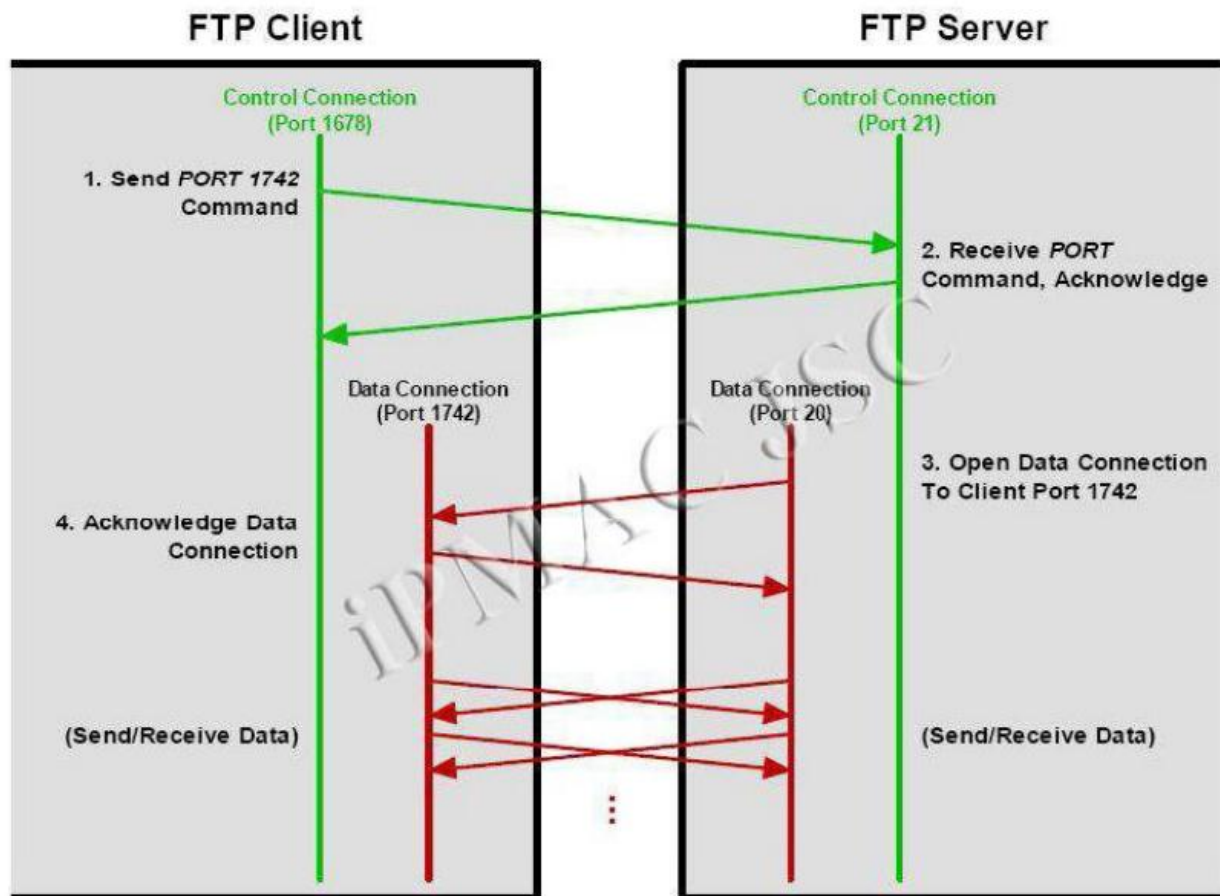


Phụ lục FTP

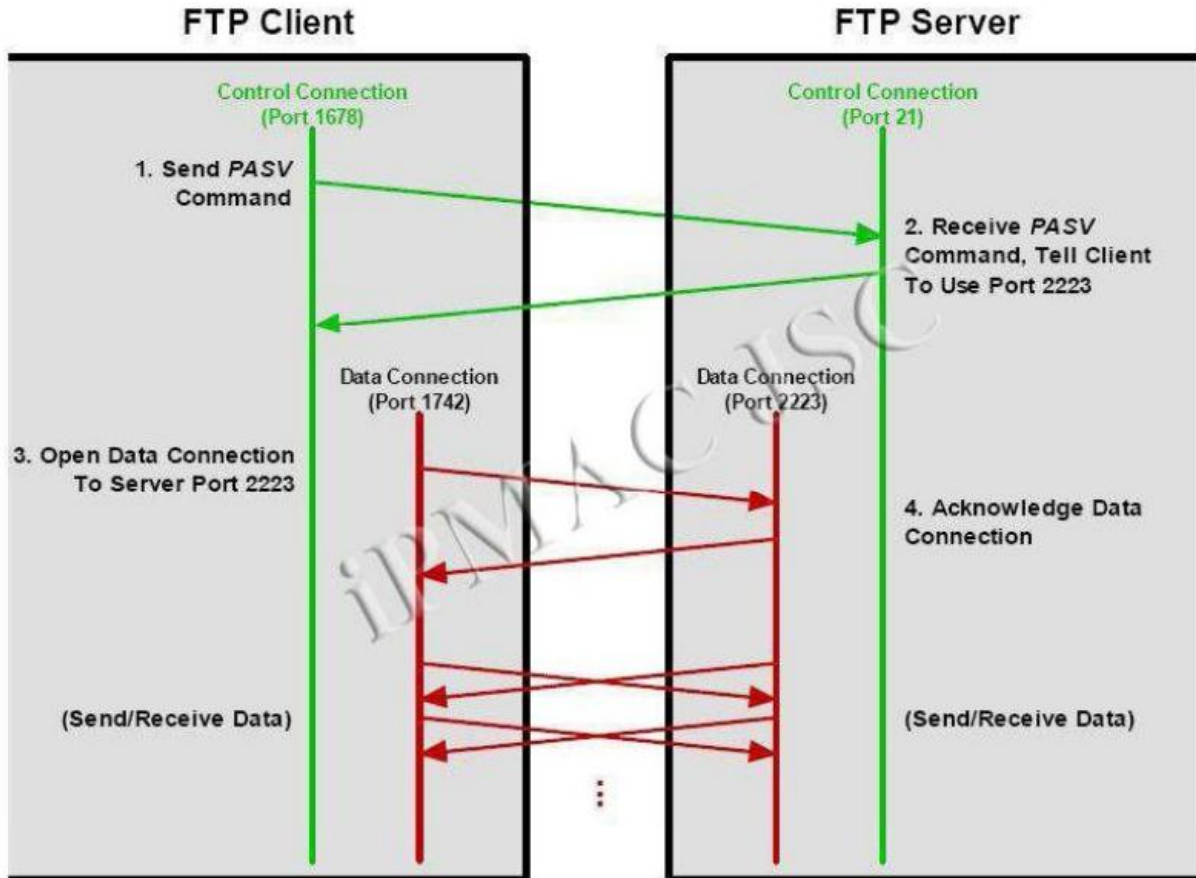
(Tóm tắt các thành phần cơ bản nhất của giao thức FTP – tham khảo thêm về FTP trên RFC 959)

1. Các mode hoạt động:

- FTP hỗ trợ 2 mode hoạt động chính
 - o **Active:** Server chủ động kết nối đến Client để truyền / nhận dữ liệu



- **Passive:** Server mở kết nối kênh dữ liệu và client connect vào để truyền/nhận dữ liệu.



2. Một số lệnh thông dụng:

(Sinh viên đọc RFC 959 để biết thêm thông tin về mã trạng thái trả về sau khi thực hiện lệnh)

- Đăng nhập:
 - **USER** <username>
 - Gửi qua server tên user name.
 - **PASS** <password>
 - Gửi password tương ứng.
- Truy vấn thông tin:
 - **PWD**
 - In ra đường dẫn của thư mục hiện hành.
 - **LIST**
 - Liệt kê danh sách tập tin, thư mục nằm trong thư mục hiện hành.

- Thao tác khác:
 - *MKD* <Folder name>
 - Tạo thư mục mới
 - *CWD* <folder name>
 - Di chuyển đến thư mục khác (tương tự lệnh CD trên linux)
 - *DELE* <file name>
 - Xóa file.
 - *RMD* <Folder name>
 - Xóa thư mục rỗng.
 - *PORT* <a, b, c, d, p₁, p₂>
 - Cho Server biết client muốn thực hiện truyền nhận dữ liệu ở mode Active.
