

Protocolo FTP

Redes y Comunicaciones

Historia de FTP

- FTP (File Transfer Protocol) conforma el grupo de los protocolos más viejos de la Internet aún utilizados,
- Propuesta original RFC-114, año 1971, MIT.
- Existió antes de TCP/IP, ejecutaba sobre NCP.
- El protocolo original ha sufrido varias, adaptado a IP, la esencia es la misma.
- Tuvo gran auge con a Internet comercial, en 1992 era el protocolo que más volumen transportaba (En la actualidad, ha sido superado por HTTP).
- FTP es un protocolo para copiar archivos completos, a diferencia de otros protocolos que brindan acceso a archivos: NFS, CIFS.

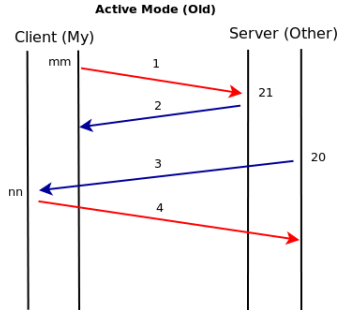
Características del Protocolo

- Estandarizado por RFC-765, luego convertido en obsoleto por RFC-959. Actualizado por RFC-2228 y RFC-3659.
- Mensajes se codifican en ASCII estándar (de 7 bits codificados en 8). (terminal ASCII NVT -Network Virtual Terminal-CRLF).
- Modelo cliente/servidor, command/response.
- Protocolo corre sobre TCP (requiere protocolo de transporte confiable).
- Los clientes FTP, no requieren interfaz gráfica.
- Soportado por los browsers/clientes Web mediante la URI: ftp://... Ejemplos de Clientes: WS_FTP de Ipswitch, ftp, gFTP, FileZilla.
- Ejemplos de Servidores: BSD ftpd(8) de Unix BSD, WU-FTP, Pure-FTPd, vsFTPD, FileZilla-Server.

Funcionamiento de FTP

- Usa 2 (dos) conexiones TCP:
 - Conexión de Control (Out-Of-Band Control) port 21.
 - Conexión para la transferencia de datos.
- Cada conexión requerimientos servicios diferentes:
 - Conexión de Control: min delay.
 - Conexión de Datos: max throughput.
- El cliente escoge cualquier puerto no privilegiado, ($n > 1023$) y genera conexión de control contra el puerto 21 del servidor.
- El servidor recibe los comandos por dicha conexión y responde/recibe por la conexión de datos aquellos que lo requieran.
- La conexión de datos se crea y de cierra bajo demanda.
- El estado de cada operación se transmite por el canal de control.

Esquema de FTP



Algunos Comandos FTP

RETR: obtener un archivo desde el servidor. A nivel de interfaz de usuario el comando que lo inicia es el `get`.

STOR: envía un archivo al servidor. A nivel de interfaz de usuario el comando que lo inicia es el `put`.

LIST: envía un petición de listar los archivos del directorio actual en el servidor. A nivel de interfaz de usuario el comando que lo inicia es el `ls` o `dir`.

DELE: comando para borrar un archivo en el servidor.

SIZE, STAT: obtiene información de un archivo en el servidor.

CD, PWD, RMD, MKD: cambia de directorio, obtiene el dir. actual, borra y crea dir.

Ejemplos de respuestas FTP

**** Positive preliminary reply:**

150 Here comes the directory listing.

**** Positive completion reply:**

200 Switching to ASCII mode.

200 Switching to Binary mode.

200 NOOP ok.

200 PORT command successful. Consider using PASV.

214 The following commands are recognized

226 Directory send OK.

230 Login successful.

250 Rename successful.

Ejemplos de respuestas FTP (Cont'd)

** Positive intermediate reply:

331 Please specify the password.

350 Ready for RNT0.

** Transient negative completion reply:

426 Failure writing network stream.

** Permanent negative completion reply:

500 Command not understood.

504 Bad MODE command.

530 Please login with USER and PASS.

550 RNFR command failed.

550 Failed to open file.

Sesión FTP de ejemplo

```
? ftp 127.0.0.1
Connected to 127.0.0.1.
220 (vsFTPd 2.0.5)
Name (127.0.0.1:andres): andres2
331 Please specify the password.
Password: ****
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
```

