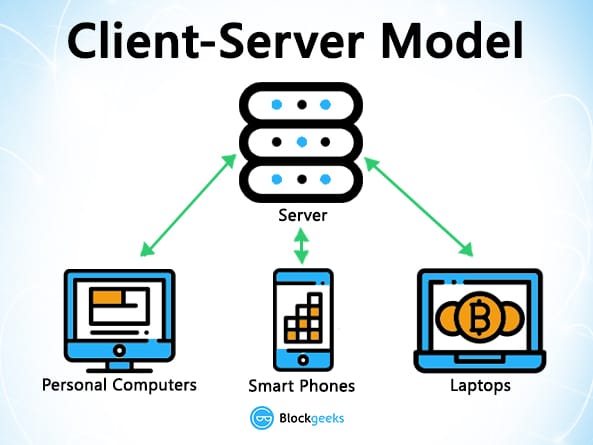
**Client/Server Model**

The client server model is distributed application framework which divides tasks between server and client which may either be on same system or communicate through computer network or internet. A client is a program that uses the services other program provides. A server the program which provides the services. A client program only runs when it request resource and services from the server while the server runs all time as it doesn't know when the services are required. A client server model follows many to one relationship as a server can provide services to multiple clients.



*Fig: Client Server model*

The client server relationship communicates in a request–response messaging pattern and must adhere to a common communications protocol, which formally defines the rules, language, and dialog patterns to be used. Client-server communication typically adheres to the TCP/IP protocol suite.

TCP protocol maintains a connection until the client and server have completed the message exchange. TCP protocol determines the best way to distribute application data into packets that networks can deliver, transfers packets to and receives packets from the network, and manages flow control and retransmission of dropped or garbled packets. IP is a connectionless protocol in which each packet traveling through the Internet is an independent unit of data unrelated to any other data units.

Example, A client can provide a form onto which a user (a person working at a data entry terminal, for example) can enter orders for a product. The client sends this order information to the server, which checks the product database and performs tasks that are needed for billing and shipping. Typically, multiple clients use a single server. For example, dozens or hundreds of clients can interact with a few servers that control database access.

**Advantages of Client-server networks:**

* Centralized: Centralized back-up is possible in client-server networks, i.e., all the data is stored in a server.
* Security: These networks are more secure as all the shared resources are centrally administered.
* Performance: The use of the dedicated server increases the speed of sharing resources. This increases the performance of the overall system.
* Scalability: We can increase the number of clients and servers separately, i.e., the new element can be added, or we can add a new node in a network at any time.

**Disadvantages of Client-Server network:**

* Traffic Congestion is a big problem in Client/Server networks. When a large number of clients send requests to the same server may cause the problem of Traffic congestion.
* It does not have a robustness of a network, i.e., when the server is down, then the client requests cannot be met.
* A client/server network is very decisive. Sometimes, regular computer hardware does not serve a certain number of clients. In such situations, specific hardware is required at the server side to complete the work.

ISO/OSI

OSI stands for open systems interconnections. It is an model developed by ISO(International Standards Organizations).It is a conceptual model that standardizes the telecommunication function of telecommunication or computer system. It aims for the interoperability of diverse communication systems with standard communication protocols. OSI is a seven layered architecture. The name of each of the seven layers are given bellow:

1. Physical layer

In physical layer bits of information are transferred from sender to receiver.

It receives data in form of data frames from Datalink layer and breaks it down into bits which are transmitted over a medium. It includes Ethernet cables, hubs ,repeater etc.

1. Datalink layer

This layer is similar to network layer in sense that it facilitates data transfer between devices on same network. It provides flow control and error control. It provides media access control (MAC).

1. Network layer

The [network layer](https://en.wikipedia.org/wiki/Network_layer) provides the functional and procedural means of transferring [packets](https://en.wikipedia.org/wiki/Network_packet) from one node to another connected in "different networks". A network is a medium to which many nodes can be connected, on which every node has an address and which permits nodes connected to it to transfer messages to other nodes connected to it by merely providing the content of a message and the address of the destination node and letting the network find the way to deliver the message to the destination node, possibly [routing](https://en.wikipedia.org/wiki/Routing) it through intermediate nodes. If the message is too large to be transmitted from one node to another on the data link layer between those nodes, the network may implement message delivery by splitting the message into several fragments at one node, sending the fragments independently, and reassembling the fragments at another node. It may, but does not need to, report delivery errors.

