

FTP – The File Transfer Protocol

Introduction

One of the primary activities found on the Internet is the transferring of files. Every minute of every day, Internet users download files from various websites and Webmasters upload files and web pages to their website. The most common method for transferring files over the Internet is via the *File Transfer Protocol*, or FTP.

Overview

FTP is a widely accepted Internet Standard. Originally introduced in the early 70's, it was officially approved by the IETF¹ in the mid 80's and assigned RFC²-959. Although there have been subsequent Internet Drafts over the years that have added extensions to the original specification, RFC-959 is the rulebook that defines FTP.

FTP falls into the standard client/server model. To use FTP, there needs to exist both an FTP client program and an FTP server program. The FTP server will store or house the files accessed during file transfer and the FTP client will connect to the FTP server and send files to, or retrieve files from the server.

Typical Conversation Structure

FTP uses a basic command/reply mechanism. The FTP client will connect to the FTP server, usually on port 21³, and will begin a synchronized conversation that involves the client sending a command to the server and the server sending a response to the client. For each command sent by the client, the server will issue a response. Responses from the server are of a standardized format where the first three characters of the response will be a 3-digit response code. The first digit of the response code is the most important as it is an indicator of the overall success or failure status of the command. Usually response codes beginning with 1, 2, or 3 are good, and response codes beginning with 4 or 5 are not good. For example, if the client were to issue a "CWD⁴ /incoming/" command to change the current directory to /incoming/, the server could reply with either a "200 Success" indicating that the command succeeded or "550 Access Denied" indicating that the client does not have adequate rights to access the specified directory.

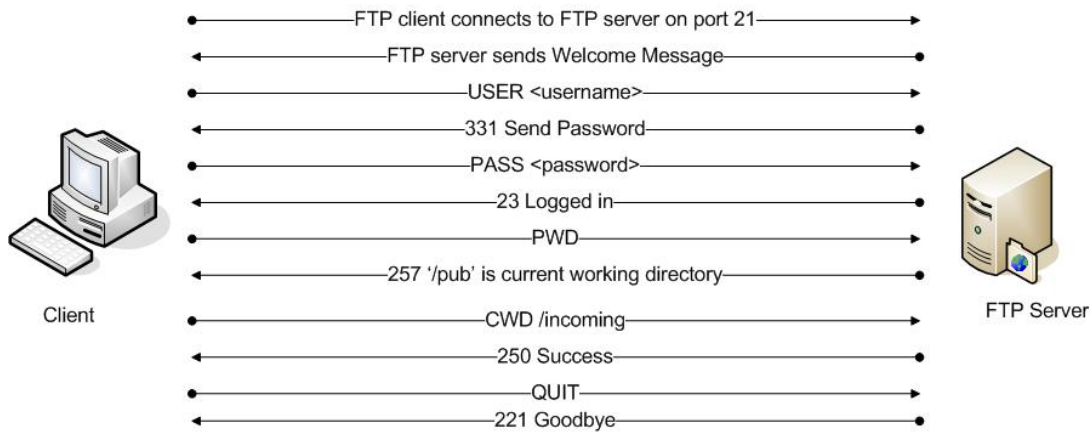
¹ Internet Engineering Task Force. The IETF is a large open international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet.

² Request For Comments. A set of technical and organizational notes about the Internet. Official specifications defined by the IETF are recorded and published as RFCs.

³ Port 21 and port 20 are the industry standard ports reserved for FTP traffic. The default port for an FTP *Control Connection* is port 21 and the default port for the a *Data Connection* is port 20.

⁴ Change Working Directory. The CWD command is used to change the current working directory on the remote FTP server.

Figure 1 illustrates the *Format of a Typical FTP Conversation* between a client and a server.

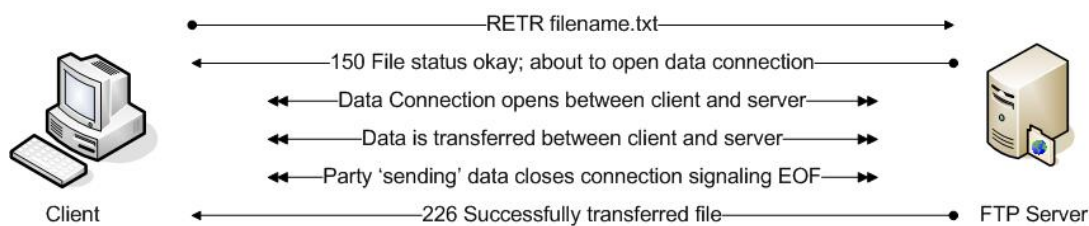


---Figure 1---

Control Connections and Data Connections

FTP will use two separate connections during a typical session where files are transferred. The *Control Connection* is the primary connection and is used to send commands back and forth between the client and server. Each time the client and server wish to transfer a file or other data, such as a directory listing, a separate connection is established between the parties. This separate connection is called the *Data Connection* and is used solely for the purpose of transferring the requested data. Once the data has been transferred between the two parties, the *Data Connection* is closed.

