Vishwakarma Institute of Technology, Pune

**Name :** Suryawanshi Aditya Udhav

**Div :** TY - D **Roll No. :** 49 **Batch :** 2

**PRN No. :** 12111453 **Assignment No.** 5

**Subject :** Computer Networks

**Problem Statement :**

Develop a client server using TCP Berkeley socket primitives to transfer a file in peer  
to peer and client server mode.  
Demonstrate the packets captured traces using Wireshark Packet Analyzer Tool for peer to peer mode.

**Code:**

**client.c**

#include <stdio.h>

#include <winsock2.h>

#define SERVER\_PORT 8080

#define SERVER\_IP "127.0.0.1"

#define BUFFER\_SIZE 1024

void receiveFile(FILE \*fp, SOCKET sockfd);

int main() {

WSADATA wsaData;

SOCKET clientSocket;

struct sockaddr\_in serverAddr;

FILE \*fp;

char \*filename = "receivedFile.txt";

WSAStartup(MAKEWORD(2, 2), &wsaData);

clientSocket = socket(AF\_INET, SOCK\_STREAM, 0);

serverAddr.sin\_family = AF\_INET;

serverAddr.sin\_addr.s\_addr = inet\_addr(SERVER\_IP);

serverAddr.sin\_port = htons(SERVER\_PORT);

connect(clientSocket, (struct sockaddr \*)&serverAddr, sizeof(serverAddr));

fp = fopen(filename, "wb");

if (!fp) {

perror("Failed to open the file");

exit(1);

}

receiveFile(fp, clientSocket);

printf("File received successfully.\n");

fclose(fp);

closesocket(clientSocket);

WSACleanup();

return 0;

}

void receiveFile(FILE \*fp, SOCKET sockfd) {

int n;

char buffer[BUFFER\_SIZE];

while ((n = recv(sockfd, buffer, BUFFER\_SIZE, 0)) > 0) {

fwrite(buffer, sizeof(char), n, fp);

}

}

**server.c**

#include <stdio.h>

#include <winsock2.h>

#define SERVER\_PORT 8080

#define BUFFER\_SIZE 1024

void sendFile(FILE \*fp, SOCKET sockfd);

int main() {

WSADATA wsaData;

SOCKET listenSocket, clientSocket;

struct sockaddr\_in serverAddr, clientAddr;

FILE \*fp;

char \*filename = "fileToSend.txt";

WSAStartup(MAKEWORD(2, 2), &wsaData);

listenSocket = socket(AF\_INET, SOCK\_STREAM, 0);

serverAddr.sin\_family = AF\_INET;

serverAddr.sin\_addr.s\_addr = INADDR\_ANY;

serverAddr.sin\_port = htons(SERVER\_PORT);

bind(listenSocket, (struct sockaddr \*)&serverAddr, sizeof(serverAddr));

listen(listenSocket, 5);

int clientSize = sizeof(clientAddr);

clientSocket = accept(listenSocket, (struct sockaddr \*)&clientAddr, &clientSize);

fp = fopen(filename, "rb");

if (!fp) {

perror("Failed to open the file");

exit(1);

}

sendFile(fp, clientSocket);

printf("File sent successfully.\n");

fclose(fp);

closesocket(clientSocket);

WSACleanup();

return 0;

}

void sendFile(FILE \*fp, SOCKET sockfd) {

int n;

char buffer[BUFFER\_SIZE];

