

Terna Engineering College
Computer Engineering Department
Program: Sem V
Course: Computer Network Lab

Faculty: Umesh B Mantale, D V Thombre and Ramesh Shahabade

LAB Manual

PART A

(PART A: TO BE REFERRED BY STUDENTS)

Experiment No. 9

A.1 Objective:

Establishing Client Server communication using TCP socket programming.

A.2 Prerequisite:

- Knowledge about LAN, MAN and WAN and NW Elements.
- Linux NW Commands
- HW and IP Address concepts.
- Concept of Port, Socket, Localhost, Client and Server,
- Any programming language such as C, C++, Java or Python
- NW libraries.

A.3 Outcome:

After successful completion of this experiment students will be able to

- Ability to establish connection
- Ability to communicate among the PCs
- Ability to write NW related system program
- Ability to program the socket.

PART B

(PART B: TO BE COMPLETED BY STUDENTS)

(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the Blackboard or emailed to the concerned lab in charge faculties at the end of the practical in case there is no Blackboard access available)

Roll No. 50	Name: Amey Thakur
Class: TE-Comps B	Batch: B3
Date of Experiment: 05/10/2020	Date of Submission: 05/10/2020
Grade :	

B.1 Document created by the student:

(Write the answers to the questions given in section 5.1 during the 2 hours of practice in the lab here)

Input -

```
#server.py
```

```
import socket
```

```
s = socket.socket()
```

```
port = 123
```

```
s.bind(('', port))
```

```
s.listen()
```

```
while True:
```

```
    c, addr = s.accept()
```

```
    message = input("Write a message : ")
```

```
    c.send(str.encode(message))
```

```
    c.close()
```

#client.py

```
import socket

s = socket.socket()

port = 123

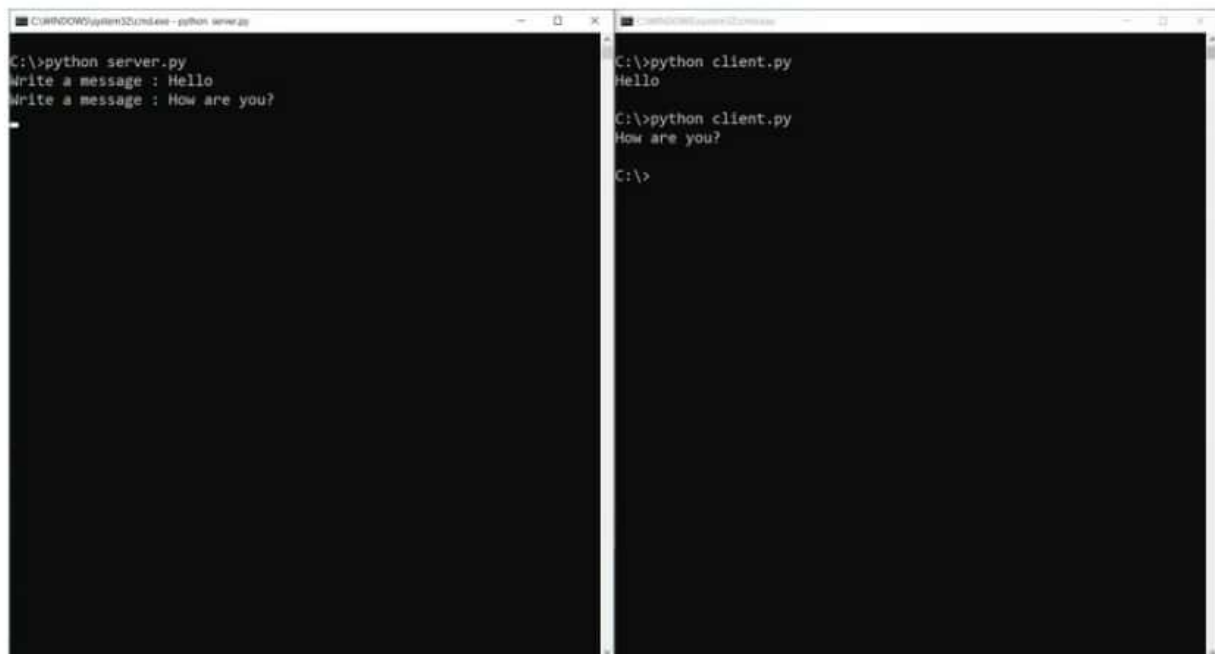
s.connect(('127.0.0.1', port))

message = s.recv(1024)

print(bytes.decode(message))

s.close()
```

