

## Chapter Five

### The OSI and TCP/IP communication models

#### 5.1 Open Systems Interconnection (OSI) model

Established in 1947, the International Standards Organization (ISO) is a multinational body dedicated to worldwide agreement on international standards. An ISO standard that covers all aspects of network communications is the Open Systems Interconnection model. It was first introduced in the late 1970s. An open system is a set of protocols that allows any two different systems to communicate regardless of their underlying architecture. The purpose of the OSI model is to show how to facilitate communication between different systems without requiring changes to the logic of the underlying hardware and software. The OSI model is not a protocol; it is a model for understanding and designing a network architecture that is flexible, robust, and interoperable.

The OSI model is a layered framework for the design of network systems that allows communication between all types of computer systems. It consists of seven separate but related layers, each of which defines a part of the process of moving information across a network. An understanding of the fundamentals of the OSI model provides a solid basis for exploring data communications.

##### 5.1.1 Layered Architecture

The OSI model is composed of seven ordered layers: physical (layer 1), data link (layer 2), network (layer 3), transport (layer 4), session (layer 5), presentation (layer 6), and application (layer 7).

The following figure shows the layers involved when a message is sent from device A to device B. As the message travels from A to B, it may pass through many intermediate nodes. These intermediate nodes usually involve only the first three layers of the OSI model. In this case the intermediate node is a router.

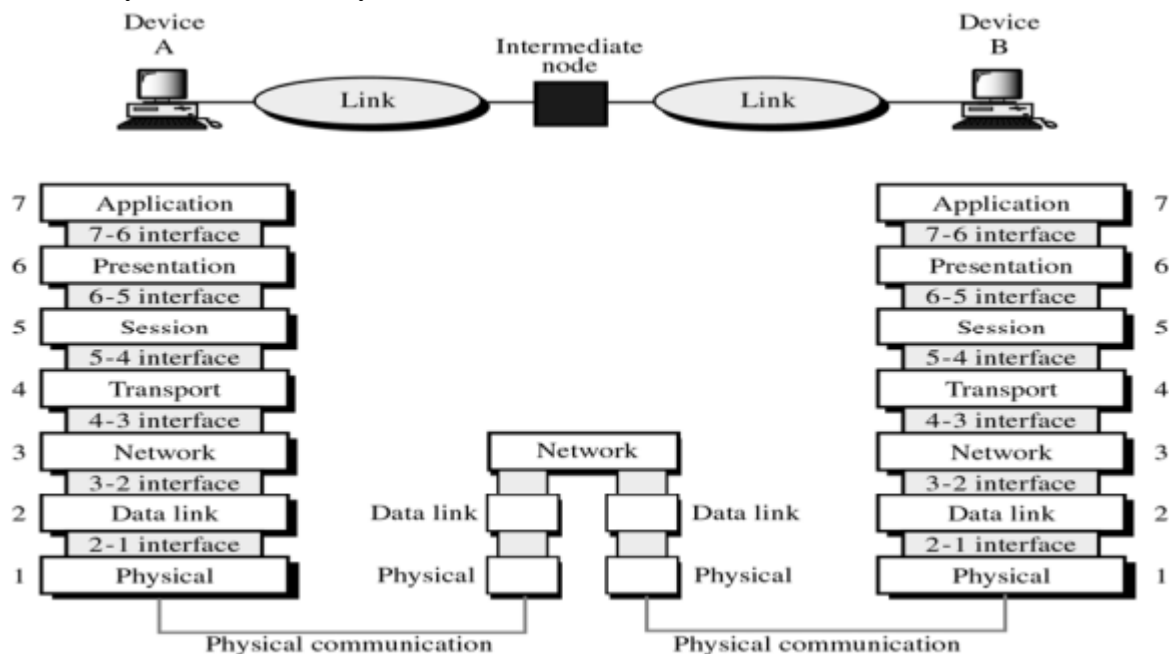


Figure 5.1 Sending message from device A to device B

In developing the model, the designers distilled the process of transmitting data to its most fundamental elements. They identified which networking functions had related uses and collected those functions into discrete groups that became the layers. Each layer defines a family of functions distinct from those of the other layers. By defining and localizing functionality in this fashion, the designers created an architecture that is both comprehensive and flexible. Most importantly, the OSI model allows complete interoperability between otherwise incompatible systems.