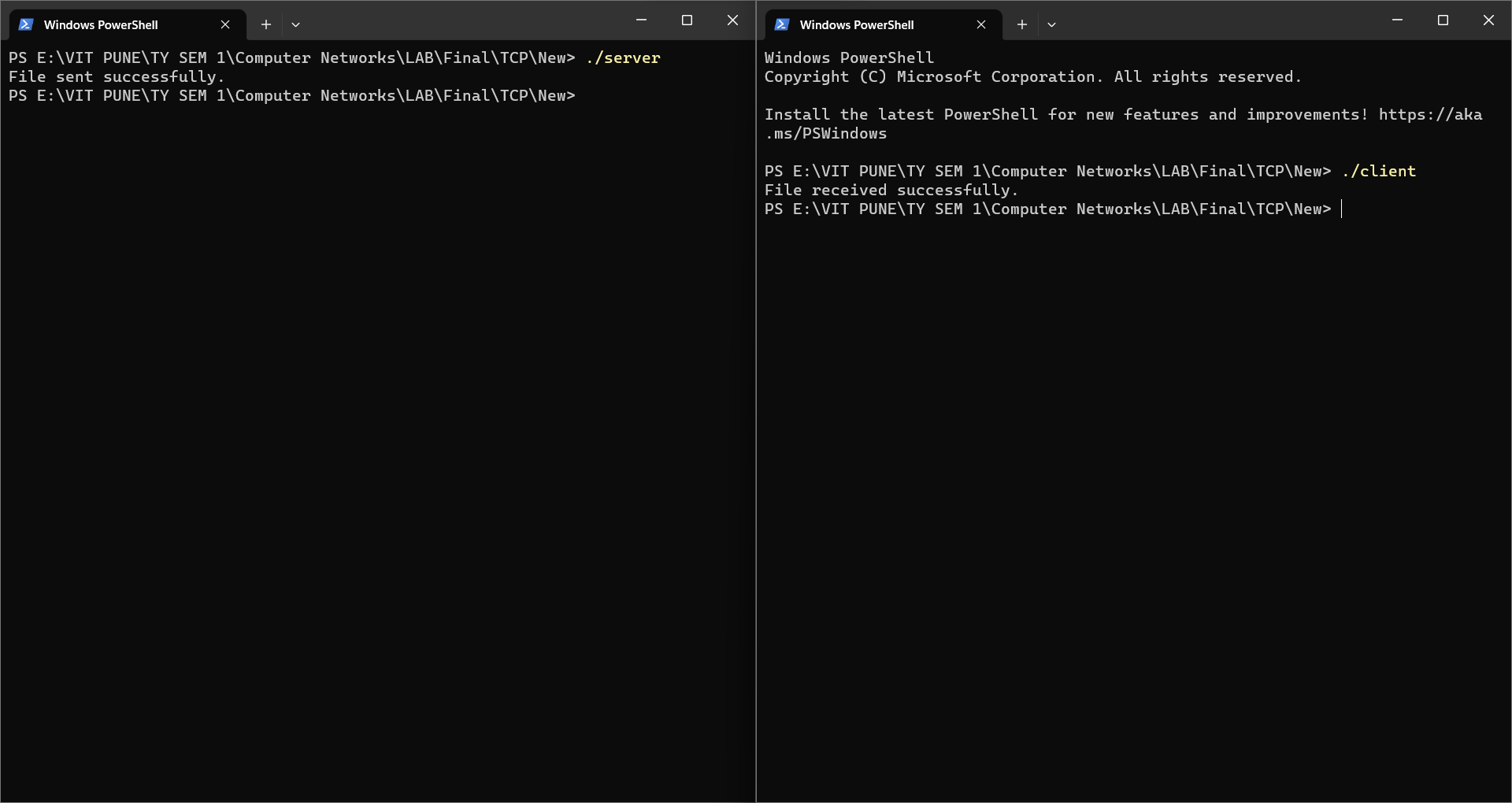
while ((n = fread(buffer, sizeof(char), BUFFER\_SIZE, fp)) > 0) {

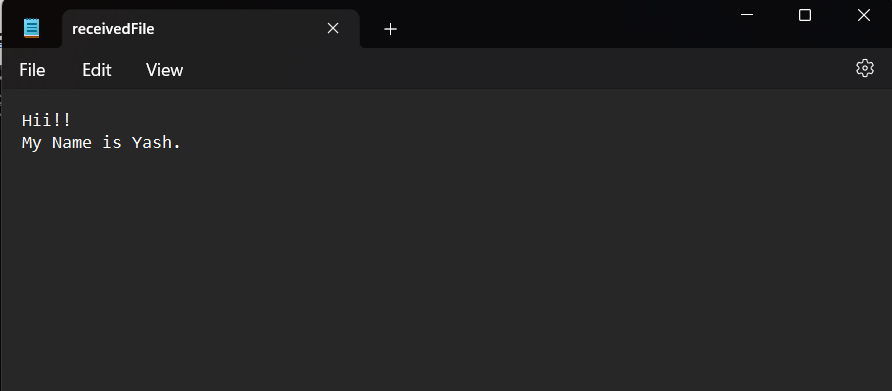
send(sockfd, buffer, n, 0);

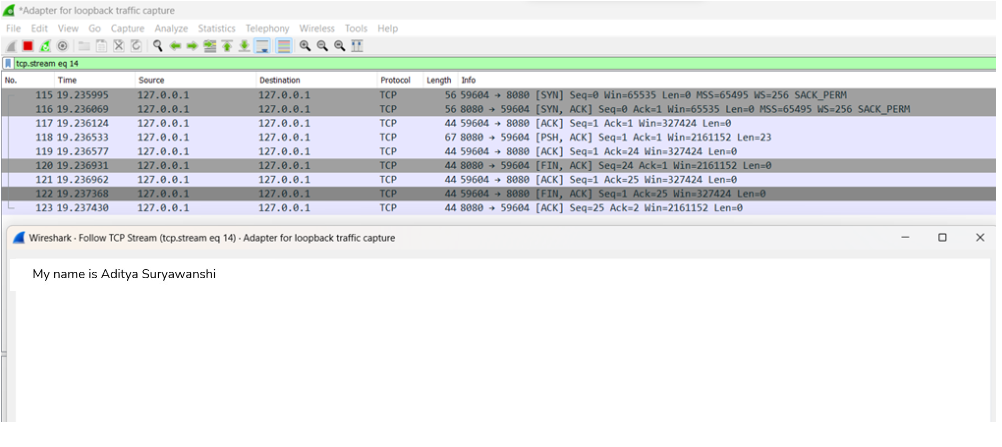
}

}

**Output :**

****

****

****