Swetha Saseendran
CSE-C
183

Assignment 2: Echo Client

SERVER.C

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#define SA struct sockaddr
#define SERVER_PORT 4000
#define MAX 256

int error(char *msg){
        perror(msg);
        exit(1);
}

int main(int argc, char const *argv[])
{
        int sock_fd = 0, server_fd;
        struct sockaddr_in servaddr;
        int n;
        char buf[MAX];

        //CREATE SOCKET
        if ((sock_fd = socket(PF_INET,SOCK_STREAM,0)) <= 0) error("SOCKET FAILED");
```

```
        //RESET servaddr / ASSIGN IP, PORT, FAMILY  OF **SERVER**
        bzero(&servaddr, sizeof(servaddr));
        if ((inet_aton("127.0.0.1", &servaddr.sin_addr)) == 0) error("IP ERROR");
        servaddr.sin_port = htons(SERVER_PORT);
        servaddr.sin_family = PF_INET;

    //CONNECT
        if ((connect(sock_fd, (SA*)&servaddr, sizeof(servaddr))) != 0) error("CONNECT ERROR");

    //CLIENT(THIS)
    bzero(buf, MAX);
        printf("Message received from CLIENT: ");
        n=0;
        while ((buf[n++] = getchar()) != '\n')
                ;
        write(sock_fd, buf, sizeof(buf));
        bzero(buf, MAX);
        //SERVER
        read(sock_fd ,buf, sizeof(buf));
        printf("Message to the CLIENT: %s\n", buf);
        bzero(buf, MAX);


    //CLOSE SOCKET
        close(sock_fd);
        return 0;
}
```

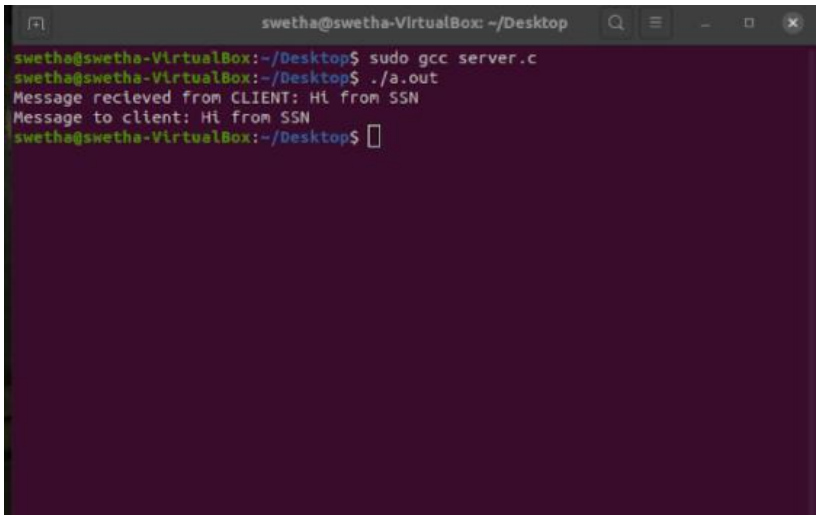
CLIENT.C

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#define SA struct sockaddr
#define SERVER_PORT 4000
#define MAX 256

int error(char *msg){
        perror(msg);
        exit(1);
}

int main(int argc, char const *argv[])
{
        int sock_fd = 0, server_fd;
        struct sockaddr_in servaddr;
        int n;
        char buf[MAX];

        //CREATE SOCKET
        if ((sock_fd = socket(PF_INET,SOCK_STREAM,0)) <= 0) error("SOCKET FAILED");

        //RESET servaddr / ASSIGN IP, PORT, FAMILY  OF **SERVER**
        bzero(&servaddr, sizeof(servaddr));
        if ((inet_aton("127.0.0.1", &servaddr.sin_addr)) == 0) error("IP ERROR");
```

```
        servaddr.sin_port = htons(SERVER_PORT);
        servaddr.sin_family = PF_INET;

    //CONNECT
        if ((connect(sock_fd, (SA*)&servaddr, sizeof(servaddr))) != 0) error("CONNECT ERROR");

    //CLIENT(THIS)
    bzero(buf, MAX);
        printf("Message from CLIENT: ");
        n=0;
        while ((buf[n++] = getchar()) != '\n')
                ;
        write(sock_fd, buf, sizeof(buf));
        bzero(buf, MAX);
        //SERVER
        read(sock_fd ,buf, sizeof(buf));
        printf("Echo received from SERVER: %s\n", buf);
        bzero(buf, MAX);


    //CLOSE SOCKET
        close(sock_fd);
        return 0;
}
```

Snapshot of the output:

Server Side:



Client Side: