



# 인터넷에서 사용되는 여러 기술

## FTP 이야기 3

File  
Transfer  
Protocol

글  
조인준  
KBS 미디어기술연구부 수석연구원

지난 편에서는 FTP(File Transfer Protocol) 연결이 어떻게 만들어지고 유지·종료되는지, 데이터 전송이 어떤 절차로 이루어지는지, 그리고 FTP가 지원하는 여러 전송 모드와 데이터 타입에는 어떤 것들이 있는지 알아보았습니다. 이런 기능들을 실행하여 파일 전송이라는 작업을 완료하기 위해 대부분의 인터넷 프로토콜과 마찬가지로 FTP도 클라이언트 측 FTP 프로세스가 서버 측 FTP 프로세스에 특정 명령을 보내면, 서버는 그 명령을 처리한 뒤 결과를 ‘응답’ 형태로 반환합니다. 지금부터 FTP가 내부적으로 어떻게 통신하고 동작하는지를 구체적으로 알 수 있는 FTP 명령과 응답의 구조 및 역할을 자세히 알아보겠습니다.

FTP 세션(Session: 두 시스템이 일정 시간 서로 통신하기로 한 논리적 연결 상태)은 클라이언트가 FTP 서버에 TCP 연결을 설정하는 것으로 시작됩니다. 이 연결이 만들어지면, 클라이언트 측의 FTP 프로세스와 서버 측의 FTP 프로세스 사이에 논리적인 제어 연결이 형성됩니다. 이 제어 연결을 통해 FTP의 모든 동작을 제어하기 위한 통신이 이루어집니다. 즉, FTP의 동작을 관리하기 위한 명령과 응답이 모두 이 연결을 통해 오가게 되는 것입니다. 클라이언트 측의 FTP 프로세스가 제어 연결을 통해 서버 측의 FTP 프로세스로 명령을 보내면, 서버는 그 명령에 대해 로그인 확인, 디렉터리 변경, 파일 전송 등의 동작을 수행한 후 각각에 대한 결과를 세 자리 숫자 코드로 된 응답으로 돌려보내어 명령 처리의 성공 여부나 추가 정보 등의 필요 여부를 클라이언트에게 알려줍니다. FTP의 각 명령은 RMD, USER, PASS, LIST와 같은 3~4글자의 알파벳 약자로 식별됩니다. FTP에는 다양한 명령이 존재하는데 기능적 성격에 따라 Access Control Commands(접근 제어 명령), Transfer Parameter Commands(전송 매개변수 명령), FTP Service Commands(서비스 명령)의 세 가지 그룹으로 분류됩니다. 적당한 한국어 번역을 찾지 못하여 영문을 기준으로 기재하였습니다.

### ☁ Access Control Commands (접근 제어 명령)

클라이언트가 FTP 서버에 로그인하고 인증을 받는 과정에 사용되며, 파일 등의 자원 접근이나 세션 관리(연결 유지·종료 등)에도 사용하는 명령어 그룹입니다.

#### ☑ USER

- FTP 서버는 누가 접속했는지 확인할 필요가 있음
- 클라이언트는 가장 먼저 USER 명령으로 사용자 이름(사용자 ID)을 서버에 알려줌
- 서버는 이에 따라 암호가 필요할 경우 암호를 전달하는 PASS 명령을 요구하거나, 암호가 필요 없으면 바로 인증 성공 코드 등을 반환

- 아래 예시에서 USER 명령에 대해 암호가 필요한 경우 331 코드를 반환하고, 암호가 필요 없는 경우 230 코드를 반환
- 응답 코드는 FTP 표준에 정의되어 있는 코드를 사용해야 하며, 뒤에 붙는 텍스트는 표준화된 텍스트가 아닌 사람이 이해할 수 있는 텍스트로 서버 구현마다 달라질 수 있음(이후의 설명에도 적용되며 응답코드의 전체적 내용은 추후 자세히 설명)

| 구분 | 방향         | 내용                                                              |
|----|------------|-----------------------------------------------------------------|
| 명령 | 클라이언트 → 서버 | USER kobeta                                                     |
| 응답 | 서버 → 클라이언트 | 331 Password required for kobeta<br>230 User logged in, proceed |

