

SnapIt

A SIMPLE FILE TRANSFER SERVER

A TCP File Transfer Application in Python with GUI

Project for Computer Networks (MC-308)

Submitted By:

Aiman Siddiqua
Apoorva

2K18/MC/008
2K18/MC/019



ABOUT THE PROJECT

The project timeline included the following steps

1. Learning the basics of Socket Programming.
2. Implementing a basic Server-Client connection via Sockets.
3. Outlining a simple process for the File Transfer Server to follow.
4. Creating a GUI in Python using tkinter module.
5. Implementing Socket Programming in the project.
6. Verifying Output.

Github Link: <https://github.com/apxxxva784/Simple-FTP-Server>



TCP/IP

Transmission Control Protocol is a transport layer protocol that facilitates the transmission of packets from source to destination. It is a connection-oriented protocol. This protocol is used with an IP protocol, so together, they are referred to as a TCP/IP.



Features of TCP include:

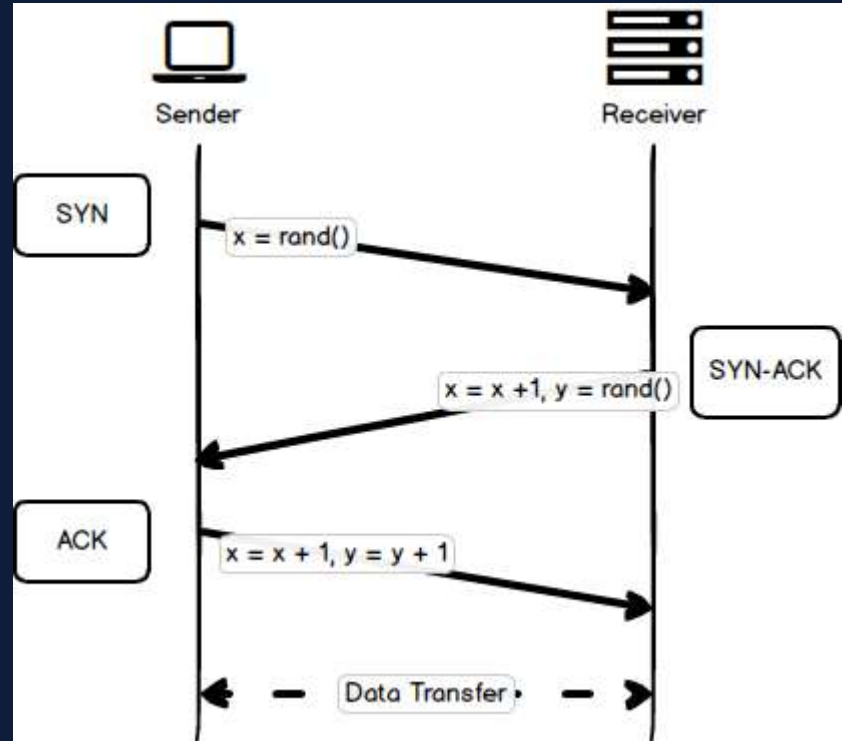
- ☐ Reliable
- ☐ Order of data is maintained
- ☐ Full Duplex
- ☐ Stream-Oriented
- ☐ Connection-Oriented



WORKING OF TCP/IP

A TCP connection is established by **three-way handshaking**.

- ❑ The client sends the segment with its sequence number.
- ❑ The server, in return, sends its segment with its own sequence number as well as the acknowledgment sequence, which is one more than the client sequence number.
- ❑ When the client receives the acknowledgment of its segment, then it sends the acknowledgment to the server.



THEORY OF SOCKET PROGRAMMING



Socket programming is a way of connecting two nodes on a network to communicate with each other. One socket(node) listens on a particular port at an IP, while the other socket reaches out to the other to form a connection. The server forms the listener socket while the client reaches out to the server.



Sockets may be implemented over a number of different channel types: Unix domain sockets, TCP, UDP, and so on. The socket library(or various languages) provides specific classes for handling the common transports as well as a generic interface for handling the rest.

Domain

Family of Protocols used as a transfer mechanism (eg. AF_INET (IPv4))

Type

Type of communication between two endpoints, this includes TCP (connection-oriented) or UDP (connection less)

Protocol

Typically 0, Variant of a protocol within a domain and type

Hostname And Port

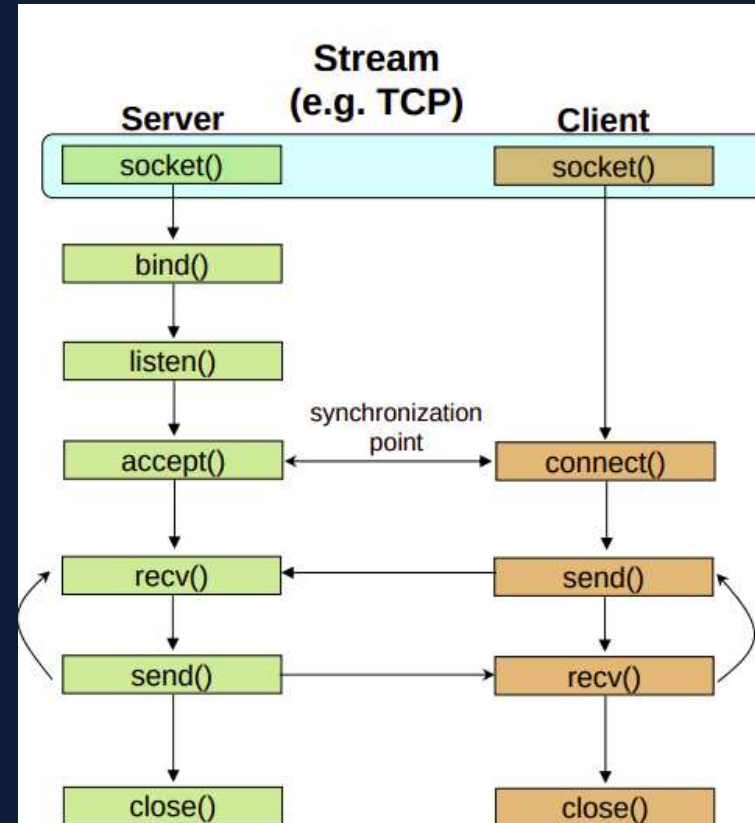
Identifiers of Network Interface
Such as IP Address & Port number



