FILE TRANSFER PROTOCOL

INTRODUCTION

The objectives of FTP are 1) to promote sharing of files (computer
programs and/or data), 2) to encourage indirect or implicit (via
programs) use of remote computers, 3) to shield a user from
variations in file storage systems among Hosts, and 4) to transfer
data reliably and efficiently. FTP, though usable directly by a user
at a terminal, is designed mainly for use by programs.

The attempt in this specification is to satisfy the diverse needs of
users of maxi-Hosts, mini-Hosts, and TIPs, with a simple, and easily
implemented protocol design.

This paper assumes knowledge of the following protocols described in
the ARPA Internet Protocol Handbook.

The Transmission Control Protocol

The TELNET Protocol

DISCUSSION

In this section, the terminology and the FTP model are discussed.
The terms defined in this section are only those that have special
significance in FTP. Some of the terminology is very specific to the
FTP model; some readers may wish to turn to the section on the FTP
model while reviewing the terminology.

TERMINOLOGY

ASCII

The ASCII character set as defined in the ARPA Internet
Protocol Handbook. In FTP, ASCII characters are defined to be
the lower half of an eight-bit code set (i.e., the most
significant bit is zero).

access controls

Access controls define users' access privileges to the use of a
system, and to the files in that system. Access controls are
necessary to prevent unauthorized or accidental use of files.
It is the prerogative of a server-FTP process to invoke access
controls.

byte size

>   There are two byte sizes of interest in FTP: the logical byte
>   size of the file, and the transfer byte size used for the
>   transmission of the data. The transfer byte size is always 8
>   bits. The transfer byte size is not necessarily the byte size
>   in which data is to be stored in a system, nor the logical byte
>   size for interpretation of the structure of the data.

data connection

>   A simplex connection over which data is transferred, in a
>   specified mode and type. The data transferred may be a part of
>   a file, an entire file or a number of files. The path may be
>   between a server-DTP and a user-DTP, or between two
>   server-DTPs.

data port

>   The passive data transfer process "listens" on the data port
>   for a connection from the active transfer process in order to
>   open the data connection.

EOF

>   The end-of-file condition that defines the end of a file being
>   transferred.

EOR

>   The end-of-record condition that defines the end of a record
>   being transferred.

error recovery

>   A procedure that allows a user to recover from certain errors
>   such as failure of either Host system or transfer process. In
>   FTP, error recovery may involve restarting a file transfer at a
>   given checkpoint.

FTP commands

>   A set of commands that comprise the control information flowing
>   from the user-FTP to the server-FTP process.

2

file

   An ordered set of computer data (including programs), of
   arbitrary length, uniquely identified by a pathname.

mode

   The mode in which data is to be transferred via the data
   connection. The mode defines the data format during transfer
   including EOR and EOF. The transfer modes defined in FTP are
   described in the Section on Transmission Modes.

NVT

   The Network Virtual Terminal as defined in the TELNET Protocol.

NVFS

   The Network Virtual File System. A concept which defines a
   standard network file system with standard commands and
   pathname conventions. FTP only partially implements the NVFS
   concept at this time.

page

   A file may be structured as a set of independent parts called
   pages. FTP supports the transmission of discontinuous files as
   independent indexed pages.

pathname

   Pathname is defined to be the character string which must be
   input to a file system by a user in order to identify a file.
   Pathname normally contains device and/or directory names, and
   file name specification. FTP does not yet specify a standard
   pathname convention. Each user must follow the file naming
   conventions of the file systems involved in the transfer.

record

   A sequential file may be structured as a number of contiguous
   parts called records. Record structures are supported by FTP
   but a file need not have record structure.

reply

    A reply is an acknowledgment (positive or negative) sent from
    server to user via the TELNET connections in response to FTP
    commands. The general form of a reply is a completion code
    (including error codes) followed by a text string. The codes
    are for use by programs and the text is usually intended for
    human users.

server-DTP

    The data transfer process, in its normal "active" state,
    establishes the data connection with the "listening" data port,
    sets up parameters for transfer and storage, and transfers data
    on command from its PI. The DTP can be placed in a "passive"
    state to listen for, rather than initiate a, connection on the
    data port.

server-FTP process

    A process or set of processes which perform the function of
    file transfer in cooperation with a user-FTP process and,
    possibly, another server. The functions consist of a protocol
    interpreter (PI) and a data transfer process (DTP).

server-PI

    The protocol interpreter "listens" on Port L for a connection
    from a user-PI and establishes a TELNET communication
    connection. It receives standard FTP commands from the
    user-PI, sends replies, and governs the server-DTP.

TELNET connections

    The full-duplex communication path between a user-PI and a
    server-PI, operating according to the TELNET Protocol.

type

    The data representation type used for data transfer and
    storage. Type implies certain transformations between the time
    of data storage and data transfer. The representation types
    defined in FTP are described in the Section on Establishing
    Data Connections.

4

user

    A human being or a process on behalf of a human being wishing
    to obtain file transfer service. The human user may interact
    directly with a server-FTP process, but use of a user-FTP
    process is preferred since the protocol design is weighted
    towards automata.

user-DTP

    The data transfer process "listens" on the data port for a
    connection from a server-FTP process. If two servers are
    transferring data between them, the user-DTP is inactive.

user-FTP process

    A set of functions including a protocol interpreter, a data
    transfer process and a user interface which together perform
    the function of file transfer in cooperation with one or more
    server-FTP processes. The user interface allows a local
    language to be used in the command-reply dialogue with the
    user.

user-PI

    The protocol interpreter initiates the TELNET connection from
    its port U to the server-FTP process, initiates FTP commands,
    and governs the user-DTP if that process is part of the file
    transfer.

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE

fastcase®
Smarter legal research.