconds to pause prior to reporting a failed login.

Default: 1

#### **delay\_successful\_login**

The number of seconds to pause prior to allowing a successful



login.

Default: 0

**file\_open\_mode**

The permissions with which uploaded files are created. Umask are applied on top of this value. You may wish to change to 0777 if you want uploaded files to be executable.

Default: 0666

**ftp\_data\_port**

The port from which PORT style connections originate (as long as the poorly named **connect\_from\_port\_20** is enabled).

Default: 20

**idle\_session\_timeout**

The timeout, in seconds, which is the maximum time a remote client may spend between FTP commands. If the timeout triggers, the remote client is kicked off.

Default: 300

**listen\_port**

If vsftpd is in standalone mode, this is the port it will listen on for incoming FTP connections.

Default: 21

**local\_max\_rate**

The maximum data transfer rate permitted, in bytes per second, for local authenticated users.

Default: 0 (unlimited)

**local\_umask**

The value that the umask for file creation is set to for local users. NOTE! If you want to specify octal values, remember the "0" prefix otherwise the value will be treated as a base 10 integer!

Default: 077

**max\_clients**

If vsftpd is in standalone mode, this is the maximum number of clients which may be connected. Any additional clients connecting will get an error message.

Default: 0 (unlimited)

**max\_login\_fails**

After this many login failures, the session is killed.

Default: 3

**max\_per\_ip**

If vsftpd is in standalone mode, this is the maximum number of clients which may be connected from the same source internet address. A client will get an error message if they go over this limit.

Default: 0 (unlimited)

**pasv\_max\_port**

The maximum port to allocate for PASV style data connections. Can be used to specify a narrow port range to assist firewalling.

Default: 0 (use any port)

**pasv\_min\_port**

The minimum port to allocate for PASV style data connections. Can be used to specify a narrow port range to assist firewalling.

Default: 0 (use any port)

**trans\_chunk\_size**

You probably don't want to change this, but try setting it to something like 8192 for a much smoother bandwidth limiter.

Default: 0 (let vsftpd pick a sensible setting)

**STRING OPTIONS**

Below is a list of string options.

#### **anon\_root**

This option represents a directory which vsftpd will try to change into after an anonymous login. Failure is silently ignored.

Default: (none)

#### **banned\_email\_file**

This option is the name of a file containing a list of anonymous e-mail passwords which are not permitted. This file is consulted if the option **deny\_email\_enable** is enabled.

Default: /etc/vsftpd.banned\_emails

#### **banner\_file**

This option is the name of a file containing text to display when someone connects to the server. If set, it overrides the banner string provided by the **ftpd\_banner** option.

Default: (none)

#### **ca\_certs\_file**

This option is the name of a file to load Certificate Authority certs from, for the purpose of validating client certs. The loaded certs are also advertised to the client, to cater for TLSv1.0 clients such as the z/OS FTP client. Regrettably, the default SSL CA cert paths are not used, because of vsftpd's use of restricted filesystem spaces (chroot). (Added in v2.0.6).

Default: (none)

#### **chown\_username**

This is the name of the user who is given ownership of anonymously uploaded files. This option is only relevant if another option, **chown\_uploads**, is set.

Default: root

#### **chroot\_list\_file**

The option is the name of a file containing a list of local users which will be placed in a `chroot()` jail in their home directory. This option is only relevant if the option **chroot\_list\_enable** is enabled. If the option **chroot\_local\_user** is enabled, then the list file becomes a list of users to NOT place in a `chroot()` jail.

Default: /etc/vsftpd.chroot\_list

#### **cmds\_allowed**

This options specifies a comma separated list of allowed FTP commands (post login. USER, PASS and QUIT and others are always allowed pre-login). Other commands are rejected. This is a powerful method of really locking down an FTP server. Example: `cmds_allowed=PASV,RETR,QUIT`

Default: (none)

#### **cmds\_denied**

This options specifies a comma separated list of denied FTP commands (post login. USER, PASS, QUIT and others are always allowed pre-login). If a command appears on both this and **cmds\_allowed** then the denial takes precedence. (Added in v2.1.0).

Default: (none)

#### **deny\_file**

This option can be used to set a pattern for filenames (and directory names etc.) which should not be accessible in any way. The affected items are not hidden, but any attempt to do anything to them (download, change into directory, affect something within directory etc.) will be denied. This option is very simple, and should not be used for serious access control - the filesystem's permissions should be used in preference. However, this option may be useful in certain virtual user setups. In particular aware that if a filename is accessible by a variety of names (perhaps due to symbolic links or hard links), then care must be taken to deny access to all the names. Access will be denied to items if their name contains the string given by `hide_file`, or if they match the regular expression specified by `hide_file`. Note that vsftpd's regular expression matching code is a simple implementation which is a subset of full regular expression functionality. Because of this, you will

need to carefully and exhaustively test any application of this option. And you are recommended to use filesystem permissions for any important security policies due to their greater reliability. Supported regex syntax is any number of \*, ? and unnested {,} operators. Regex matching is only supported on the last component of a path, e.g. a/b/? is supported but a/?/c is not. Example: deny\_file={\*.mp3,\*.mov,.private}

Default: (none)

#### **dsa\_cert\_file**

This option specifies the location of the DSA certificate to use for SSL encrypted connections.

Default: (none - an RSA certificate suffices)

#### **dsa\_private\_key\_file**

This option specifies the location of the DSA private key to use for SSL encrypted connections. If this option is not set, the private key is expected to be in the same file as the certificate.