#### ☑ PASS (Password)

- USER 명령으로 사용자를 지정한 후 서버가 331 응답코드로 비밀번호를 보내라고 하면 클라이언트는 PASS 명령으로 비밀번호를 전송
- 비밀번호는 보안이 적용되지 않은 초기의 일반 FTP에서는 텍스트 그대로 전송되었고, 암호화되지 않았기 때문에 비밀번호가 노출될 수 있었음
- 오늘날에는 TLS로 암호화된 FTPS 또는 SSH 기반의 SFTP가 선호됨
- 비밀번호가 맞으면 서버는 230 응답코드로 로그인 성공을 알리거나, 틀리면 530 코드로 로그인 실패를 알림

| 구분 | 방향         | 내용                                               |
|----|------------|--------------------------------------------------|
| 명령 | 클라이언트 → 서버 | PASS kobeta_1234                                 |
| 응답 | 서버 → 클라이언트 | 230 User logged in, proceed<br>530 Not logged in |

#### ☑ ACCT (Account)

- 초기 FTP에 포함되어 있었지만, 오늘날 거의 사용되지 않는 명령
- 예전에는 하나의 사용자 이름으로 프로젝트 등에 따라 여러 계정을 가지고 접근 권한을 구분했음
- 서버가 USER와 PASS만으로는 인증이 충분하지 않다고 판단하면 332(계정 필요) 응답을 보내고, 클라이언트는 ACCT 명령으로 추가 계정을 서버에 전달해야 함
- ACCT 명령에 대한 응답으로는 230(성공), 332(서버가 기대한 계정명이 아니거나 권한이 없는 계정), 530(실패) 등이 있을 수 있음
- 오늘날 대부분의 FTP 서버는 ACCT를 필요로 하지 않기 때문에 202(불필요) 응답을 보냄

| 구분 | 방향         | 내용                                                                             |
|----|------------|--------------------------------------------------------------------------------|
| 명령 | 클라이언트 → 서버 | ACCT account-name                                                              |
| 응답 | 서버 → 클라이언트 | 230 User logged in, proceed<br>332 Need account for login<br>530 Not logged in |

☑ CWD (Change Working Directory)

- CWD는 현재 작업 중인 디렉터리를 변경
- FTP 서버는 보통 사용자 홈 디렉터리에서 세션을 시작
- CWD를 이용해 다른 경로로 이동할 수 있음
- 응답은 250(이동 성공) 또는 550(실패, 해당 경로 없음) 등이 있을 수 있음

| 구분 | 방향         | 내용                                                           |
|----|------------|--------------------------------------------------------------|
| 명령 | 클라이언트 → 서버 | CWD /home/kobeta/tech                                        |
| 응답 | 서버 → 클라이언트 | 250 CWD command successful<br>550 Failed to change directory |

☑ CDUP (Change Directory Up)

- 현재 폴더의 상위 디렉터리로 이동
- 응답 코드로는 200 또는 250(이동 성공)이나 550(상위 폴더가 없거나 권한이 없는 경우) 등이 있을 수 있음

| 구분 | 방향         | 내용                                                                                            |
|----|------------|-----------------------------------------------------------------------------------------------|
| 명령 | 클라이언트 → 서버 | CDUP                                                                                          |
| 응답 | 서버 → 클라이언트 | 200 CDUP command successful<br>250 Directory changed to home/kobeta/<br>550 Permission denied |

☑ SMNT (Structure Mount)

- 과거 메인 시스템 외의 다른 파일 시스템(네트워크 드라이브 등)을 마운트해서 접근하는 기능을 FTP 명령에서 제공하려 했음
- 현대에는 사용되지 않는 기능으로 표준에서만 남아 있으며 FTP 서버들은 대부분 응답코드 502(명령에 관한 구현 없음)로 응답

☑ REIN (Reinitialize)

- REIN은 로그아웃과 비슷하지만, 연결은 유지하는 특별한 명령
- TCP 연결은 그대로 두고 사용자 인증만 초기화
- 이후 다른 사용자로 다시 USER, PASS 절차를 밟을 수 있음
- 응답은 220(새로운 연결 준비완료) 등이 있을 수 있음

| 구분 | 방향         | 내용                             |
|----|------------|--------------------------------|
| 명령 | 클라이언트 → 서버 | REIN                           |
| 응답 | 서버 → 클라이언트 | 220 Service ready for new user |

#### ☑ QUIT

- FTP 세션을 종료할 때 반드시 QUIT을 보내는 것을 권장
- 서버에게 모든 작업을 마쳤으니, 연결을 정상적으로 닫겠다고 알리는 목적
- 만약 클라이언트가 QUIT 없이 연결을 끊으면 서버는 비정상 종료로 인식하여 로그에 오류를 남길 수 있음
- 응답코드는 221(서버가 제어 연결을 정상적으로 닫고 있음) 등이 있을 수 있음

| 구분 | 방향         | 내용          |
|----|------------|-------------|
| 명령 | 클라이언트 → 서버 | QUIT        |
| 응답 | 서버 → 클라이언트 | 221 Goodbye |

