Department of Computer Science and Engineering

S.G.Shivanirudh , 185001146, Semester V

30 August 2020

UCS1511 - Networks Laboratory

Exercise 1: Echo Server using TCP

Objective:

Develop a socket program to simulate Echo Server.
Echo Server: The client sends data to server. The server in turn sends the message back to the client. Send multiple lines of text.

Code:

Server:

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

```c
#include<string.h>

int main(int argc,char **argv){
    //Server and Client addresses
    struct sockaddr_in server_address, client_address;
    //Buffer to handle messages
    char buffer[1024];

    //Socket file descriptor
    int sockfd = socket(AF_INET, SOCK_STREAM, 0); //domain = IPv4, type = TCP, protocol = IP
    if(sockfd < 0)
        perror("Error: Unable to create socket");

    //Filling server_address with null bytes
    bzero(&server_address, sizeof(server_address));

    server_address.sin_family = AF_INET; // Uses the Internet address family
    server_address.sin_addr.s_addr = INADDR_ANY;// Use any of the available addresses
    server_address.sin_port = htons(4500);// Connect to specified port 4500

    //Bind socket to the specified port
    if(bind(sockfd, (struct sockaddr*)&server_address, sizeof(server_address))<0)
        perror("Bind error");

    //Look for clients to serve, with a maximum limit of 2.
    listen(sockfd, 2);

    //New socket file descriptor to handle connections.
    int len = sizeof(client_address);
    int newfd = accept(sockfd, (struct sockaddr*)&client_address, &len);

    //Read message from buffer
    read(newfd, buffer, sizeof(buffer));
    printf("\nMessage from Client: %s\n", buffer);

    //Echo message back to client
    write(newfd, buffer, sizeof(buffer));
    printf("\nMessage sent: %s\n", buffer);

```

```
    close(sockfd);
    close(newfd);
    return 0;
}
```

Client:

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

int main(int argc,char** argv){
    //Server and client addresses
    struct sockaddr_in serveraddr, clientaddr;
    //Buffer to handle messages
    char buffer[1024];

    //Server socket file descriptor
    int sockfd = socket(AF_INET, SOCK_STREAM, 0);//(domain = Ipv4, type = TCP, protocol = 0
    if(sockfd < 0)
        perror("Error: Unable to create socket");

    //Filling server address with null bytes
    bzero(&serveraddr, sizeof(serveraddr));

    serveraddr.sin_family = AF_INET;//Use the Internet address family
    serveraddr.sin_addr.s_addr = inet_addr(argv[1]);//Use ip address passed as command line argument
    serveraddr.sin_port = htons(4500);//Connect socket to port 4500

    //Attempt to connect client to socket on specified port
    connect(sockfd, (struct sockaddr*)&serveraddr, sizeof(serveraddr));

    //Sending Message
    int len = sizeof(clientaddr);

```

```
    //Write message into buffer
    printf("Enter the message: ");scanf(" %[^\n]", buffer);
    write(sockfd, buffer, sizeof(buffer));

    //Read echoed message from buffer
    read(sockfd, buffer, sizeof(buffer));
    printf("Message from Server: %s", buffer);

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

Output:

Server:

```
Message from Client: hi this is from ssn cse

Message sent: hi this is from ssn cse
```

Client:

```

Enter the message: this is from ssn cse
Message from Server: this is from ssn cse
```