**STUDY ON SOCKET PROGRAMMING!**

***What is computer network?***  
computer network is a of connected devices that can share resources and communicate with each other.

These networks can vary in size and scope.They can be LAN(local area network) and WAN(Wide area network).

***What is a node?***

The device which are connected to network are called nodes.They can be computers,servers,smartphones,printers.

***What is a link?***

They are the communication path that connect nodes.links canbe physical or wireless.

***What are routers?***

Devicces that connect different network together and use IP addresses to forward data between networks.

***What are protocols?***

Set of rules that governs datat communication over netowork.

TCP-transfer control protocol

HTTP-HypterText Transfer protocol

FTP-File transfer Protocol

***What is socket programming?***

Socket programming is a way of connecting two nodes on a networks to communicate with one another.

SOCKET:

Socket is a endpoint for communication between two machines over a network.(A network that is connected with two devices as a link to execute two-way communication on the network).It allows for data to be sent and received between devices in a network.

The socket is a type of mechanism that is used to exchange data between different processes. Here these processes are either present in different devices or the same device which are connected over a network. Once the connection for the socket is created, then the data can be sent in both directions and continues until one of the endpoints closes the connection.

They are used to connection between a client and a server,so they can communicate with each other.

So what is client or server?

|  |  |
| --- | --- |
| CLIENT | SERVER |
| Client is a device or a program that request services resources from server. | Server is a device or program thar provides resources or services to the client. |
| Clients initiate communication and send requests to server | Server waits for incoming request from the user(client) |
| They provide UI(user Interface) for end-users to interact with services provided by server. | They handle the incoming requests and send back the appropriate response or data.They can handle multiple clients simultaneously. |

The socket address is a combination of IP address and port.

IP address?

It is the unique address assigned to each dice connected to a network that uses IP for communication.

FUNCTIONS:identifies hosts,location of host in network TYPES:IPv4,IPv6

Public vs private IP addresses?

Public IP is provided by ISP(internet service provider),they are used on the internet.

Private IP address:They are used within private networks,not routable on the internet.

RANGE:192.168.0.0 to 192.168.255.255

And 10.0.0.0 to 10.255.255.255

Port?

Port is numerical identifier in networking to specify particular process or services on a device within a network.

RANGE:0-65535

NOTE:0-1023 are reserved for well-known services and protocols

HTTP uses port 80,HTTPS uses port 443

Now,in the TCP or IP layer socket is bound as a port number which can identify whether the data is to be sent to a applicant or not.

TYPES OF SOCKETS:

There are two types of sockets TCP(transfer control protocol) AND UDP(user datagram protocol)

|  |  |
| --- | --- |
| TCP(transfer control protocol) | UDP(user datagram protocol) |
| Connection-oriented | Connectionless |
| Reliable(ensures data is received in order)Order in which data received is guaranteed. | Unreliable(no guarantee of the data itself)No guarantee of order of dtata. |
| Dtata transmission is stream based(there is continuous flow of data) | Datagram-based(individual packets)  Datagram-refers to self-conatined,independent packets of data that is sent over a network. |
| Connection setup happens inna 3WAY handshake manner.  (SYN,SYN-ACK,ACK) | No connection setup required. |
| Automatic retransmission of lost packets. | No retransmission. |
| Higher overhead due to reliability,hence slow | Lower overhead,hence fast |
| Ex:HTTPS,HTTP,FTP  USE:suitable for application where reliablity and data integrity is critical.  Ie)web browsing,email,file transfer. | Ex:DNS-Domain name system  USE: suitable when speed is critical and occasional data loss is acceptable.ie)gaming,video streaming. |

THREE-WAY HANDSHAKE:

In context of TCP

SYN:Synchronise

The client send a TCP packet with SYN flag set to server,this packet indicates that the clint wants to establish a connection.

SYN-ACK:Synchronise-Acknowledge

The server responds to the client with a TCP packet that has both SYN and ACK flag set.This packet acknowledges the receipt(clien’t SYN packet) and also contaisn SYN flag indicating server is willing to make/establish connection

ACK:Acknowledge

The client sends a final TCP packet with the ACK flag to set to the server.

This packet acknowledges the receipt of the servers’s SYN-ACK packet.

PROCEDURE IN CLIENT-SEVER COMMUNICATION:

1.socket:creates new communication endpoint

2.Bind:attach a local address to a socket

3.Listen:Announce a willingness to make/accept connections

4.connect:Actively attempt to establish connection

5.send:send some dat over the connection

6.Receive:receive some data over the connection

7.close:release the connection

STAGES FOR SREVER SOCKET CREATION:

1.Socket creation:

int sockfd = socket(domain, type, protocol)

sockfd:it’s a socket descriptor,an integer,it’s like a file handler.

Domain:it’s a integer,specifies the communication domain.

This is same as family.DOMAIN=FAMILY,AF\_prefix stands for “Address family”.The family is one of,

AF\_unix—unix protocol

AF\_INET---internet protocols

AF\_LOCAL is a POSIX standard for communication between different process on same host.

Type:this tells about the communication type

SOCK\_STREAM: TCP(reliable, connection-oriented)—stream socket  
SOCK\_DGRAM: UDP(unreliable, connectionless)—datagram socket

Protocol:protocol value for IP is 0

2.sockopt

This helps in manipulating objects for the socket reffered by file descriptor(sockfd).but this is optional,but this helps in reuse of address and port.

NOTE;this prevents error asuch as “address already in use”

int setsockopt(int sockfd, int level, int optname,  const void \*optval, socklen\_t optlen);

3.Bind

The bind system call assigns a name to a unnamed socket.

int bind(int sockfd, const struct sockaddr \*addr, socklen\_t addrlen);

sockd:it’s the socket descriptor returned from socket system call

sockaddr:it’s a pointer to protocol-specific address

socklen\_t: it’s the size of the address(this is a integer value)

**4.connect**

A client process connects a socket descriptor following the socket system call to establish a connection withba server.

int connect(int sockfd, struct sockaddr \*servaddr, int addrlen);

**5. Listen**

int listen(int sockfd, int backlog);

It puts the server socket in a passive mode, where it waits for the client to approach the server to make a connection.

Backlog:It defines the maximum length to which the queue of pending connections for sockfd may grow. If a connection request arrives when the queue is full, the client may receive an error with an indication of ECONNREFUSED.

**6. Accept**

int new\_socket= accept(int sockfd, struct sockaddr \*addr, socklen\_t \*addrlen);

It extracts the first connection request on the queue of pending connections for the listening socket, sockfd, creates a new connected socket, and returns a new file descriptor referring to that socket.

At this point, the connection is established between client and server, and they are ready to transfer data.

STAGES FOR CLIENT SOCKET CREATION: