

Name : Sahil Dattatray Mohite

Div : SY-IT-A Batch : B1

Roll No. 30 PRN : 12010501

Computer Network – Lab Assignment 2

Unit : Client Server Socket Program

Question:

Write a C program to Implement Client Server Communication Using TCP/UDP Socket

Server.c

```
#include<stdio.h>
#include<stdlib.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
int main()
{
    int serversocket,client_socket,bindstatus;
    char servermessage[256]="Hello World";
    struct sockaddr_in serveraddress;

    serversocket=socket(AF_INET,SOCK_STREAM,0);
    serveraddress.sin_family=AF_INET;
    serveraddress.sin_port=htons(9000);
    //serveraddress.sin_addr.s_addr= htons("192.168.56.1");
    serveraddress.sin_addr.s_addr= INADDR_ANY;

    bindstatus=bind(serversocket,(struct sockaddr *)&serveraddress,
    sizeof(serveraddress));
    if(bindstatus<0)
    printf("binding failed\n");
    else
    printf("binding is successful\n");

    listen(serversocket,3);
    printf("Send reply to the client\n");

    //clientsocket=accept(serversocket,( struct sockaddr *)&clientsocket,
    sizeof(clientsocket));
    client_socket=accept(serversocket,NULL,NULL);

    send(client_socket,servermessage,sizeof(servermessage),0);

    close(serversocket);
    return 0;
}
```

Client.c

```
#include<stdio.h>
#include<stdlib.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>

int main()
{
    //create Socket
    int clientsocket;
    //char serverresponse[256];
    struct sockaddr_in serveraddress;

    clientsocket=socket(AF_INET,SOCK_STREAM,0);
    //SOCKET function returns the integer
    if(clientsocket<0)
        printf("socket is not created\n");
    else
        printf("socket is created succesfully\n");

    serveraddress.sin_family=AF_INET;
    serveraddress.sin_port=htons(9000);
    //serveraddress.sin_addr.s_addr= htons("192.168.56.1");
    serveraddress.sin_addr.s_addr= INADDR_ANY;

    int constatus=connect(clientsocket,(struct sockaddr *) &serveraddress,
        sizeof(serveraddress));

    if(constatus==-1)
        printf("there was an error in the connection\n");

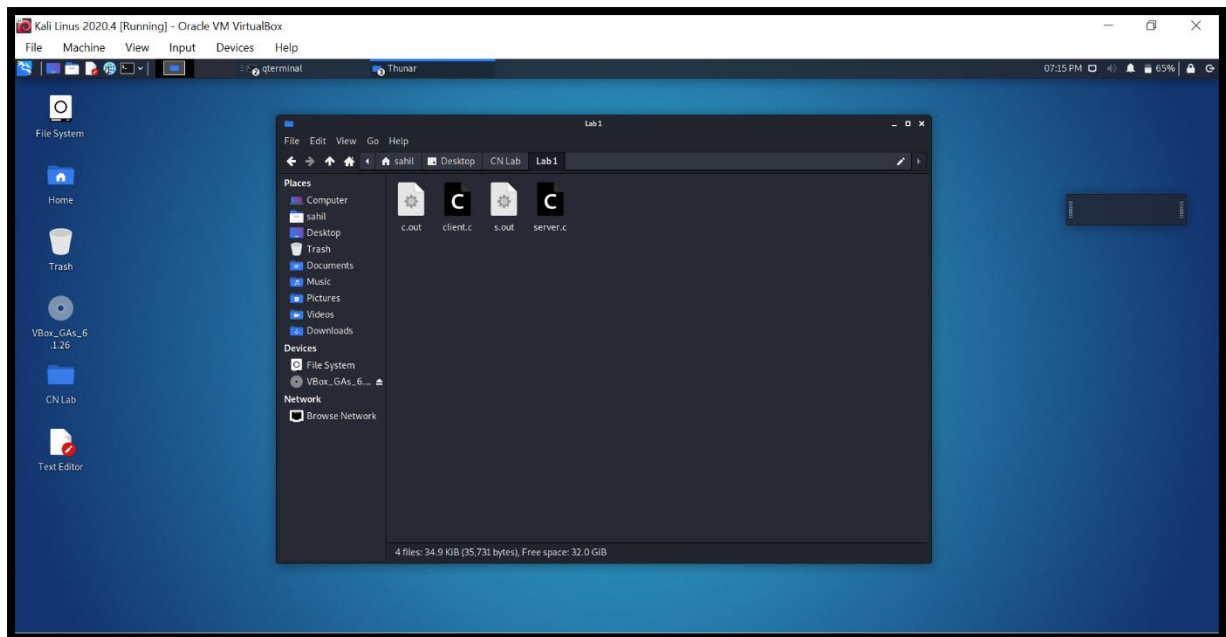
    char serverresponse[256];
    recv(clientsocket,&serverresponse,sizeof(serverresponse),0);

    printf("The reply from the server is:%s\n", serverresponse);

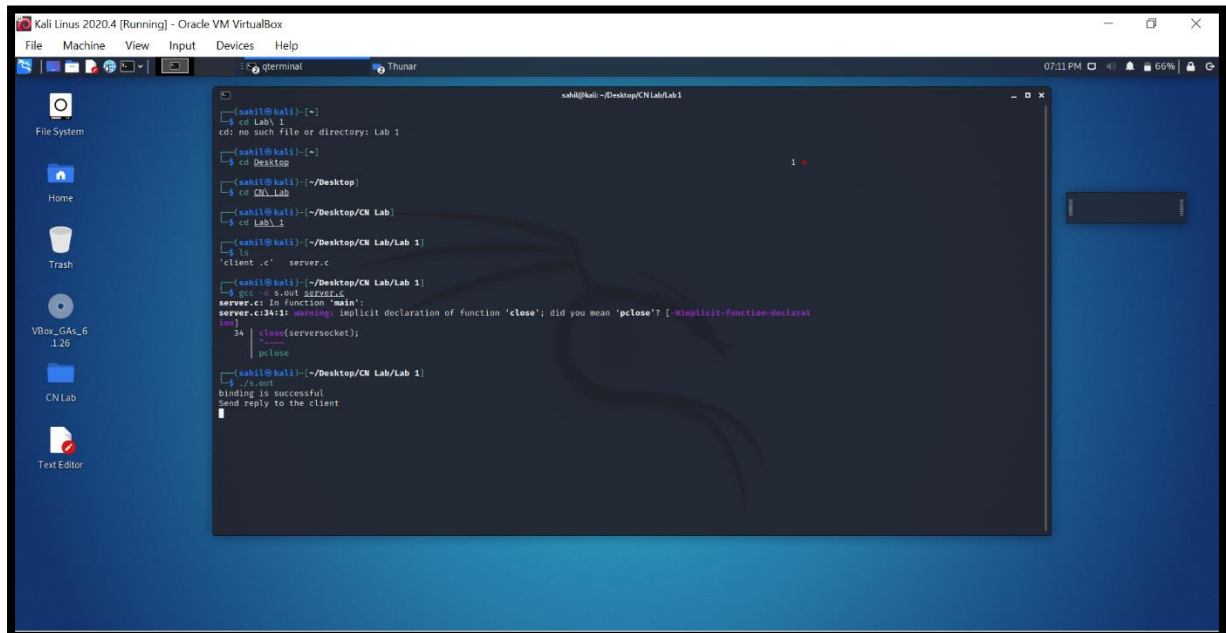
    close(clientsocket);

    return 0;
}
```

