

# Install and Configure PureFTPd Ubuntu 18.04

Install the PureFTPd :

```
sudo apt-get install pure-ftpd -y
```

Edit **/etc/inetd.conf** file and comment out (add a # at the start of) the line containing **ftp** if such an entry exists:

```
sudo vi /etc/inetd.conf
```

Edit **/etc/default/pure-ftpd-common** and verify the **STANDALONE\_OR\_INETD=standalone** entry is set:

```
sudo vi /etc/default/pure-ftpd-common
```

Add a "ftpgroup" in the system:

```
sudo groupadd ftpgroup
```

Add a "ftpuser" user in the system:

```
sudo useradd -g ftpgroup -d /dev/null -s /etc ftpuser
```

Add a virtual PureFTPd user. I'm going to use "joe" as an example:

```
sudo pure-pw useradd joe -u ftpuser -d /name/of/directory
```

where **/name/of/directory** is the directory where you want user **joe** to have FTP access. This directory is where user **joe** is going to be locked in once they log on the server with FTP. Whether you create a directory for **joe** to have access or you use an existing directory, ensure the user/group **ftpuser/ftpgroup** you created earlier is the owner of that directory as follows:

```
chown -R ftpuser:ftpgroup /name/of/directory
```

Now, create the PureFTPd virtual user database:

```
sudo pure-pw mkdb
```

Create the following symbolic links for PureFTPd to function properly:

```
sudo ln -s /etc/pure-ftpd/pureftpd.passwd /etc/pureftpd.passwd
sudo ln -s /etc/pure-ftpd/pureftpd.pdb /etc/pureftpd.pdb
sudo ln -s /etc/pure-ftpd/conf/PureDB /etc/pure-ftpd/auth/PureDB
```

Ensure that the file **/etc/pure-ftpd/conf/UnixAuthentication** file only contains the word **no**:

```
sudo vi /etc/pure-ftpd/conf/UnixAuthentication
```

Restart PureFTPd before changes take effect:

```
sudo /etc/init.d/pure-ftpd restart
```

## Configure PureFTPd Options

PureFTPd on Ubuntu/Debian distros use the pure-ftpd-wrapper which will parse any properly named file in the "/etc/pure-ftpd/conf" directory and read the values and in turn pass to the pure-ftpd daemon. This eliminates the need editing long configuration files. There are a lot of files that can be placed in the "/etc/pure-ftpd/conf" directory for different configuration options, but I'm only going to concentrate on a handful. For a complete list of all the files refer to the following

<http://manpages.ubuntu.com/manpages/har ... per.8.html> link.

### Passive Mode Port Range

Passive mode can be enabled by simply issuing the following from the command line for setting a range of 30000 through 31000:

```
echo 30000 31000 > /etc/pure-ftpd/conf/PassivePortRange
```

### Bind to specific address and port number

If you wish to set PureFTPd to listen to a specific port number, issue the following from the command line. In this example we set port number "666" as the FTP port:

```
echo 192.168.xxx.xxx,666 > /etc/pure-ftpd/conf/Bind
```

### Disable name resolution in PureFTPd

I highly recommend you set this option in PureFTPd. This will disable the server trying to resolve the client's hostname. If it's not set, the server will sometimes throw a 425 Invalid Address given errors. Setting this option will fix those errors as well as speed up logins.

```
echo 'yes' > /etc/pure-ftpd/conf/DontResolve
```

## Set passive IP in PureFTPd

If you are behind a NAT, it's recommended you set the public IP address of your PureFTPd server as follows:

```
echo '1.2.3.4' > /etc/pure-ftpd/conf/ForcePassiveIP
```

## Enable TLS on PureFTPd

The FTP protocol in general is very insecure. The username/passwords are sent using clear text and the data transfers are also insecure. Enabling TLS will allow you to secure your FTP sessions to include the username/passwords as well as the data transfers.

Install OpenSSL:

```
sudo apt-get install openssl -y
```

If you want to accept plain AND TLS sessions, issue the following on the command line:

```
echo 1 > /etc/pure-ftpd/conf/TLS
```

If you want to accept TLS sessions ONLY, issue the following on the command line:

```
echo 2 > /etc/pure-ftpd/conf/TLS
```

## Create the SSL certificate for TLS

Create a "private" directory under "/etc/ssl/" if one doesn't exist yet:

```
mkdir /etc/ssl/private
```

Generate a self-signed certificate as follows:

```
openssl req -x509 -nodes -days 7300 -newkey rsa:2048 -keyout /etc/ssl/private/pure-ftpd.pem -out  
/etc/ssl/private/pure-ftpd.pem
```

Fill in the certificate information as required.

For 3rd party SSL certificates, enter the private key and corresponding chain certs in the following order inside `/etc/ssl/private/pure-ftpd.pem`:

-----BEGIN RSA PRIVATE KEY-----

(Private Key)

-----END RSA PRIVATE KEY-----

-----BEGIN CERTIFICATE-----

(Primary SSL certificate)

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

(Intermediate certificate)

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

(Root certificate)

-----END CERTIFICATE-----

## Troubleshooting

You may see the following warning when trying to connect to your PureFTPd server:

```
[WARNING] Can't login as [joe]: account disabled
```

```
"Sorry, but I can't trust you"
```

```
[WARNING] Can't login as [joe]: account disabled (uid < 1021)
```

These two warnings occur if your system set the UserID (UID) and/or GroupID (GID) associated with the `ftpuser` user are below 1000. To see what the current values are, type the following at a shell:

```
id ftpuser
```

Should output similar to below:

```
uid=572(ftpuser) gid=972(ftpgroup) groups=972(ftpgroup)
```

The actual numbers don't matter much, but they should be equal or higher than 1000 for PureFTPd to be happy. To fix the UserID (UID) portion, open a shell and type:

```
sudo usermod -u 1021 -p -U ftpuser
```

To fix the GroupID (GID):

```
sudo groupmod -g 1021 ftpgroup
```

Additionally, you can set the MinUID that PureFTPd expects by setting the following:

```
echo 1021 > /etc/pure-ftpd/conf/MinUID
```

Ensure to restart the Pure-FTPd daemon:

```
systemctl restart pure-ftpd
```

## Manage PureFTPd Users

The commands below are for performing common tasks with the PureFTPd user database. This assumes that **username** is the PureFTPd virtual user you are managing, **ftpuser** is the system user you are associating the virtual user with and **/name/of/directory** is the directory you want that virtual user to have access.

Remember that after every change in the PureFTPd database, you **MUST** commit the changes by typing **sudo pure-pw mkdb** and always make sure that **ftpuser/ftpgroup** are the owners of whatever directory you want that user to have access:

### Add Users:

```
sudo pure-pw useradd username -u ftpuser -d /name/of/directory
```

### Change User Password:

```
sudo pure-pw passwd username
```

## Show User Details:

```
sudo pure-pw show username
```

## Delete user:

```
sudo pure-pw userdel username
```

## Update PureFTPd Virtual User Database:

```
sudo pure-pw mkdb
```

---

Revision #4

Created 9 April 2021 09:01:40 by Dino Edwards

Updated 6 December 2021 19:10:26 by Dino Edwards