

The z/OS FTP Server

Functions and Client and Server Exits



Steve Pryor

About DTS Software

1-919-833-8426 info@DTSsoftware.com



ABOUT DTS

PRODUCTS

DOWNLOADS

SUPPORT & TRAINING

LOGIN



Founded in 1991, DTS is a recognized world leader in enterprise storage management solutions and services.

- 1,000+ worldwide customers
- Inventors of dynamic disk space recovery and volume pooling technology
- Reliable software design in simple, common UI

Learn more

Why Us?



Agenda

- File Transfer
 - Importance, Mechanisms and Technologies
- z/OS Communications Server FTP
 - Control mechanisms, logs and SMF records, and file transfers
- Exit Routines
 - Server Exits
 - Client Exits
 - Exit Routine Automation

File Transfer

- Importance of FTP

- Open standards, widely used
- Simple to understand, robust

- File Transfer Mechanisms

- FTP (RFC959)

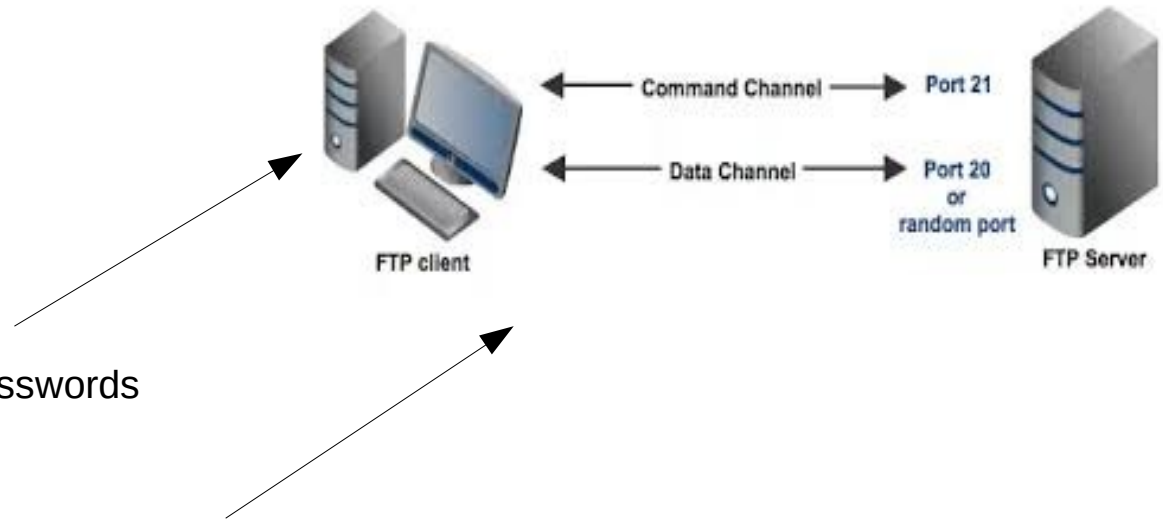
- Unencrypted, data & control conn, passwords

- FTP over SSL/TLS (RFC4217)

- Many names: FTPS, FTP AUTH-TLS, FTP AUTH-SSL, etc., etc.
- Encrypted via AT-TLS, SSL/TLS, Kerberos, etc.

- sftp – Secure Shell FTP

- A subprotocol of SSH and not the same as FTPS
- Incompatible with FTP/FTPS

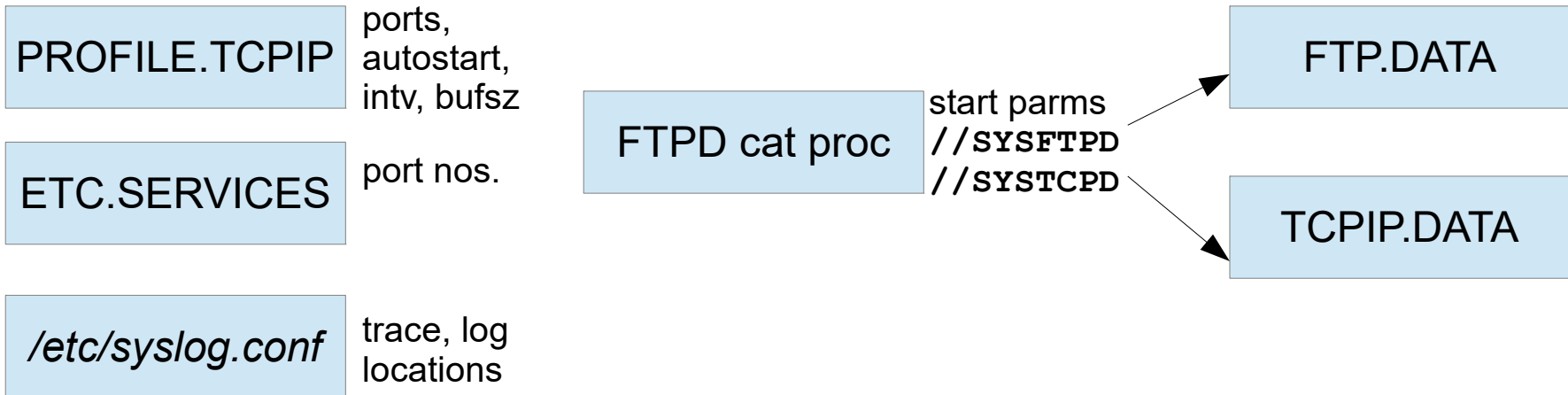


Communications Server FTP

- Functions
 - copy/move/delete/rename z/OS and USS files
 - manage the server file system
 - JES functions – submit, get status, get job output, NJE
 - SQL / DB2 queries