Sesión FTP de ejemplo (Cont'd)

```
ftp> help
```

```
Commands may be abbreviated.  Commands are:
```

!	debug	mdir	qc	send
\$	dir	mget	sendport	site
account	disconnect	mkdir	put	size
append	exit	mls	pwd	status
ascii	form	mode	quit	struct
bell	get	modtime	quote	system
binary	glob	mput	recv	sunique
bye	hash	newer	reget	tenex
case	help	nmap	rstatus	tick
cd	idle	nlist	rhel	trace
cdup	image	ntrans	rename	type
chmod	lcd	open	reset	user
close	ls	prompt	restart	umask
cr	macdef	passive	rmdir	verbose
delete	mdelete	proxy	runique	?

```
ftp> quit
```

Modalidades de FTP

- FTP Activo (modalidad vieja):
 - Conexión de control: port 21.
 - Conexión de datos: port 20.
 - Se diferencia como maneja la conexión de datos.
 - El servidor de forma activa se conecta al cliente para generar la conexión de datos.

PORT h1,h2,h3,h4,p1,p2

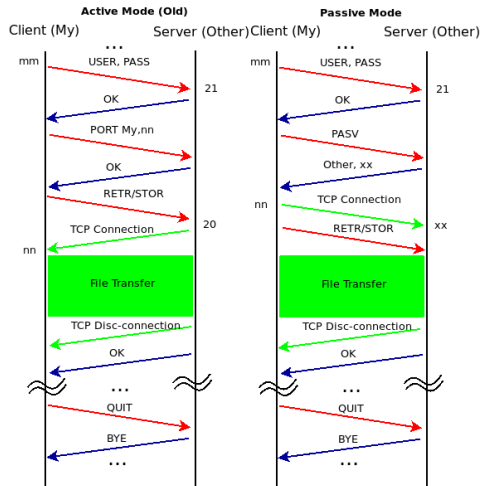
PORT 127,0,0,1,4,3 == 127.0.0.1:1027 (4*256)+3

- FTP Pasivo
 - Conexión de control: port 21.
 - Conexión de datos: port no privilegiado.
 - El servidor de forma pasiva indica al cliente a que nuevo puerto debe conectarse.

PASV

227 h1,h2,h3,h4,p1,p2 e.g. 227 127,0,0,1,4,3

FTP Activo vs. Pasivo



Algunos Comandos FTP que requieren Conexión de Datos

- Enviar un archivo al servidor (STOR).
- Traer un archivo desde el servidor (RETR).

```
ftp> put uno
```

```
local: uno remote: uno
```

```
200 PORT command successful. Consider using PASV.
```

```
150 Ok to send data.
```

```
226 File receive OK.
```

```
78257 bytes sent in 0.01 secs (15067.6 kB/s)
```

```
ftp> ls
```

```
200 PORT command successful. Consider using PASV.
```

```
150 Here comes the directory listing.
```

```
-rw-r--r--      1 1006      1007   26 May 07 18:47 otro
```

```
-rw-r--r--      1 0         0       9 May 07 18:48 uno
```

```
226 Directory send OK.
```

Algunos Comandos FTP que requieren Conexión de Datos

- Traer un archivo desde el servidor (RETR).

```
ftp> passive
ftp> get otro
local: otro remote: otro
227 Entering Passive Mode (127,0,0,1,120,149)
150 Opening BINARY mode data connection
    otro (1624334 bytes).
226 File send OK.
```

Algunos Comandos FTP que NO requieren Conexión de Datos

- Ver en el directorio donde está parado (CWD, PWD).
- Ver información de un archivo en particular (STAT, SIZE).
- Cambiar el modo (MODE), Obtener ayuda (HELP).
- Borrar un archivo (DELE).
- Crear/Borrar un directorio (MKD, RMD).
- Moverse entre directorios (CD).

```
ftp> cd Public
250 Directory successfully changed.
ftp> mkdir OTRO
257 '/home/andres2/OTRO' created
ftp> rmdir OTRO
250 Remove directory operation successful.
```

FTP, Formato de Datos (Bytes)

- FTP tiene funcionalidad de la capa ISO L6(representación).
- Debido a los diferentes tipos de plataformas, las archivos pueden ser convertidos a diferentes representaciones.
- Es responsabilidad del cliente indicarle al servidor el tipo/formato, sino el default es ASCII, aunque hoy es más común encontrar image.
- Los tipos son:
 - ASCII A NVT-ASCII.
 - EBCDIC E EBCDIC Text.
 - IMAGE I Raw binary, serie de bytes.
 - LOCAL L Raw binary, serie de bytes, usando var. byte size.