 - Lệnh port sẽ thông báo cho server thông tin về socket đang mở để server kết nối vào và thực hiện truyền nhận dữ liệu.
 - A,b,c,d là 4 block tương ứng của địa chỉ IP client, p₁, p₂ cho biết thông tin về port đang được mở với
 - **p₁=port/256** (p₁ là phần nguyên từ phép chia của port thật sự cho 256)
 - và **p₂ = port%256** (p₂ là phần dư của phép chia của port thật sự cho 256).
 - Giả sử Client đang mở socket lắng nghe ở địa chỉ 192.168.1.2:4567 thì lệnh PORT sẽ là:

PORT 192,168,1,2,17,215 (*)
 - *PASV*
 - Cho Server biết Client muốn truyền nhận dữ liệu ở mode passive
 - Sau khi lệnh PASV được gửi đến server, Server sẽ trả về địa chỉ của socket mà server đang mở để client kết nối vào như cấu trúc (*) và client từ thông số đó sẽ tự tính toán và biết được mình cần kết nối vào địa chỉ nào.
 - *RETR* <file name>
 - Down 1 file từ server về client. (Ở bước này trong trường hợp lấy thông tin về thư mục con và tập tin nằm trong thư mục hiện hành thì dùng lệnh LIST – nghĩa là LIST và RETR tương tự nhau về các bước thực hiện)
 - *STOR* <filename>
 - Upload 1 file từ client lên server.
 - *QUIT*

- Thoát phiên làm việc và ngắt kết nối đến FTP server.