1. Transport layer

The [transport layer](https://en.wikipedia.org/wiki/Transport_layer) provides the functional and procedural means of transferring variable-length data sequences from a source to a destination host, while maintaining the quality of service functions.

The transport layer controls the reliability of a given link through flow control, [segmentation/desegmentation](https://en.wikipedia.org/wiki/Packet_segmentation), and error control. Some protocols are state- and connection-oriented. This means that the transport layer can keep track of the segments and retransmit those that fail delivery. The transport layer also provides the acknowledgement of the successful data transmission and sends the next data if no errors occurred. The transport layer creates segments out of the message received from the application layer. Segmentation is the process of dividing a long message into smaller messages.

1. Session layer

The [session layer](https://en.wikipedia.org/wiki/Session_layer) controls the dialogues (connections) between computers. It establishes, manages and terminates the connections between the local and remote application. It provides for [full-duplex](https://en.wikipedia.org/wiki/Duplex_(telecommunications)), [half-duplex](https://en.wikipedia.org/wiki/Half-duplex), or [simplex](https://en.wikipedia.org/wiki/Simplex_communication) operation, and establishes procedures for checkpointing, suspending, restarting, and terminating a session. In the OSI model, this layer is responsible for gracefully closing a session. This layer is also responsible for session checkpointing and recovery, which is not usually used in the Internet Protocol Suite. The session layer is commonly implemented explicitly in application environments that use [remote procedure calls](https://en.wikipedia.org/wiki/Remote_procedure_call).

1. Presentation layer

The [presentation layer](https://en.wikipedia.org/wiki/Presentation_layer) establishes context between application-layer entities, in which the application-layer entities may use different syntax and semantics if the presentation service provides a mapping between them. If a mapping is available, presentation protocol data units are encapsulated into session protocol data units and passed down the [protocol stack](https://en.wikipedia.org/wiki/Protocol_stack).

1. Application layer

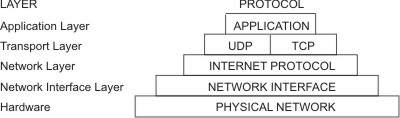
The [application layer](https://en.wikipedia.org/wiki/Application_layer) is the OSI layer closest to the end user, which means both the OSI application layer and the user interact directly with the software application. This layer interacts with software applications that implement a communicating component. Such application programs fall outside the scope of the OSI model. Application-layer functions typically include identifying communication partners, determining resource availability, and synchronizing communication. When identifying communication partners, the application layer determines the identity and availability of communication partners for an application with data to transmit.

**TCP/IP Protocol**

Protocols are sets of rules for message formats and procedures that allow machines and application programs to exchange information. These rules must be followed by each machine involved in the communication in order for the receiving host to be able to understand the message. The TCP/IP suite of protocols can be understood in terms of layers (or levels).

This figure depicts the layers of the TCP/IP protocol. From the top they are, Application Layer, Transport Layer, Network Layer, Network Interface Layer, and Hardware.

Figure 1. TCP/IP suite of protocols

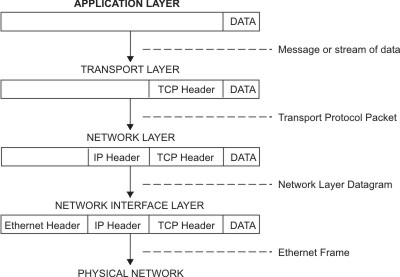


TCP/IP carefully defines how information moves from sender to receiver. First, application programs send messages or streams of data to one of the Internet Transport Layer Protocols, either the User Datagram Protocol (UDP) or the Transmission Control Protocol (TCP). These protocols receive the data from the application, divide it into smaller pieces called packets, add a destination address, and then pass the packets along to the next protocol layer, the Internet Network layer.

The Internet Network layer encloses the packet in an Internet Protocol (IP) datagram, puts in the datagram header and trailer, decides where to send the datagram (either directly to a destination or else to a gateway), and passes the datagram on to the Network Interface layer.