Figure 2 illustrates a typical *File Transfer Session* between an FTP client and server.



--- Figure 2 ---

The illustration assumes that the client has already successfully established a conversation with the server, and that the client has authenticated itself with the server. The client wishes to download/retrieve a file named *filename.txt* from the server. To initiate the file transfer, the client issues the RETR (retrieve) command followed by the name of the file to be retrieved. If the file exists and if the client has adequate rights to access the file, the server will issue a reply indicating that everything is OK and that the file transfer will now begin. At this point, the two parties will go through the process of establishing a *Data Connection* to be used to transfer the

data (The details of exactly how this is done will be discussed later). Once the *Data Connection* is established, the server sends the data to the client. After all data has been sent, the party *sending* the data will gracefully close the *Data Connection*.⁵ The last step of the process involves the FTP server sending a response code back to the FTP client indicating success or failure.

Data Connection Details

When a client and server intend to transfer data, they usually⁶ negotiate the details of the *Data Connection* prior to opening. RFC-959 defines a mechanism by which the details do not need to be negotiated; however, it's very rare to find an FTP client that relies on the default values. Nearly all FTP clients will explicitly specify the IP address and port for the *Data Connection*.

The first step to establishing a *Data Connection* involves choosing an IP address and port number to be used for the *Data Connection*.⁷ The FTP client will issue either the PORT command or the PASV⁸ command to the server. These commands are used to signal the selection of an IP address and port number to be used for the transfer. The PORT command is used when an *Active Mode Data Connection* is being negotiated and the PASV command is used when a *Passive Mode Data Connection* is being negotiated. (The difference between *Active* and *Passive Mode* is defined in the next section.) If the FTP client decides to choose the IP address and port number to be used for the *Data Connection*, it will issue the PORT command to the FTP server. The PORT command will contain the IP address and port number chosen by the client that should be used for data transfer. If the FTP client decides that the server should choose the IP address and port number, it will issue the PASV command to the server. When the server receives the PASV command, it will select an IP address and port number which is appropriate to use for the *Data Connection* and will return that information to the client.

Once the IP address and port number have been selected, the party that chose the IP address and port will begin to *listen* on that address/port and wait for the other party to connect⁹. When the other party connects to the listening party, the data transfer begins.

After the data has been transferred, the party that has sent the data will gracefully close the *Data Connection*, signaling EOF.

Passive Mode vs. Active Mode

During the address/port negotiation phase of a standard FTP session, the client and server negotiate the IP address and port number to be used to transfer the data. To do this, the client will issue either the PORT command (when in *Active Mode*) or the PASV command (when in *Passive Mode*). When running in *Active Mode*, the client issues a PORT command to the server signaling the server to *Actively* open the *Data Connection* back to the client. When running in *Passive*

⁵ *Data Connections* are closed by the party that initiated the data transfer. When sending data from a client to a server, the client will close the connection once all data has been sent. When a client is retrieving data from a server, the server will close the connection once all data has been transferred. The only exception to this would occur if there was an unexpected error during sending or receiving.

⁶ The term *usually* is used here because RFC-959 defines a mechanism by which the IP address and port number do not need to be negotiated beforehand. By default, the FTP server will attempt to open a connection from port 20 on the server back to the same client IP address that is being used for the *Control Connection*.

⁷ The process of *choosing* an IP address and port involves selecting an unused port, usually a port > 1024 and < 65535, and combining that with the same IP address that was used for the *Control Connection*. It is generally not desirable to select a port below 1024 unless you are choosing port 20 since these ports are reserved for system services.

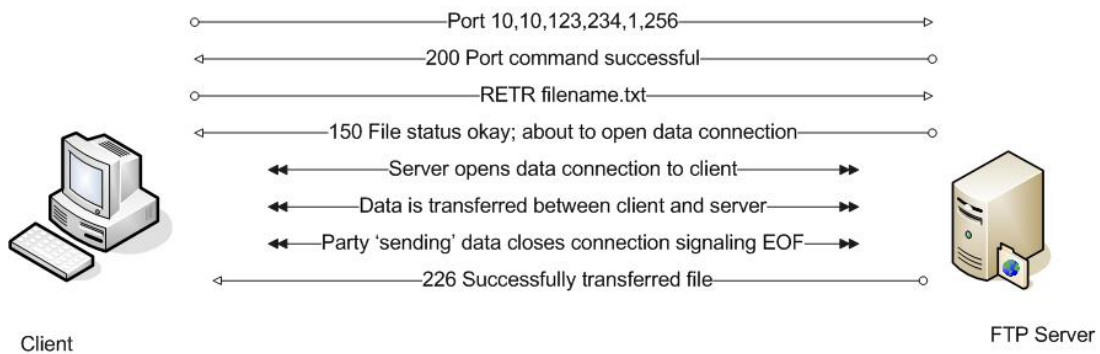
⁸ PASV is short for Passive, indicating that the server should select an IP address and port number, then enter into a passive wait state until either the client connects, or a predetermined timeout value has expired.

