2. Diagrams (architecture)

a) Architecture Diagram

graph TD

A[FTP Client] -->|TCP Control Connection: Port 21| B[FTP Server]

A -->|TCP Data Connection: Dynamic Port| B

A -->|TCP Connection: Port 8888| C[ClamAV Agent]

C -->|ClamAV Scan| D[ClamAV Database]

B -->|File Storage| E[Server File System]

**Description**: The FTP Client connects to the FTP Server using a control connection (port 21) for commands and a data connection (dynamic port) for file transfers. Before uploading, the client sends files to the ClamAV Agent (port 8888) for virus scanning. The ClamAV Agent uses the ClamAV database to scan files and returns results to the client. The FTP Server stores uploaded files in its file system.

b) Data Flow Diagram

graph LR

A[FTP Client: Select File] -->|File Data| B[ClamAV Agent: Scan]

B -->|Scan Result: OK/INFECTED| A

A -->|File Data (if OK)| C[FTP Server: Store]

C -->|Confirmation| A

B -->|Virus Signatures| D[ClamAV Database]

C -->|Store File| E[Server File System]

**Description**: The FTP Client sends a file to the ClamAV Agent for scanning. The agent checks the file against the ClamAV database and returns “OK” or “INFECTED”. If the file is clean (“OK”), the client uploads it to the FTP Server via the data connection, which stores it in the file system. The server sends a confirmation back to the client.

c) Sequence Diagram

sequenceDiagram

participant C as FTP Client

participant A as ClamAV Agent

participant S as FTP Server

C->>A: Connect (TCP:8888)

A-->>C: Accept Connection

C->>A: Send File Data

A->>A: Scan with ClamAV

A-->>C: Return Result (OK/INFECTED)

C->>C: Check Result

alt Result = OK

C->>S: Connect Control (TCP:21)

S-->>C: 220 Welcome

C->>S: USER username

S-->>C: 331 Password Required

C->>S: PASS password

S-->>C: 230 Logged In

C->>S: PORT 127,0,0,1,p1,p2

S-->>C: 200 PORT OK

C->>S: STOR filename

S-->>C: 150 Opening Data Connection

C->>S: Send File Data (Data Connection)

S-->>C: 226 Transfer Complete

else Result = INFECTED

C->>C: Abort Upload

end

C->>A: Close Connection

C->>S: QUIT

S-->>C: 221 Goodbye

**Description**: The FTP Client connects to the ClamAV Agent (port 8888) and sends a file for scanning. The agent scans the file and returns “OK” or “INFECTED”. If “OK”, the client connects to the FTP Server (port 21), logs in, sets up a data connection using the PORT command, and uploads the file using STOR. The server confirms the transfer. If “INFECTED”, the upload is aborted. Finally, the client closes connections with both the agent and the server.

3. Screenshots of a successful session

Screenshot 1: Server Startup

Screenshot 2: Client Connection

Screenshot 3: Chat Session

4. Problems Encountered and Solutions

|  |  |  |  |
| --- | --- | --- | --- |
| Problem | Cause | Solution | Result |
| Server cannot login | Connect() function don’t provide login with user and password | Make ManualLogin() function | Now can input username and password |
| Server cannot change to passive mode | Client don’t send commend PASV | Make togglePassive() send commend PASV and make server response | Now server can change to passive mode and appear log on server |
| Recursive upload and download cannot run | Not inlcude library yet | Inlucde library string and filesystem | Now function can run and dow/up multiple file |

5. Summary of how each requirement was fulfilled

|  |  |  |
| --- | --- | --- |
| Requirement | Implemenation | Evidence |
| Session control | Write FTPClient class | Screenshot |
| Real-time message sending/receiving | Used TCP sockets with threading to handle concurrent client messages | Screenshot 3: Chat session showing messages exchanged |
| Reliable message delivery | Used TCP protocol for guaranteed deliver | Test case: No message loss in 1000-message test |