The Network Interface layer accepts IP datagrams and transmits them as frames over a specific network hardware, such as Ethernet or Token-Ring networks.

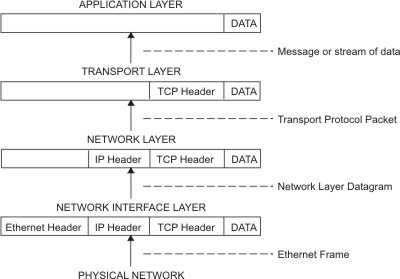
Figure 2. Movement of information from sender application to receiver host



This figure shows the flow of information down the TCP/IP protocol layers from the Sender to the Host.

Frames received by a host go through the protocol layers in reverse. Each layer strips off the corresponding header information, until the data is back at the application layer.

Figure 3. Movement of information from host to application

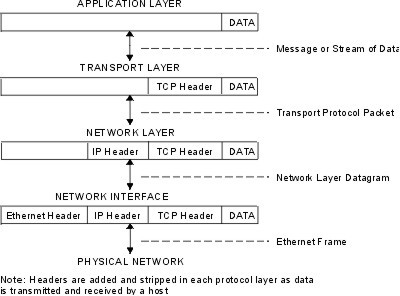


This figure shows the flow of information up the TCP/IP protocol layers from the Host to the Sender.

Frames are received by the Network Interface layer (in this case, an Ethernet adapter). The Network Interface layer strips off the Ethernet header, and sends the datagram up to the Network layer. In the Network layer, the Internet Protocol strips off the IP header and sends the packet up to the Transport layer. In the Transport layer, the TCP (in this case) strips off the TCP header and sends the data up to the Application layer.

Hosts on a network send and receive information simultaneously. [Figure 4](https://www.ibm.com/docs/en/aix/7.2?topic=protocol-tcpip-protocols#tcpip_protocols__snhjfkjdshaon34ferg) more accurately represents a host as it communicates.

Figure 4. Host data transmissions and receptions



**Unix standards(POSIX,Opengroup,IETF)**

**1)POSIX**

POSIX is an IEEE standard that acts as a standard UNIX version. It is a consortium of vendors that helps users easily port applications across different platforms. POSIX is considered a subset of UNIX and is used to cover different Unix-like environments for many other operating systems. POSIX initially contained different environments, such as Eunice for Virtual Machines, POSIX Personality, and NT from Windows OS. POSIX is portable between different variants of UNIX. In general terms, we can call POSIX as an operating system of UNIX.

**2)Open Group**

The Open Group is an international industry consortium that came into being in 1996 through a merger of the Open Software Foundation (OSF) and X/Open Company, Ltd (X/Open) for the purpose of establishing [standards](http://www.linfo.org/standards.html) in [software](http://www.linfo.org/software.html) engineering.

X/Open was a consortium that was founded in 1984 for the purposes of identifying and promoting open standards in [information](http://www.linfo.org/information.html) technology. It also managed the [UNIX](http://www.linfo.org/unix_upper.html) trademark from 1993 to 1996. OSF was established in 1988 by IBM, Hewlett-Packard, Groupe Bull, Digital Equipment Corporation, Nixdorf Computer, Siemens AG and others to create an open standard for an implementation of UNIX.

A major focus of the San Francisco, California-based Open Group is standards for application programming interfaces (APIs). An API is a set of definitions of the ways in which one piece of software communicates with another, usually between [application programs](http://www.linfo.org/appslist.html) and an [operating system](http://www.linfo.org/operating_systems_list.html).The group is perhaps best known for its publication of the Single UNIX Specification, which provides a well-accepted alternative to POSIX (Portable Operating System Interface for uniX) standards. POSIX is a set of programming [interface](http://www.linfo.org/interface.html) standards governing how to write application [source code](http://www.linfo.org/source_code.html) so that the applications are portable between operating systems.

The Open Group is also the owner of the UNIX trademark. This trademark and the certification rights for it were purchased by X/Open in 1994 from Novell, Inc, a Provo, Utah-based software company that had acquired UNIX-related assets from AT&T, the original developer of UNIX.

