FTP CLIENT SIDE

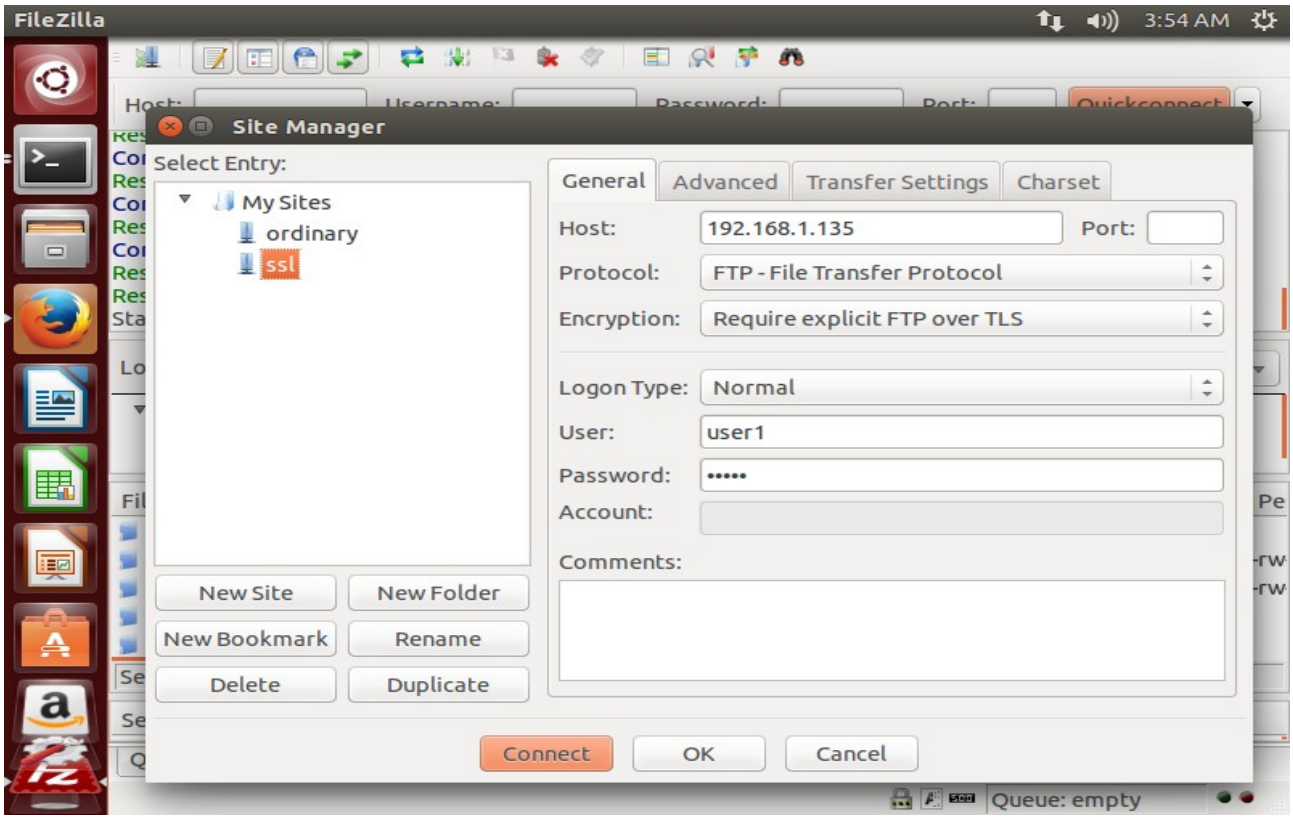

Fig : config in filezilla client side

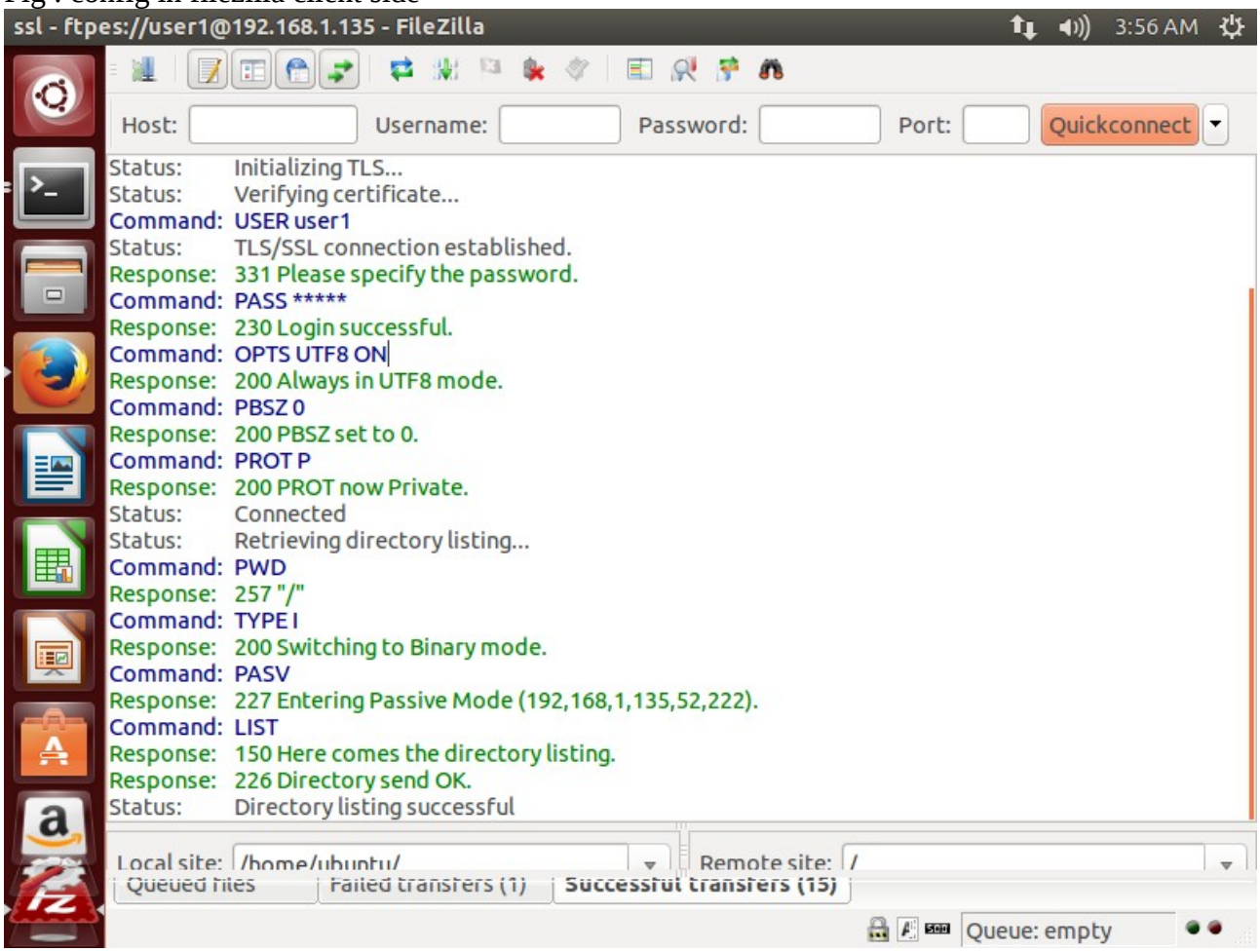

fig : output of config in filezilla

FTP over TLS (Explicit)

Explicit security requires that the FTP client issues a specific command to the FTP server after establishing a connection to establish the SSL link. In explicit TLS the FTP client needs to send an explicit command (i.e. "AUTH TLS") to the FTP server to initiate a secure control connection. The default FTP server port is used. This formal method is documented in RFC 4217.

FTP over TLS (Implicit) - Deprecated

Implicit security is a mechanism by which security is automatically turned on as soon as the FTP client makes a connection to an FTP server. In this case, the FTP server defines a specific port for the client (990) to be used for secure connections. **FTP over TLS Implicit has been deprecated** and should no longer be used.

How To make a secure connection

Protocol Summary

FTP	No secure connection.
FTPS (Explicit)	Explicit security requires that the FTP client issues a specific command to the FTP server after establishing a connection to establish the SSL link. The default FTP server port is used. This formal method is documented in RFC 2228. Wikipedia: FTPS
FTPS (Implicit)	Implicit security is a mechanism by which security is automatically turned on as soon as the FTP client makes a connection to an FTP server. In this case, the FTP server defines a specific port for the client (990) to be used for secure connections. Wikipedia: FTPS **Note**: Implicit is deprecated.
SFTP over SSH	Wikipedia: SSH file transfer protocol

SSL (Secure Sockets Layer) is a protocol for transmitting data between a client and a server via the Internet that uses a cryptographic system designed to prevent eavesdropping, tampering and message forgery that might occur during a regular FTP transfer.

How To select the Protocol

To create a secure connection you have the following options:

Quick Connect

1. In the menu select **File - New Connection**
2. Select the protocol from the **Protocol** selection

From a Favorite

1. Select **Favorites - Edit Favorites** from the menu. The **Favorites Browser** appears.
2. From the Favorites Browser window select the Favorite to be used. Right-click on the Favorite and select **Properties** from the context menu. The Properties dialog appears.