Default: (none)

#### **email\_password\_file**

This option can be used to provide an alternate file for usage by the **secure\_email\_list\_enable** setting.

Default: /etc/vsftpd.email\_passwords

#### **ftp\_username**

This is the name of the user we use for handling anonymous FTP. The home directory of this user is the root of the anonymous FTP area.

Default: ftp

#### **ftpd\_banner**

This string option allows you to override the greeting banner displayed by vsftpd when a connection first comes in.

Default: (none - default vsftpd banner is displayed)

#### **guest\_username**

See the boolean setting **guest\_enable** for a description of what constitutes a guest login. This setting is the real username which guest users are mapped to.

Default: ftp

#### **hide\_file**

This option can be used to set a pattern for filenames (and directory names etc.) which should be hidden from directory listings. Despite being hidden, the files / directories etc. are fully accessible to clients who know what names to actually use. Items will be hidden if their names contain the string given by hide\_file, or if they match the regular expression specified by hide\_file. Note that vsftpd's regular expression matching code is a simple implementation which is a subset of full regular expression functionality. See **deny\_file** for details of exactly what regex syntax is supported. Example:  
hide\_file={\*.mp3,.hidden,hide\*,h?}

Default: (none)

#### **listen\_address**

If vsftpd is in standalone mode, the default listen address (of all local interfaces) may be overridden by this setting. Provide a numeric IP address.

Default: (none)

#### **listen\_address6**

Like listen\_address, but specifies a default listen address for the IPv6 listener (which is used if listen\_ipv6 is set). Format is standard IPv6 address format.

Default: (none)

#### **local\_root**

This option represents a directory which vsftpd will try to change into after a local (i.e. non-anonymous) login. Failure is silently ignored.

Default: (none)

**message\_file**

This option is the name of the file we look for when a new directory is entered. The contents are displayed to the remote user. This option is only relevant if the option **dirmessage\_enable** is enabled.

Default: .message

**nopriv\_user**

This is the name of the user that is used by vsftpd when it wants to be totally unprivileged. Note that this should be a dedicated user, rather than nobody. The user nobody tends to be used for rather a lot of important things on most machines.

Default: nobody

**pam\_service\_name**

This string is the name of the PAM service vsftpd will use.

Default: vsftpd

**pasv\_address**

Use this option to override the IP address that vsftpd will advertise in response to the PASV command. Provide a numeric IP address, unless **pasv\_addr\_resolve** is enabled, in which case you can provide a hostname which will be DNS resolved for you at startup.

Default: (none - the address is taken from the incoming connected socket)

**rsa\_cert\_file**

This option specifies the location of the RSA certificate to use for SSL encrypted connections.

Default: /usr/share/ssl/certs/vsftpd.pem

**rsa\_private\_key\_file**

This option specifies the location of the RSA private key to use for SSL encrypted connections. If this option is not set, the private key is expected to be in the same file as the certificate.

Default: (none)

**secure\_chroot\_dir**

This option should be the name of a directory which is empty. Also, the directory should not be writable by the ftp user. This directory is used as a secure chroot() jail at times vsftpd does not require filesystem access.

Default: /var/run/vsftpd/empty

**ssl\_ciphers**

This option can be used to select which SSL ciphers vsftpd will allow for encrypted SSL connections. See the **ciphers** man page for further details. Note that restricting ciphers can be a useful security precaution as it prevents malicious remote parties forcing a cipher which they have found problems with.

Default: DES-CBC3-SHA

**user\_config\_dir**

This powerful option allows the override of any config option specified in the manual page, on a per-user basis. Usage is simple, and is best illustrated with an example. If you set **user\_config\_dir** to be **/etc/vsftpd\_user\_conf** and then log on as the user "chris", then vsftpd will apply the settings in the file **/etc/vsftpd\_user\_conf/chris** for the duration of the session. The format of this file is as detailed in this manual page! PLEASE NOTE that not all settings are effective on a per-user basis. For example, many settings only prior to the user's session being started. Examples of settings which will not affect any behaviour on a per-user basis include **listen\_address**, **banner\_file**, **max\_per\_ip**, **max\_clients**, **xferlog\_file**, etc.

Default: (none)

**user\_sub\_token**

This option is useful in conjunction with virtual users. It is used to 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 fred logs in, he will end up (usually)

when virtual user fred logs in, he will end up (usually chroot()'ed) in the directory **/home/virtual/fred**. This option also takes affect if **local\_root** contains **user\_sub\_token**.

Default: (none)

#### **userlist\_file**

This option is the name of the file loaded when the **userlist\_enable** option is active.

Default: /etc/vsftpd.user\_list

#### **vsftpd\_log\_file**

This option is the name of the file to which we write the vsftpd style log file. This log is only written if the option **xferlog\_enable** is set, and **xferlog\_std\_format** is NOT set. Alternatively, it is written if you have set the option **dual\_log\_enable**. One further complication - if you have set **syslog\_enable**, then this file is not written and output is sent to the system log instead.

Default: /var/log/vsftpd.log

#### **xferlog\_file**

This option is the name of the file to which we write the wu-ftp style transfer log. The transfer log is only written if the option **xferlog\_enable** is set, along with **xferlog\_std\_format**. Alternatively, it is written if you have set the option **dual\_log\_enable**.

Default: /var/log/xferlog

#### **AUTHOR**

[scarybeasts@gmail.com](mailto:scarybeasts@gmail.com)

VSFTPD.CONF(5)