# Lab12 : Postlab

**Name:** Divita P

**TE COMPS**

**Roll no:** 8626

## 1. What are sockets? What are different types of sockets? What are ephemeral ports?

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 inter-process communication (IPC) by establishing named contact points between which the communication takes place.

Like ‘Pipe’ is used to create pipes and sockets is created using ‘socket’ system call. The socket provides bidirectional FIFO Communication facility over the 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, attaches it to a network port address then waits for the client to contact it. The client creates a socket and then attempts to connect to the server socket. When the connection is established, transfer of data takes place.

There are four types of sockets:

**Datagram Socket :**

This is a type of network which has connectionless points for sending and receiving packets. It is similar to a mailbox. The letters (data) posted into the box are collected and delivered (transmitted) to a letterbox (receiving socket).

**Stream Socket**

In Computer operating system, a stream socket is type of interprocess communications socket or network socket which provides a connection-oriented, sequenced, and unique flow of data without record boundaries with well defined mechanisms for creating and destroying connections and for detecting errors. It is similar to a phone. A connection is established between the phones (two ends) and a conversation (transfer of data) takes place.

**Raw Sockets**

These provide users access to the underlying communication protocols, which support socket abstractions. These sockets are normally datagram oriented, though their exact characteristics are dependent on the interface provided by the protocol. Raw sockets are not intended for the general user; they have been provided mainly for those interested in developing new communication protocols, or for gaining access to some of the more cryptic facilities of an existing protocol.

**Sequenced Packet Sockets**

They are similar to a stream socket, with the exception that record boundaries are preserved. This interface is provided only as a part of the Network Systems (NS) socket abstraction, and is very important in most serious NS applications. Sequenced-packet sockets allow the user to manipulate the Sequence Packet Protocol (SPP) or Internet Datagram Protocol (IDP) headers on a packet or a group of packets, either by writing a prototype header along with whatever data is to be sent, or by specifying a default header to be used with all outgoing data, and allows the user to receive the headers on incoming packets.

An Ephemeral port is a temporary communication hub used for Internet Protocol (IP) communications. It is created from a set range of port numbers by the IP software and used as an end client's port assignment in direct communication with a well-known port used by a server.

Ephemeral means temporary or short-lived, as is the characteristic of this type of port.

## 2. Why does the bind error occur?

Socket consists of a tuple of 4 values (server ip, server port , client ip, client port). When any two sockets tuples match, error is thrown. When you terminate the code on the server side, it means you are ending connection with tcp client. Now the server is the one which sends FIN to the client and goes to TIME\_WAIT state. Typically, the TIME\_WAIT state server sends ack packets continuously to clients, assuming that if any ack gets lost in between. Time out depends on implementation of code. It could be from 30 seconds to 2 minutes or more.

Thus when you try to make a new socket connection again between server and client, because the server has not timed out yet from the previous connection, it will throw a bind error.

## 3. A client process P needs to make a TCP connection to a server process S. Consider the following situation:

## The server process S executes a socket(), a bind() and a listen() system call in that order, following which it is preempted. Subsequently, the client process P executes a socket() system call followed by connect() system call to connect to the server process S. The server process has not executed any accept() system call.

## Which one of the following events could take place? Justify your answer.

## (A) connect () system call returns successfully

## (B) connect () system call blocks

## (C) connect () system call returns an error

## (D) connect () system call results in a core dump

**Solution**:

(C) connect () system call returns an error

Explanation: All the sockets are by default in blocking mode. Since connect() is in default blocking mode then calling connect() sends SYN packet to the server. Since the server has not executed any accept() call it cannot acknowledge the SYN packet. Connect() in blocking mode keeps sending SYN packets at fixed intervals of time. This is done until an error ETIMEDOUT is returned by the TCP.