

It is necessary to ensure that the right representation is used on each machine. Functions are used to convert from host to network form before transmission- htons for short integers and htonl for long integers.

The value for servaddr.sin\_addr is assigned using the following function

```
inet_pton(AF_INET, "IP_Address", &servaddr.sin_addr);
```

The binary value of the dotted decimal IP address is stored in the field when the function returns.

### **3. Binding of the client socket to a local port**

This is optional in the case of client and we usually do not use the bind function on the client side.

### **4. Connection of client to the server**

A server is identified by an IP address and a port number. The connection operation is used on the client side to identify and start the connection to the server.

```
int connect(int sd, struct sockaddr * addr, int addrlen);
```

**sd** – file descriptor of local socket

**addr** – pointer to protocol address of other socket

**addrlen** – length in bytes of address structure

The header files to be used are sys/types.h and sys/socket.h

It returns 0 on success and -1 in case of failure.

### **5. Reading from socket**

In the case of TCP connection reading from a socket can be done using the read system call

```
int read(int sd, char * buf, int length);
```

### **6. writing to a socket**

In the case of TCP connection writing to a socket can be done using the write system call

```
int write( int sd, char * buf, int length);
```