The Open Group also provides conformance testing, certifications and white papers (i.e., reports outlining policies or positions on major issues) usually concerning operating systems. In addition, it certifies some things that it does not specify and control itself, such as CORBA (Common Request Broker Architecture) implementations as specified by the Object Management Group (OMG) and the Linux Standard Base (LSB) from the [Free Standards Group](http://www.linfo.org/free_standards_group.html) (FSG). The OMG is a consortium aimed at setting standards in object-oriented programming.

**3)IETF**

The [Internet Engineering Task Force (IETF)](https://www.ietf.org/) is the leading Internet standards body. It develops open standards through open processes with one goal in mind: to make the Internet work better. A large open international community of network designers, operators, vendors, and researchers, the IETF focuses on the evolution of the Internet architecture and the smooth operation of the Internet. The [Internet Architecture Board (IAB)](https://www.iab.org/)  and the [Internet Research Task Force (IRTF)](https://www.irtf.org/) complement the work of the IETF by, respectively, providing long-range technical direction for Internet development and promoting research important to the Internet’s evolution.

[Open standards](https://www.internetsociety.org/issues/open-internet-standards/) are a cornerstone of the Internet. They are key to allowing devices, services, and applications to work together across a wide and dispersed network of networks. In addition to the standards themselves, the open processes and principles on which they are developed ensure the evolution of Internet technologies that meet the need of the growing number of devices and uses that empower people around the world.

The Internet Society supports the work of these groups through a variety of programs, and  provides a corporate home for the IETF LLC, the administrative entity that supports the IETF, the IAB, and the IRTF.

Network Utilities (telnet, netstat, ifconfig, ping, ftp)

• telnet

A terminal emulation program for TCP/IP networks such as the Internet. The Telnet program

runs on your computer and connects your PC to a server on the network. You can then enter

commands through the Telnet program and they will be executed as if you were entering them

directly on the server console. This enables you to control the server and communicate with

other servers on the network. To start a Telnet session, you must log in to a server by entering a

valid username and password. Telnet is a common way to remotely control Web servers.

telnet [-8] [-E] [-L] [-c] [-d] [-r] [ -e escape\_char ] [ -l user ] [-n file ] [ host [ port ] ]

• netstat

Print network connections, routing tables, interface statistics, masquerade connections, and

multicast memberships.

netstat [-a] [-b] [-e] [-f] [-n] [-o] [-p protocol] [-r] [-s] [-t] [-x] [-y] [time\_interval] [/?]

-a = This switch displays active TCP connections, TCP connections with the listening state, as well

as UDP ports that are being listened to.

-b = This netstat switch is very similar to the -o switch listed below, but instead of displaying the

PID, will display the process's actual file name. Using -b over -o might seem like it's saving you a

step or two but using it can sometimes greatly extend the time it takes netstat to fully execute.

-e = Use this switch with the netstat command to show statistics about your network connection.

This data includes bytes, unicast packets, non-unicast packets, discards, errors, and unknown

protocols received and sent since the connection was established.

-f = The -f switch will force the netstat command to display the Fully Qualified Domain

Name(FQDN) for each foreign IP addresses when possible.

-n = Use the -n switch to prevent netstat from attempting to determine host names for foreign IP

addresses. Depending on your current network connections, using this switch could considerably

reduce the time it takes for netstat to fully execute.

• ifconfig

ifconfig (short for interface configuration) is a system administration utility in Unix-like

operating systems to configure, control, and query TCP/IP network interface parameters from a

command line interface (CLI) or in system configuration scripts.

ifconfig [interface]

ifconfig interface [aftype] options | address ...

Options

interface

The name of the interface. This is usually a driver name followed by a unit number, for

example eth0 for the first Ethernet interface.

up

This flag causes the interface to be activated. It is implicitly specified if an address is assigned to

the interface.

down

This flag causes the driver for this interface to be shut down.

... ... ...

E.g.

ifconfig eth0 : View the network settings on the first Ethernet adapter installed in the computer.

• ping

Short for Packet InterNet Groper, ping is a utility used to verify if a network data packet is

capable of being distributed to an address without errors. The ping utility is commonly used to

check for network errors. It sends ICMP ECHO\_REQUEST packets to network hosts.

Syntax

ping -s [-d] [-l] [-L] [-n] [-r] [-R] [-v] [ -i interface\_address ] [-I interval] [-t ttl] host [packetsize] [count]