3. Select **General**. In the the **General** dialog change the **Protocol.**

Mostly explicit ssl connection used.

Open a secured connection (FTPS) to the server:

1. menu: **File - Connnection**
2. Select **FTPS (Explicit)** for the Protocol
3. Enter the username, password and FTP host name
4. Click **OK** to connect

active and passive mode inftp

Active and passive are the two modes that FTP can run in. FTP uses two channels between client and server, the command channel and the data channel, which are actually separate TCP connections. The command channel is for commands and responses, the data channel is for actually transferring files. It's a nifty way of sending commands to the server without having to wait for the current data transfer to finish.

In active mode, the client establishes the command channel (from client port `X` to server port `21`$^{(b)}$) but the server establishes the data channel (from server port `20`$^{(b)}$ to client port `Y`, where `Y` has been supplied by the client).

In passive mode, the client establishes both channels. In that case, the server tells the client which port should be used for the data channel.

Active mode:

- Client opens up command channel from client port 2000$^{(a)}$ to server port 21$^{(b)}$.
- Client sends `PORT 2001`$^{(a)}$ to server and server acknowledges on command channel.
- Server opens up data channel from server port 20$^{(b)}$ to client port 2001$^{(a)}$.
- Client acknowledges on data channel.

Passive mode:

- Client opens up command channel from client port 2000$^{(a)}$ to server port 21$^{(b)}$.
- Client sends `PASV` to server on command channel.
- Server sends back (on command channel) `PORT 1234`$^{(a)}$ after starting to listen on that port.
- Client opens up data channel from client 2001$^{(a)}$ to server port 1234$^{(a)}$.

- Server acknowledges on data channel.

At this point, the command and data channels are both open.

---

[a]Note that the selection of ports on the client side is up to the client, as the selection of the server data channel port in passive mode is up to the server.

[b]Further note that the use of port 20 and 21 is only a convention (although a strong one). There's no absolute requirement that those ports be used although the client and server both have to agree on which ports are being used. I've seen implementations that try to hide from clients by using different ports (futile, in my opinion).