Communications Server FTP

- Control Mechanisms



Server and Client Config Options

- Configuration Statement **Server** Options – *search order*
 - //SYSFTPD in the FTP catalogued procedure
 - ftpserver.job.name.FTP.DATA
 - /etc/ftp/data
- Configuration Statement **Client** Options
 - TSO: **-f** operand, //SYSFTPD, userid.FTP.DATA, /etc/ftp/data
 - USS: **-f** operand, \$home/ftp/data, userid.FTP.DATA /etc/ftp/data,
- Defaults
 - SYS1.TCPPARMS(FTPDATA)
 - tcpip_hlq.FTP.DATA

Samples:

SEZAINST(FTPDATA) - 185 options
SEZAINST(FTCDATA) - 127 options

Retcode/msg configuration options

<u>Option name</u>	<u>Where</u>	<u>Values</u>	<u>Result</u>
CHKCONFIDENCE	both	TRUE/FALSE	High, Low, NoEOF, unknown
CLIENTERRORCODES	client	TRUE/FALSE/EXTENDED	FTP retcodes / Commserv codes / subcmd no.
CLIENTEXIT	client	TRUE/FALSE	Exit with nonzero RC if error
CONDDISP	both	CATLG/DELETE	Dataset status if error
LOGCLIENTERROR	client	TRUE/FALSE	EZZ9830I if subcmd error

Managing FTP File Transfers

- Manage **login**
 - Name, password, ip address port
- Manage **commands**
 - Command name, command type, operands
 - Allow, disallow, or modify
- Manage **errors**
 - Detect command failures, failed transfers
- Manage **logs** and **messages**
 - Syslog, SMF records, logger, etc.
 - Detect messages and take actions
 - Define and issue custom messages

Server Exit Requirements

- APF-authorized
- Reentrant
- Linklist/STEPLIB/LPA
- RACF program-controlled

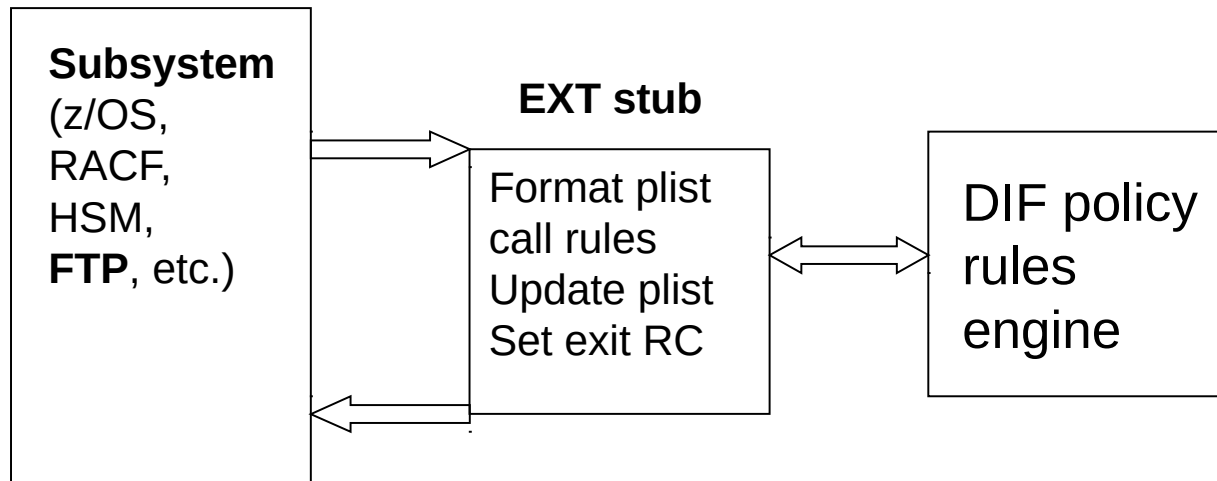
FTP Server Exits

- FTCHKCMD – called for any FTP command
 - **Examine, accept** or **reject** the command or **modify** arguments
 - 14-word Parmlist pointers to:
 - Userid,command, parameters, retcode
 - Current directory and type (MVS/Unix), filetype (SEQ,JES,SQL)
 - Command modify buffer, reply extension buffer (for reject)
 - Session id buffer, scratchpad buffer
 - IP socket addresses for client & server control connection

Samples in SEZAINST(FTCHKCMD, FTCHKCM1, FTCHKCM2)

Easy/Exit Architecture

- Parameter lists and return codes become DIF variables
- Assembler language is eliminated
- Logic changes and updates are simple refreshes