Within a single machine, each layer calls upon the services of the layer just below it. Layer 3, for

example, uses the services provided by layer 2 and provides services for layer 4. Between machines, layer  $x$  on one machine communicates with layer  $x$  on another machine. This communication is governed by an agreed-upon series of rules and conventions called protocols. The processes on each machine that communicate at a given layer are called peer-to-peer processes. Communication between machines is therefore a peer-to-peer process using the protocols appropriate to a given layer.

### 5.1.2 Organization of the Layers

The seven layers can be thought of as belonging to three subgroups. Layers 1, 2, and 3-physical, data link, and network-are the network support layers; they deal with the physical aspects of moving data from one device to another (such as electrical specifications, physical connections, physical addressing, and transport timing and reliability). Layers 5, 6, and 7-session, presentation, and application-can be thought of as the user support layers; they allow interoperability among unrelated software systems. Layer 4, the transport layer, links the two subgroups and ensures that what the lower layers have transmitted is in a form that the upper layers can use. The upper OSI layers are almost always implemented in software; lower layers are a combination of hardware and software, except for the physical layer, which is mostly hardware.

At each layer, a **header**, or possibly a **trailer**, can be added to the data unit. Commonly, the trailer is added only at layer 2. When the formatted data unit passes through the physical layer (layer 1), it is changed into an electromagnetic or optical signal and transported along a physical link.

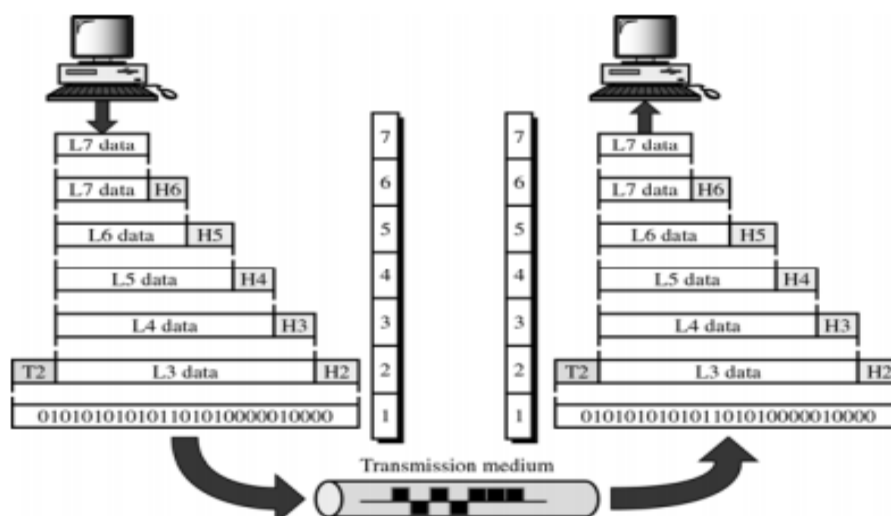


Figure 5.2 Headers and Trailer

### Encapsulation

Figure 5.2 reveals another aspect of data communications in the OSI model: encapsulation. A packet (header and data) at level 7 is encapsulated in a packet at level 6. The whole packet at level 6 is encapsulated in a packet at level 5, and so on.

In other words, the data portion of a packet at level  $N - 1$  carries the whole packet (data and header and maybe trailer) from level  $N$ . The concept is called *encapsulation*; level  $N - 1$  is not aware of which part of the encapsulated packet is data and which part is the header or trailer. For level  $N - 1$ , the whole packet coming from level  $N$  is treated as one integral unit.

### 5.1.3 Layers in the OSI model

#### i. Physical Layer

The physical layer coordinates the functions required to carry a bit stream over a physical medium. It deals with the mechanical and electrical specifications of the interface and transmission medium. It also defines the procedures and functions that physical devices and interfaces have to perform for transmission to occur.

Figure 5.3 shows the position of the physical layer with respect to the transmission medium and the data link layer.