3. *Áp dụng các lệnh trong phiên làm việc cơ bản*

- Giao thức FTP dùng 2 kênh truyền
 - **Lệnh:** truyền các thông tin về lệnh, các respond...
 - **Dữ liệu:** chỉ mở ra khi giữa client và server cần truyền dữ liệu cho nhau
- Thông thường thì kênh truyền dữ liệu sẽ được mở trong 2 trường hợp chính sau đây

a. Lấy dữ liệu từ FTP server về với mode Active:

- Việc lấy dữ liệu từ Server về client khi client dùng 1 trong 2 lệnh sau
 - LIST: lấy danh sách file và thư mục con.
 - RETR: tải file.
- Giả sử **Client cần down 1 file từ server** về thì với **Active mode**, các lệnh và công việc cần làm sẽ như sau:
 - B1: Client tạo **csocket** và kết nối vào port 21 của server
 - B2: gửi thông tin USERNAME và PASSWORD để đăng nhập qua **csocket**
 - USER
 - PASS
 - B3: Client tạo thêm **dsocket** mới và lắng nghe
 - B4: dùng lệnh gửi thông tin socket vừa được tạo qua cho server thông qua **csocket**:
 - PORT a,b,c,d,p1,p2
 - B4: thông qua **csocket** gửi lệnh tải file
 - RETR <filename>
 - B5: chấp nhận kết nối từ server tại **dsocket** và thực hiện nhận file tại **dsocket**
 - B6: sau khi file đã truyền xong, đóng **dsocket**
 - B7: đóng **csocket**

b. Up 1 file lên FTP server với active mode:

- Các công việc cần thực hiện khi muốn up 1 file lên Server ở Active mode
 - B1: Client tạo **csocket** và kết nối vào port 21 của server
 - B2: gửi thông tin USERNAME và PASSWORD để đăng nhập qua **csocket**
 - USER
 - PASS
 - B3: Client tạo thêm **dsocket** mới và lắng nghe

- B4: dùng lệnh gửi thông tin socket vừa được tạo qua cho server thông qua **csocket**:
 - PORT a,b,c,d,p1,p2
- B4: thông qua **csocket** gửi lệnh tải file
 - STOR <filename>
- B5: chấp nhận kết nối từ server tại **dsocket** và thực hiện truyền file qua cho server tại **dsocket**
- B6: sau khi file đã truyền xong, đóng **dsocket**
- B7: đóng **csocket**

(Chú ý: nếu Client đã kết nối và đăng nhập vào FTP server thì có thể bỏ qua B1, B2 và nếu Client muốn giữ phiên làm việc với Server thì không cần thực hiện B7)

c. Lấy dữ liệu từ FTP server về với mode Passive:

- Giả sử **Client cần down 1 file từ server** về thì với **Passive mode**, các lệnh và công việc cần làm sẽ như sau (Tham khảo thêm các bước thực hiện trong project demo):
 - B1: Client tạo **csocket** và kết nối vào port 21 của server
 - B2: gửi thông tin USERNAME và PASSWORD để đăng nhập qua **csocket**
 - USER
 - PASS
 - B3: Client thông qua **csocket** cho server biết cần dùng passive mode
 - PASV
 - B4: nhận thông tin về IP và port từ server thông qua **csocket** và tạo **dsocket** kết nối vào server
 - Thông tin nhận về có dạng: a,b,c,d,p1,p2
 - Tạo **dsocket** và kết nối vào địa chỉ tính được (tham khảo mục 2)
 - B4: thông qua **csocket** gửi lệnh tải file
 - RETR <filename>
 - B5: thực hiện nhận file tại **dsocket**
 - B6: sau khi file đã truyền xong, đóng **dsocket**
 - B7: đóng **csocket**

d. Up 1 file lên FTP server với passive mode:

- Các công việc cần thực hiện khi muốn up 1 file lên Server ở Active mode
 - B1: Client tạo **csocket** và kết nối vào port 21 của server
 - B2: gửi thông tin USERNAME và PASSWORD để đăng nhập qua **csocket**

- USER
- PASS
- B3: Client thông qua **csocket** cho server biết cần dùng passive mode
 - PASV
- B4: nhận thông tin về IP và port từ server thông qua **csocket** và tạo **dsocket** kết nối vào server
 - Thông tin nhận về có dạng: a,b,c,d,p1,p2
 - Tạo dsocket và kết nối vào địa chỉ tính được (tham khảo mục 2)
- B5: thông qua **csocket** gửi lệnh tải file
 - STOR <filename>
- B6: thực hiện truyền file qua cho server trên **dsocket**
- B7: sau khi file đã truyền xong, đóng **dsocket**
- B8: đóng **csocket**

(Chú ý: nếu Client đã kết nối và đăng nhập vào FTP server thì có thể bỏ qua B1, B2 và nếu Client muốn giữ phiên làm việc với Server thì không cần thực hiện B8)