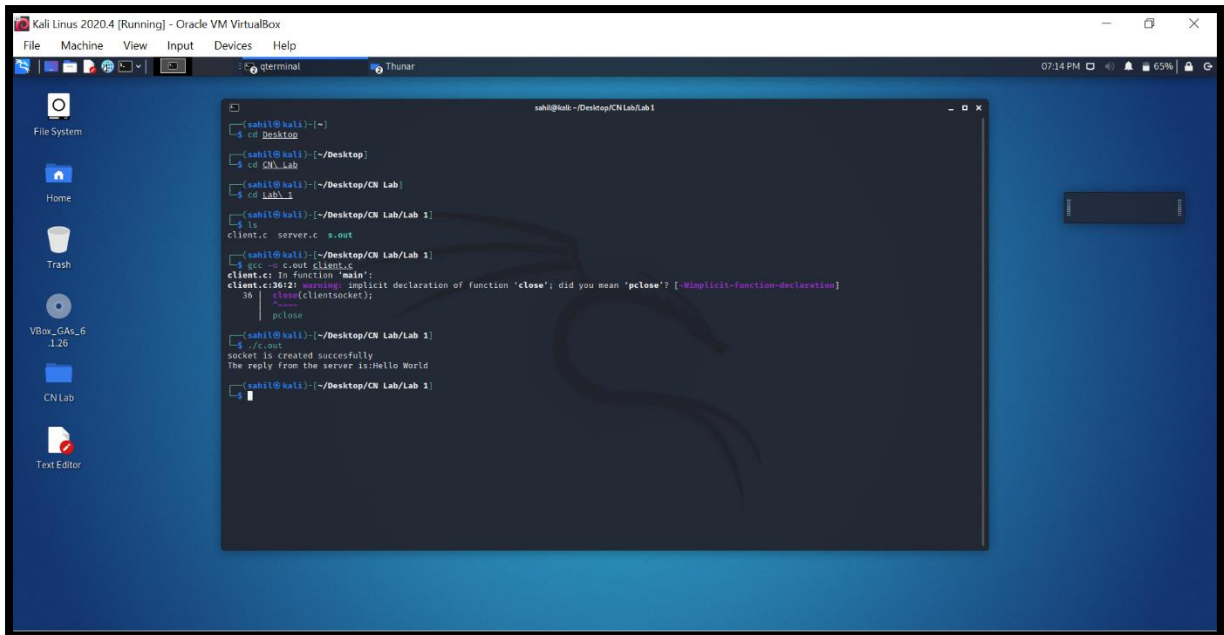
File:



Compiling and executing server.c



Compiling and executing client.c



The screenshot shows a Kali Linux 2020.4 virtual machine running in Oracle VM VirtualBox. The desktop environment includes icons for File System, Home, Trash, VBox_GAs_6, CN Lab, and Text Editor. A terminal window is open, displaying the following commands and output:

```
sahil@kali: ~/Desktop/CN Lab/Lab 1
[sahil@kali]~$ cd Desktop
[sahil@kali]~/Desktop$ cd CN Lab
[sahil@kali]~/Desktop/CN Lab$ cd Lab1
[sahil@kali]~/Desktop/CN Lab/Lab 1$ ls
client.c  server.c  s.out
[sahil@kali]~/Desktop/CN Lab/Lab 1$ gcc -o c.out client.c
client.c: In function 'main':
client.c:36:12: warning: implicit declaration of function 'close'; did you mean 'pclose'? [-Wimplicit-function-declaration]
   36 |     close(clientSocket);
      |     ~~~~~
      |     pclose

[sahil@kali]~/Desktop/CN Lab/Lab 1$ ./c.out
socket is created successfully
The reply from the server is:Hello World

[sahil@kali]~/Desktop/CN Lab/Lab 1$
```