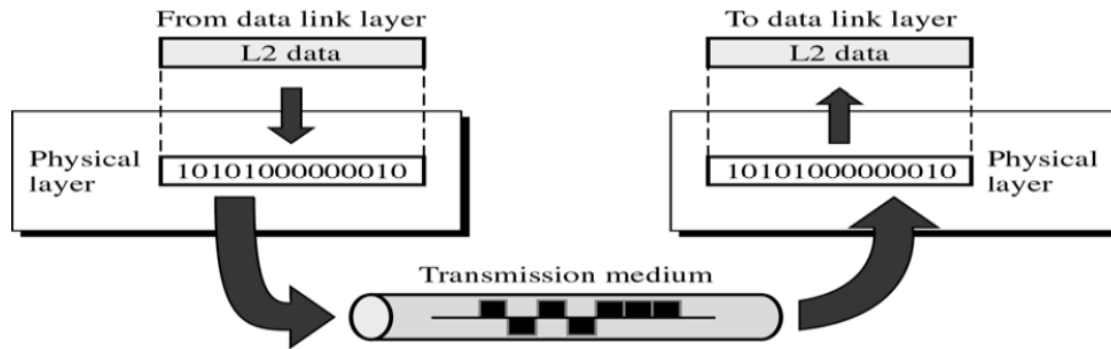


Figure 5.3 Physical Layer

The physical layer is responsible for movements of individual bits from one hop (node) to the next.

The physical layer is also concerned with the following:

- Physical characteristics of interfaces and medium: The physical layer defines the characteristics of the interface between the devices and the transmission medium. It also defines the type of transmission medium.
- Representation of bits: The physical layer data consists of a stream of bits (sequence of 0s or 1s) with no interpretation. To be transmitted, bits must be encoded into signals-electrical or optical. The physical layer defines the type of encoding (how 0s and 1s are changed to signals).
- Data rate: The transmission rate-the number of bits sent each second-is also defined by the physical layer. In other words, the physical layer defines the duration of a bit, which is how long it lasts.
- Synchronization of bits: The sender and receiver not only must use the same bit rate but also must be synchronized at the bit level. In other words, the sender and the receiver clocks must be synchronized.
- Line configuration: The physical layer is concerned with the connection of devices to the media. In a point-to-point configuration, two devices are connected through a dedicated link. In a multipoint configuration, a link is shared among several devices.
- Physical topology: The physical topology defines how devices are connected to make a network. Devices can be connected by using a mesh topology, a star topology, a ring topology, a bus topology, or a hybrid topology.
- Transmission mode: The physical layer also defines the direction of transmission between two devices: simplex, half-duplex, or full-duplex.

## ii. Data Link Layer

The data link layer transforms the physical layer, a raw transmission facility, to a reliable link. It makes the physical layer appear error-free to the upper layer (network layer). Figure 5.4 shows the relationship of the data link layer to the network and physical layers.

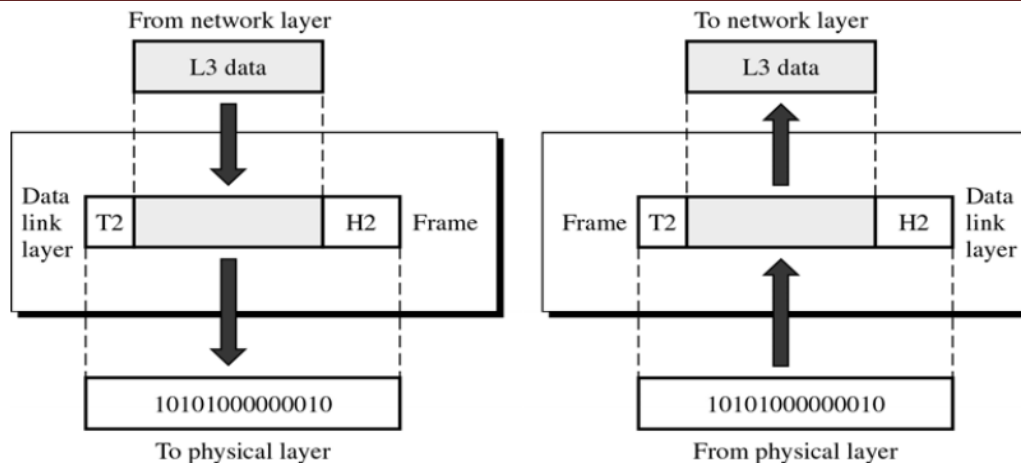


Figure 5.4 Data link layer

The data link layer is responsible for moving frames from one hop (node) to the next.