FTP, Formato de Archivos

- Las plataformas (OS) pueden almacenar los archivos en diferentes estructuras.
- FTP define estructuras para transportar datos.
- Formato default File (F).
- Se especifica el formato para transferencia con el comando STRU.
 - File F Unstructured, sequence of bytes.
 - Record R Series of records.
 - Page P Series of data blocks (pages).

FTP, Modo de Transferencia

- MODE se usa para especificar una codificación adicional aplicada sobre los datos transmitidos, de forma independiente del Formato del Archivo.
 - Stream S stream of bytes: Si es R el formato EOF se pone como registro Si es F el formato EOF indica el cierre del stream.
 - Block B Archivo se envía como header más secuencia de bloques. Permite interrumpir y reiniciar.
 - Compressed C Datos comprimidos usando RLE: Run Length Encoding. BBBBNN == 6B2N.
- Habitualmente no soportado: :-()

```
ftp> mode C
```

```
We only support stream mode, sorry.
```

Alternativas a FTP

- Versiones de FTP seguras:
 - FTPS, FTP over SSL/TLS.
- Versiones integradas con la suite de Open-SSH:
 - SCP (Secure remote Copy).
 - SFTP (Secure FTP).
- Aplicación para compartir recursos:
 - NFS.
 - SMB/CIFS.
 - iSCSI, ...
- TFTP.

FTP vs. HTTP

FTP	HTTP
Diseñado para Upload	Se puede con PUT, POST
Soporte de Download	Soporte de Download
Formato ASCII/binary cliente selecciona	meta-data with files, Content-Type, más flexible
No maneja Headers	Manjea Headers, mas información
FTP Command/Response, archivos pequeños más lento	Pipelining
Más complejo con Firewalls y NAT	Más amigable con Firewalls y NATs
Dos conexiones, y modo Activo o Pasivo	Una conexión, más sencillo
Soportan Ranges/resume	Range/resume HTTP opciones más avanzadas
Soporte de Autenticación	Soporte de Autenticación
Soporte de Cifrado (problemas fwall)	Soporte de Cifrado
Soporte de Compresión RLE	Soporte de Compresión Deflate: LZ77+Huffman



TCP/IP Illustrated, Volume 1: The Protocols, Addison-Wesley, 1994. W. Richard Stevens.



TCP/IP Illustrated, Volume 1: The Protocols, Addison-Wesley, (2nd. Ed). 2011. Kevin R. Fall, W. Richard Stevens.



Computer Networking: A Top-Down Approach, Addison-Wesley, (6th Edition). 2012. Kurose/Ross.



The Linux Home Page: <http://www.linux.org/>.



Linux in a Nutshell, Fourth Edition June, 2003. O'Reilly. Ellen Siever, Stephen Figgins, Aaron Weber.



FileZilla Project. <http://filezilla-project.org/>. vsftpd <http://vsftpd.beasts.org/>.



<http://www.rfc-editor.org/rfc/rfc793.txt>. TCP Transmission Control Protocol (Jon Postel 1981 USC-ISI IANA).



A FILE TRANSFER PROTOCOL. A. Bhushan (MIT), 1971.
<http://tools.ietf.org/html/rfc114>.



FILE TRANSFER PROTOCOL. J. Postel (ISI), 1980. <http://tools.ietf.org/html/rfc765>.



FILE TRANSFER PROTOCOL. J. Postel, J. Reynolds (ISI), 1985.
<http://tools.ietf.org/html/rfc959>.



Firewall-Friendly FTP, S. Bellovin (AT&T), 1995. <http://tools.ietf.org/html/rfc1579>.



FTP Security Extensions. M. Horowitz (Cygnus Solutions), S. Lunt (Bellcore), 1997.
<http://tools.ietf.org/html/rfc2228>.



OpenSSL project: <http://www.openssl.org/>.



Ethereal, Wireshark. Autor original Gerald Combs, 2005.

<http://www.ethereal.com/>.

<http://www.wireshark.org/>.