⁹ The listening party will usually have a pre-determined timeout period for which it will wait for the other party to connect. If the other party does not establish a connection in the given time period, the listening party will stop listening. If the *Data Connection* fails to establish, the server will return a 400 level error to the client.

Mode, the client issues a PASV command to the server signaling the server to select an appropriate IP address and port number, and then listen *passively* while the client attempts to open the data connection.

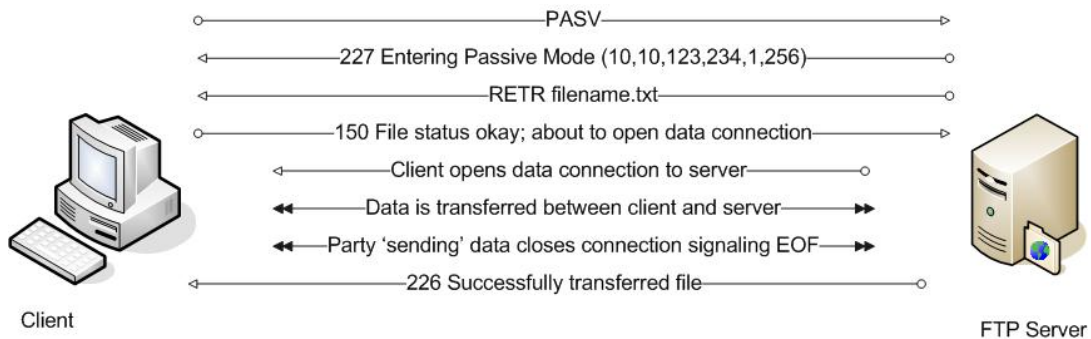
One of the primary reasons for having the choice of *Active* or *Passive Mode Data Connections* is due to any firewalls that may be present between the client and server. A common scenario would be an FTP server that sits behind a firewall. The firewall would be blocking all incoming traffic except for traffic bound for port 21 (FTP traffic). If the FTP client were to issue the PASV command to the FTP server, the server would respond with an IP address and port that the client should use to connect back to the server. However, since the firewall is blocking access to *all* ports except for Port 21, the FTP client will not be able to connect to the server on the Port that the server chose. To correct this problem, the FTP client would need to issue the PORT command to the server. Since the client is now the passive entity in the connection establishment, the server would be able to open an outbound connection through the firewall to the FTP client.

Figure 3 illustrates a typical *Active Mode Data Connection*.



--- Figure 3 ---

Figure 4 illustrates a typical *Passive Mode Data Connection*.



--- Figure 4 ---

Conclusion

The longevity of FTP ensures its strong foothold and widespread acceptance in the Internet community. There are numerous FTP clients and servers available in the market today, nearly all of which support the features of RFC-959 (plus a few extras). Nearly all ISP's and broadband providers supply FTP features which allow customers to upload their web pages to their website. An FTP server is the preferred repository for storing software patches and drivers for many hardware vendors. LinkSys and Dell for example, have FTP servers that house patches and drivers for their products where customers are encouraged to download updates when needed.

Even in the security conscious era, FTP is managing to adapt. Recent revisions to the original RFC-959 specification have added support for security extensions that allow FTP traffic to be secure. There are now many servers that support FTPS¹⁰. Many companies are finding it more cost effective to upgrade their FTP server to an FTPS server rather than replace it with a more expensive VPN¹¹ or SSH¹² server.

About South River Technologies

South River Technologies, developers of Internet File Management and Collaboration Solutions, has full FTP support in its products. [WebDrive](#), the world's first drive mapping client based on Internet protocols, supports both FTP and FTPS. Also, [Titan FTP server](#), SRT's enterprise class secure server product, fully supports both FTP and FTPS. For more information about South River Technologies and our products, please visit us on the web at <http://www.southrivertech.com>.

The information contained in this document represents the current view of South River Technologies on the issues discussed, as of the date of publication. This White Paper is for information purposes only. SRT makes no warranties, express or implied, as to the information in this document. This White Paper may not be reproduced in any form without written permission from SRT.

Titan FTP Server and WebDrive are trademarks of South River Technologies, Inc. All other trademarks are the property of their respective owners.

¹⁰ FTPS is short for FTP over SSL. FTP over SSL (Secure Sockets Layer) is outlined in RFC-2228 and includes mechanisms for using SSL v3.0 and SSL v3.1 (which is also known as TLS v1.0).

¹¹ VPN – Virtual Private Network. Software or hardware used between a client and server that generates a secure tunnel for all traffic, not just FTP traffic.

¹² SSH – Secure Shell. Similar to SSL; offered as a secure replacement to FTP and FTPS.