FTP Server Exits

- FTPOSTPR – *After* completion of file processing
 - Actions for RETR, STOR, STOU, APPE, DELE, RNTD cmds
 - whether successful or not
 - 20-word Parmlist points to:
 - Userid, client IP address, port, FTP command code
 - Current directory and type (MVS/Unix), filetype (SEQ,JES,SQL)
 - Session id buffer, scratchpad buffer, socket control connection info
 - Dataset name or Unix file name
 - FTP reply code, FTP reply line sent to client, FTP reply buffer
 - file transfer completion code, confidence level, bytes transferred

Samples in SEZAINST(FTPOSTPA, FTOPOSTPR)

FTP Server Exits

- FTCHKIP – at login or OPEN command for new connection
 - **Examine** and optionally **reject** control connection
 - *Before* userid and password prompts
 - Results in 421 USER EXIT REJECTS OPEN
 - 8-word Parmlist pointing to:
 - Remote IP addr and port
 - Local IP addr and port
 - Socket info for server and client connections
 - Unique session identifier

FTCHKIP changes
require restart of FTPD

Sample in SEZAINST(FTCHKIP)

FTP Server Exits

- FTCHKPWD – *after* userid/password but *before* authentication
 - **Examine** and **accept** or **reject** login for user, password, passphrase
 - 530 PASS COMMAND FAILED (more if ACCESSERRORMSG=TRUE)
 - For anonymous, called after USER prompt *and* PASSWORD prompt
 - 9-word Parmlist points to:
 - Retcode, userid, [password/phrase](#), userdata buffer
 - [no. of bad passwords this session](#)
 - Socket info for client and server control connections

Sample in SEZAINST(FTCHKPWD)

FTP Server Exits

- FTCHKJES – when a job is submitted via FILETYPE=JES
 - Called for every JCL record
 - **Examine, allow, or reject** the job
 - 550 USER EXIT REFUSES THIS JOB TO BE SUBMITTED BY userid
 - 14-word Parmlist points to:
 - Retcode, userid
 - Current logical record no., bytes transmitted so far
 - JES LRECL, RECFM (F/V)
 - Client id, JES user exit anchor info area
 - Socket info, session id, scratchpad

Sample in SEZAINST(FTCHKJES)

FTP Client Exits - Uses

- Protect datasets from being transferred off-host
- Examine or modify DSNs being transferred
- Examine, modify, or reject commands
- Inspect server replies
- End the client address space

Requirements

- APF-authorized
- Reentrant
- Linklist/STEPLIB
- Assembler (not C)
- Dynamic Exits

FTP Client Exits

- EZAFCCMD – *before* command is sent to server
 - **Examine, accept, reject**, command, **modify** arguments, **end** client
 - 15 word Parmlist points to:
 - Connection id, ftp userid, ftp command
 - Command arguments buffer (5122 bytes)
 - Current local directory type (MVS/UNIX) and directory name, unix FILE/FIFO, MVS dataset name or absolute path name
 - Local client config (SITE) options
 - Buffers for modified command arguments and additional message
 - Socket info and APPLDATA

Except AUTH, EPSV, SITE, and some others



Sample in SEZAINST(EZAFCEXT)--> EZAFCCMD/EZAFCREP/EZAFCCOM/ASMTSYSYL)

FTP Client Exits

- EZAFCREP – for each reply from server
 - **Examine** reply, **end** client (with optional additional message)
 - for *each* line of a multi-line reply
 - 5-word Parmlist points to:
 - Connection ID
 - Server reply
 - Socket APPLDATA
 - Additional message buffer for EZZ1556I message

SMF and Log Recording

- Type 118 – older format
 - FTPSMFEX exit
- Type 119 – common record, mapped by EZASMF77
 - IEFU83/IEFU85
- FTP.DATA controls (STD / nnn / TYPE119)

SMF
SMFAPPE
SMFDCFG
SMFDEL
SMFEXIT

SMFJES
SMFLOGN
SMFREN
SMFRETR
SMFSQL
SMFSTOR

FTPLOGGING, TRACE (syslogd)

Solving Problems with Easy/Exit

- Not just for FTP – virtually any exit
 - TSO, MPF, JES, RACF, IEFACTRT, IEFUTL, ALLOC ...
- You probably have exits in use
 - Suffering from lack of support, lack of time, resources, expertise
 - Easy/Exit can automate it!
- Every installation has some nagging problems
 - But they rarely tell us!
 - Because the problem is ‘small’ (but still a problem)
 - Because they’re unaware of a solution

Documentation & More Info

- z/OS Comm Server IP Config Guide SC27-3650-40
 - Chapter 12, Transferring Files Using FTP
- z/OS Comm Server IP Config Ref SC27-3651-40
 - Chapter 14, File Transfer Protocol
- z/OS Comm Server IP Prog Guide & Ref SC27-3659-40
 - Chapter 12, FTP Client API
- DTS Software Easy/Exit User's Guide
 - Chapter 13, FTP Exits

Questions ???

- Upcoming Webinars
 - An Introduction to Traditional DASD Backup/Recovery with DFSMSHsm and FDRABR – September 8, 2020
 - Using REXX with DFSMSrmm – October 6, 2020