Output -



```
C:\WINDOWS\system32\cmd.exe - python server.py
C:\>python server.py
Write a message : Hello
Write a message : How are you?
C:\>

C:\WINDOWS\system32\cmd.exe
C:\>python client.py
Hello
C:\>python client.py
How are you?
C:\>
```

B.3 Observations and learning:

(Students are expected to understand the selected topic. Have to list out the components & functionality. Prepare a flow of the algorithm defined in the paper. List the performance metrics that are used)

- If we are creating a connection between client and server using TCP then it has few functionalities like, TCP is suited for applications that require high reliability, and transmission time is relatively less critical.
- It is used by other protocols like HTTP, HTTPS, FTP, SMTP, Telnet. TCP rearranges data packets in the order specified. There is an absolute guarantee that the data transferred remains intact and arrives in the same order in which it was sent.
- TCP does Flow Control and requires three packets to set up a socket connection before any user data can be sent. TCP handles reliability and congestion control.
- It also does error checking and error recovery. Erroneous packets are retransmitted from the source to the destination.

B.4 Conclusion:

(Students must write the conclusion as per the attainment of individual outcome listed above and learning/observation noted in section B.3)

We learned to establish Client-Server communication using TCP socket programming.

B.5 Question of Curiosity

(To be answered by the student based on the practical performed and learning/observations)

Questions to answer:

1. What is Client? Give example
2. What is Server? Give Example.
3. What is Port? How many ports are possible? What are the well-known ports, registered ports and Dynamic or private port numbers?
4. What is Socket? And why should one program it?
5. What are the server side commands/library functions used and mention their purpose?
6. What are the Client side commands/library functions used and mention their purpose?
7. What is Remote procedure call?
8. What is connection oriented and connectionless communication? Which protocols are used?

Q1. What is client? Give example.

Ans:

- A client is a computer or program that as part of its operation, relies on sending a request to another program or computer hardware or software that accesses a service made available by a server, which may or may not be located on another computer.
- A client is a part of a client-server model which is still used today.
- Example: Web browser are clients that connect to web servers and retrieve web pages for display.

Q.2. What is server? Give example.

Ans:

- A server is a piece of computer hardware or software that provides functionality for other programs or devices called clients.
- This architecture is called as client server model.
- Servers can provide various functionalities often called services such as sharing data or resources among multiple clients or performing computation for a client.
- A single server can serve multiple clients and a client can use multiple servers.
- Typical servers are database servers, file servers, print servers, mail servers, game servers and application servers.
- Example: A user accessing a database server to retrieve some data.

Q3 What is port?

How many ports are possible?

What are the well known ports, registered ports and dynamic or private port numbers?

Ans:

- A port is a physical docking point using which an external device can be connected to the computer. It can also be programmatic docking point through which information flows from a program to the computer or over the internet.
- A network port which is provided by the transport layer protocol of Internet Protocol Suite such as (TCP) Transmission Control Protocol and (UDP) User Datagram Protocol, is a number serving end point communication between two computers.
- To determine what protocol incoming traffic should be directed to, different port numbers are used. They allow a single host with a single IP address to an network service.
- Each port number have a distinct service and for each host can have 65535 ports per IP addresses. Internet Assigned Number Authority (IANA) is responsible for managing the uses of these ports.
- There are 3 categories for ports.
 - ① 0 - 1023: Well known ports or system ports.
 - ② 1024 - 49151: Registered ports assigned by IANA to a specific service upon application by specific entity.
 - ③ 49152 - 65535: Dynamic (Private, high) which can be used by private or customer services or temporal purpose.

Q.4. What is socket? And why one should program it?

Ans:

- A socket is one endpoint of a two way communication link between two programs running on the network. The socket mechanism provides a means of interprocess communication by establishing named contact points between which the communication takes place.
- The socket provides bidirectional FIFO communication facility over a network.
- A socket connecting to the network is created at each end of the communication. Each socket has a specific address. This address is composed of an IP address and a port number.
- Sockets are generally employed in client server applications. The server creates a socket, attach it to a network port address then waits for a client to contact it. The client creates a socket and then attempts to connect to the server socket. When connection is established, transfer of data takes place.
- 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 with an IP, while other socket reaches out to the other to form a connection.

Q5. What are the server side commands / library functions used and mention their purpose?

Ans:

- In the client-server architecture, there is one centralized server that provides services and many clients receive service from that centralized server.
- The client also does the request to server. A few important server socket methods in the architecture are as follows:

① `Socket.bind()`: This method binds the address (hostname, port number) to the socket.

② `Socket.listen()`: This method basically listens to the connections made to the socket. It starts TCP listener. Backlog is an argument of this method which specifies the maximum number of queued connections. Its minimum value is zero and maximum value is 5.