#### ☁ Transfer Parameter Commands (전송 매개변수 명령어)

앞서 설명한 Access Control Commands(접근 제어 명령)가 누가, 어디에 접속하는지에 관한 명령들이라면 Transfer Parameter Commands(전송 매개변수 명령)는 파일이 어떤 방식으로 전송될지를 설정하는 명령들입니다. 예를 들어 전송할 파일의 데이터 타입(텍스트/바이너리)을 지정하거나, FTP가 데이터를 주고받을 때 능동 모드(Active) 또는 수동 모드(Passive)를 사용할지를 결정하는 등의 명령이 여기에 포함됩니다.

#### ☑ TYPE

- 파일을 어떤 형식으로 보낼지 서버에 알려주는 전송 매개변수 명령
- TYPE I(Image): Binary 모드, 가공 없이 바이트 그대로 전송하며 현대 FTP의 기본값
- TYPE A(ASCII): 텍스트 모드, 개행문자(줄 바꿈 문자)를 상대 OS에 맞게 변환
- TYPE E(EBCDIC): 옛 IBM 계열 텍스트 모드, 현대에 거의 사용 안 함
- TYPE L(Local byte size) n: 비표준 바이트 길이를 n에 지정, 오늘날 거의 사용 안 함
- 아래와 같이 TYPE I, TYPE A 등과 같이 명령을 보내면 응답코드로 200(성공) 등을 받을 수 있음

| 구분 | 방향         | 내용                |
|----|------------|-------------------|
| 명령 | 클라이언트 → 서버 | TYPE I            |
| 응답 | 서버 → 클라이언트 | 200 Type set to I |

| 구분 | 방향         | 내용                |
|----|------------|-------------------|
| 명령 | 클라이언트 → 서버 | TYPE A            |
| 응답 | 서버 → 클라이언트 | 200 Type set to A |

#### ☑ STRU (Structure)

- 파일을 어떤 구조로 전송할지 서버에 알려주는 FTP 전송 매개변수 명령
- STRU F(File): 데이터를 연속된 바이트로 전송하며 오늘날 사용되는 FTP의 기본값
- STRU R(Record): 각 레코드(줄)가 명확히 구분되어 전송되며 거의 사용되지 않음
- STRU P(Page): 데이터가 페이지/블록으로 나뉘고 각 페이지에 메타 정보(길이, 인덱스/플래그 등)를 부여해 전송되며 STRU R과 같이 거의 사용되지 않음
- 다음의 예와 같이 STRU F 같이 명령을 보내고 응답코드로 200(성공) 등을 받을 수 있음

| 구분 | 방향         | 내용                     |
|----|------------|------------------------|
| 명령 | 클라이언트 → 서버 | STRU F                 |
| 응답 | 서버 → 클라이언트 | 200 Structure set to F |

☑ MODE (Mode)

- 데이터를 어떤 방식으로 전달할지를 정하는 FTP 전송 매개변수 명령
- MODE S(Stream): 가장 일반적인 방식으로 연결된 바이트 스트림으로 데이터를 전송
- MODE B(Block): 전송 데이터를 블록 단위로 메타정보와 함께 보내며 거의 사용되지 않음
- MODE C(Compressed): 런-렝스(Run-Length) 압축 같은 간단한 압축 방식으로 전송하며 거의 사용되지 않음
- 아래의 예와 같이 MODE S 같이 명령을 보내고 응답코드로 200(성공) 등을 받을 수 있음

| 구분 | 방향         | 내용                |
|----|------------|-------------------|
| 명령 | 클라이언트 → 서버 | MODE S            |
| 응답 | 서버 → 클라이언트 | 200 Mode set to S |

☑ PORT

- FTP의 능동 모드 데이터 전송을 위해 클라이언트가 자신의 IP 및 포트 번호를 서버에 알려주어 데이터 연결을 만드는 명령
- 예로써 PORT 192,0,2,10,195,80과 같이 명령을 보내면 처음 네 자리는 IP 주소 192.0.2.10을 가리키며 포트 번호는 남은 두 숫자 195,80과 256을 이용해  $195 \times 256 + 80 = 50000$ 이 됨
- 클라이언트가 외부에서 들어오는 연결을 받아야 하므로 NAT/방화벽에서 문제가 생길 수 있음
- 오늘날 FTP에서는 능동 모드보다 수동 모드를 선호
- 아래의 예와 같이 PORT 명령 후 응답코드로 200(성공) 등을 받을 수 있음

| 구분 | 방향         | 내용                          |
|----|------------|-----------------------------|
| 명령 | 클라이언트 → 서버 | PORT 192,0,2,10,195,80      |
| 응답 | 서버 → 클라이언트 | 200 PORT command successful |