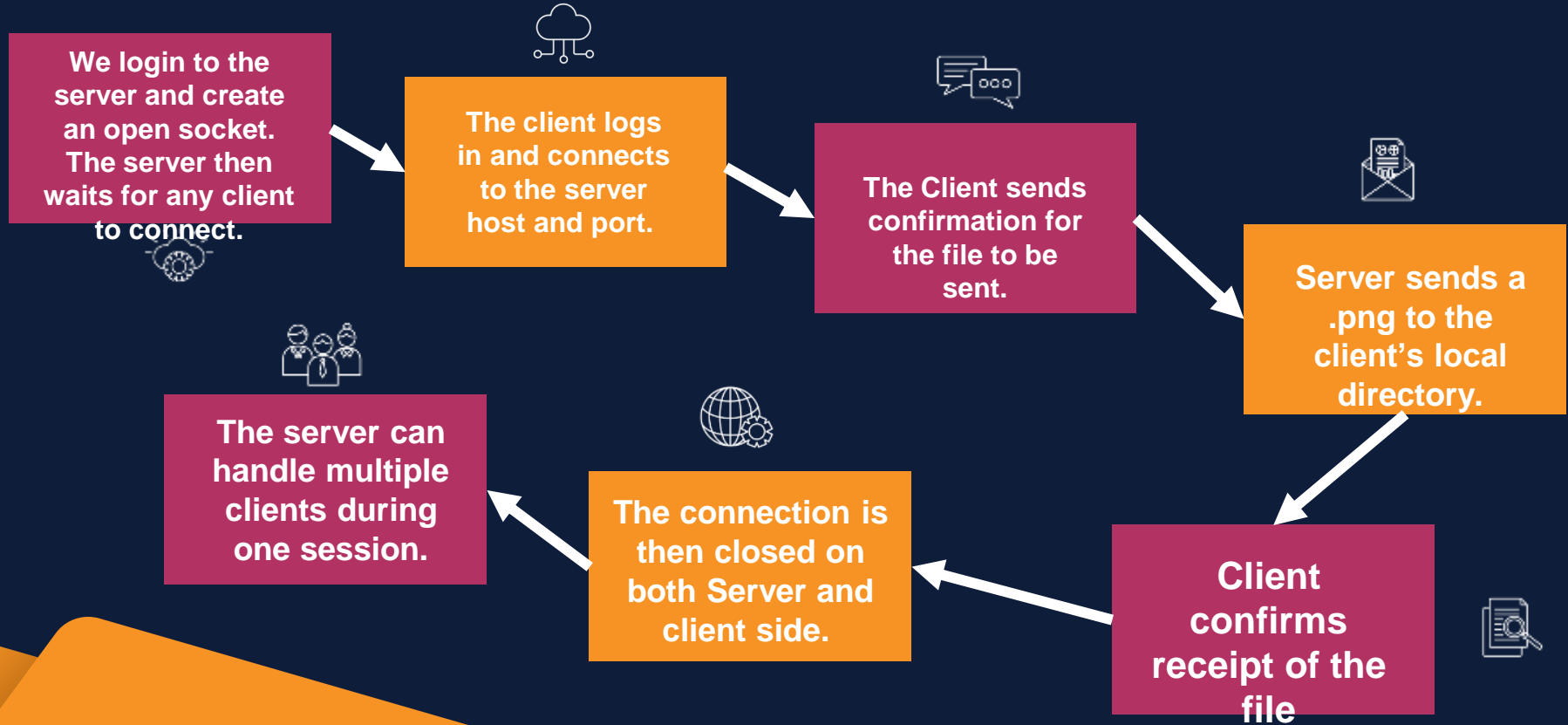
SOME TERMINOLOGY



STATE DIAGRAM



STEPS INVOLVED IN THE PROJECT



SCREENSHOTS FROM THE PROJECT

Server and Client Login

Simple FTP Server

Server Software

Username

Password

Login

FTP Client

Client Software

Username

Password

Login

Connection Screen

Simple FTP Server

Server Software

Enter Host NAME

Enter Port

Client Message

Waiting for connection...

Connect

FTP Client

Client Software

Enter Host NAME

Enter Port

Server Message

Send

Final Output Screen

Simple FTP Server

Server Software

Enter Host NAME

Enter Port

Client Message

```
Waiting for connection...
connected ('127.0.0.1', 59334)
Sending file..
File successfully sent!
Waiting for connection...
```

Connect

FTP Client

Client Software

Enter Host NAME

Enter Port

Server Message

```
Connected to server..
File Opened !
Receiving data...
File successfully received
Connection Closed
```

Send

CONFIRMING RECEIPT OF
PNG FILE



dtu.png



received_
file.png



Thank You