③ `Socket.accept()`: This will accept TCP client connection. The pair (conn, address) is the return value pair of this method. Here conn is socket object used to send and receive data on the connection and the address is address bound to socket before this method. `Socket.bind()` and `Socket.listen()` methods must be used.

Q.6. What are the client side command/library functions used and mention their purpose?

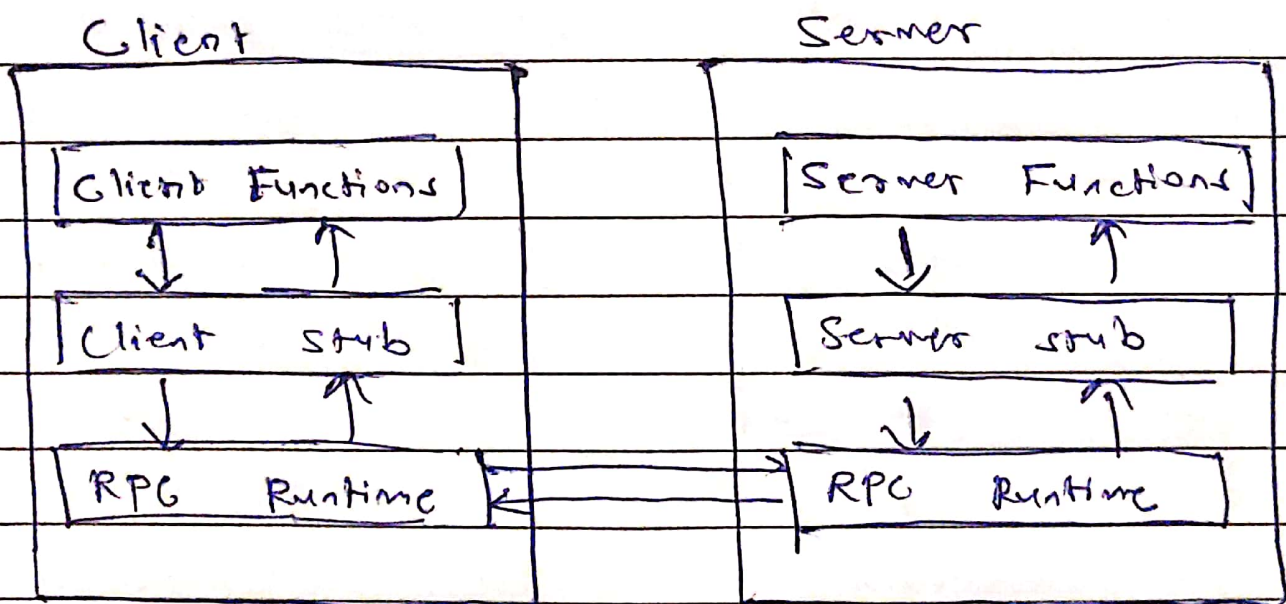
Ans:

- The client in the client server architecture requests the server and receives services from the server. For this, there is only one method dedicated for clients.
- `socket.connect (address)`: This method actively initiates server connection or in simple words this method connects the client to the server. The argument `address` represents the address of the server.

Q.7. What is Remote Call Procedure ?

Ans:

- A Remote Procedure Call is an interprocess communication technique that is used for client - server based applications.
- It is also known as a subroutine call or function call.
- A client has a request message that the RPC translates and sends to the server. This request may be a procedure or a function call to a remote server. When the server receives the request, it sends the required response back to the client. The client is blocked while the server is processing the call and only resumed execution after the server is finished.



Advantages of RPC

- ① RPC support process oriented and thread oriented models.
- ② The internal message passing mechanism of RPC is hidden from the user.
- ③ Many layers of the protocol are omitted by the RPC to improve performance.

Disadvantages of RPC

- ① RPC is a concept that can be implement in different ways
- ② There is no flexibility in RPC for hardware architecture
- ③ There is an increase in costs because of RPC.

Q.8. What is connection oriented and connectionless communication? Which protocols are used?

Ans:

- Two distinct techniques are used in data communication to transfer data.

① Connection oriented services

- In connection orient service, we have to establish a connection before starting the communication. When connection is established, we send the message or information and then we release the connection.
- This is more reliable than connectionless service.
- We can send the message in connection orient service if there is an error at receiver's end.
- Ex. TCP (Transmission Control Protocol)

② Connectionless Service

- It is similar to the postal service, as it carries the full address where the message (letter) is to be carried. Each message is routed independently from source to destination.
- The order of message sent can be different from the order received.
- In connectionless, the data is transferred in one direction from source to destination without checking that destination is still there or not. If it is prepared to accept the message.
- Authentication isn't needed.
- Ex - UDP (User Datagram Protocol)