Assignment 1 – Report

Intro:

We have been tasked with cresting a python file sharing application with the aim of teaching us about basics of protocol design and socket programming for TCP connections. The brief provided to students specifies a client-server architecture and suggests features to add. This is a report summing up the functionality and protocol specification of said application.

**Description of system functionality and features:**

Client commands - Get, Post, Help, List

Encryption – to protect the files as they are being sent between the client and server, this application has implemented end-to-end encryption.

Reliable data transfer – Despite TCP having its own integrated solutions to reliable data transfer, as per the assignment brief, we have implemented our own solution… (timers ect.)

Data storage (server storage file and specifying client directory)

File protection and access (passowrds)

GUI

Checksum: This system should includes a file validation system….

, where the sender includes extra validation information within each message, and where the receiver is able to check that a file is exactly as it was sent by the source (i.e., file has not been altered in transit).

This is achieved by using a xxx checksum…

Loading bar

Stress testing – talk about gigo

**Specification (Protocol design & specification (sequence diagrams & message formats/structure))**

**Key phrases:**

defining the framework of communication

specifying requirements and constraints (i.e. reliability and authentication)

defining the types and structure of messages (three types: commands, data transfer, and control)

communication rules that specify the sequence of messages at every stage of communication

clearly specifying messages and reactions for every communication scenario. You will need to represent such rules with sequence diagrams (at lease two sequence diagrams will be required, one for upload process, and another for download process).

As per the description brief, this application uses Client-Server architecture. Here the server contains all the files and the clients

Reliable Data transfer

Message Header format

Server Sequence diagram

Client Sequence diagram

|  |
| --- |
| **Robust protocol implementation (stress tests)** |
| File types and sizes  GIGO – user control  No input can crash system |