☑ PASV (Passive)

- FTP의 수동 모드 데이터 전송을 위해 서버가 열어놓은 자신의 IP 및 포트 번호를 클라이언트에 알려주어 데이터 연결을 만들 수 있도록 하는 명령
- 예로써 PASV 명령에 대한 서버의 응답이 227 Entering Passive Mode(203,0,113,10,195,80)라면 서버의 IP 주소는 203.0.113.10이며 포트 번호는 PORT 명령의 예와 같이  $195 \times 256 + 80 = 50000$ 이 됨
- 227 Entering Passive Mode(203,0,113,10,195,80)와 같은 경우 응답코드 227 및 203,0,113,10,195,80 같은 숫자 패턴은 표준에 정의되어 있고, Entering Passive Mode 같은 텍스트는 서버 구현에 따라 달라질 수 있음
- 클라이언트가 외부로 나가는 연결만 열려있으면 되므로 클라이언트 방화벽/NAT 문제가 발생하지 않아서 오늘날 FTP 데이터 연결에 주로 사용됨

| 구분 | 방 향        | 내 용                                             |
|----|------------|-------------------------------------------------|
| 명령 | 클라이언트 → 서버 | PASV                                            |
| 응답 | 서버 → 클라이언트 | 227 Entering Passive Mode (203,0,113,10,195,80) |

#### ✓ EPRT (Extended Port)

- IPv6 시대를 대비한 FTP 능동모드 확장 명령
- IPv4/IPv6 주소와 포트 번호를 표기할 수 있음
- EPRT |a|addr|port| 형식으로 쓰이며 |는 구분자, af(Address Family) 1은 IPv4 및 2는 IPv6, addr은 IPv4 또는 IPv6 주소 문자열, port는 포트 번호를 나타냄
- EPRT |1|192.168.0.10|50000|은 IPv4 클라이언트 주소 192.168.0.10 및 포트 번호 50000으로 서버의 데이터 연결을 명령
- EPRT |2|2025:0db8::1|50000|은 IPv6 클라이언트 주소 2025:0db8::1 및 포트 번호 50000으로 서버의 데이터 연결을 명령
- 응답코드로 200(성공) 등을 받을 수 있음

| 구분 | 방 향        | 내 용                                                                   |
|----|------------|-----------------------------------------------------------------------|
| 명령 | 클라이언트 → 서버 | IPv4: EPRT  1 192.168.0.10 50000 <br>IPv6: EPRT  2 2025:0db8::1 50000 |
| 응답 | 서버 → 클라이언트 | 200 EPRT command successful                                           |

#### ✓ EPSV (Extended Passive)

- IPv6 시대를 대비한 FTP 수동모드 확장 명령
- EPSV af(Address Family) 형태로 쓰이며, af가 1이면 IPv4, 2이면 IPv6, 마지막으로 이를 생략하면 현재 제어 연결과 동일한 네트워크 프로토콜 사용을 의미
- 응답코드로 아래 예의 229(확장 수동모드 사용) 등을 받을 수 있음
- 229 응답코드에 따라오는 |||50000|은 EPRT의 |a|addr|port|와 동일한 형식이며, | 구분자 사이에 생략된 인자들은 현재 제어 연결과 동일한 네트워크 프로토콜(IPv4 or IPv6) 사용을 의미

| 구분 | 방 향        | 내 용                                            |
|----|------------|------------------------------------------------|
| 명령 | 클라이언트 → 서버 | EPSV                                           |
| 응답 | 서버 → 클라이언트 | 229 Entering Extended Passive Mode (   50000 ) |

지금까지 FTP 명령어 그룹 세 가지 중에 Access Control Commands(접근 제어 명령)와 Transfer Parameter Commands(전송 매개변수 명령)에 대해 알아보았습니다. 다음 편에서는 비교적 수가 많은 FTP Service Commands(FTP 서비스 명령)에 대해 알아보도록 하겠습니다. 📖



#### P.S.

C군이 여러분께 전하는 내용 중 전문적 성격이 짙은 것은 엄밀한 언어를 사용하여 설명하기에는 한계가 있습니다. 본 내용은 설명하는 대상에 대한 전체적 맥락의 이해에만 이용하시고, 그 이상은 권위 있는 전문자료를 참고하시기 바랍니다.