Other responsibilities of the data link layer include the following:

- Framing: The data link layer divides the stream of bits received from the network layer into manageable data units called frames.
- Physical addressing: If frames are to be distributed to different systems on the network, the data link layer adds a header to the frame to define the sender and/or receiver of the frame. If the frame is intended for a system outside the sender's network, the receiver address is the address of the device that connects the network to the next one.
- Flow control: If the rate at which the data are absorbed by the receiver is less than the rate at which data are produced in the sender, the data link layer imposes a flow control mechanism to avoid overwhelming the receiver.
- Error control: The data link layer adds reliability to the physical layer by adding mechanisms to detect and retransmit damaged or lost frames. It also uses a mechanism to recognize duplicate frames. Error control is normally achieved through a trailer added to the end of the frame.
- Access control: When two or more devices are connected to the same link, data link layer protocols are necessary to determine which device has control over the link at any given time.

### iii. Network Layer

The network layer is responsible for the source-to-destination delivery of a packet, possibly across multiple networks (links). Whereas the data link layer oversees the delivery of the packet between two systems on the same network (links), the network layer ensures that each packet gets from its point of origin to its final destination.

If two systems are connected to the same link, there is usually no need for a network layer. However, if the two systems are attached to different networks (links) with connecting devices between the networks (links), there is often a need for the network layer to accomplish source-to-destination delivery. Figure 5.5 shows the relationship of the network layer to the data link and transport layers.

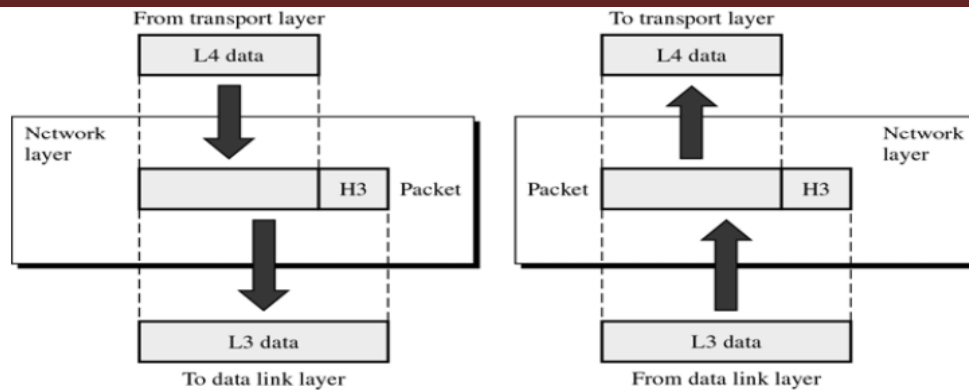


Figure 5.5 Network Layer

The network layer is responsible for the delivery of individual packets from the source host to the destination host.

*Other responsibilities of the network layer include the following:*

- Logical addressing: The physical addressing implemented by the data link layer handles the addressing problem locally. If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems. The network layer adds a header to the packet coming from the upper layer that, among other things, includes the logical addresses of the sender and receiver.
- Routing: When independent networks or links are connected to create *internetworks* (network of networks) or a large network, the connecting devices (called *routers* or *switches*) route or switch the packets to their final destination. One of the functions of the network layer is to provide this mechanism.

#### iv. Transport Layer

The transport layer is responsible for process-to-process delivery of the entire message. A process is an application program running on a host. Whereas the network layer oversees source-to-destination delivery of individual packets, it does not recognize any relationship between those packets. It treats each one independently, as though each piece belonged to a separate message, whether or not it does. The transport layer, on the other hand, ensures that the whole message arrives intact and in order, overseeing both error control and flow control at the source-to-destination level. Figure 5.6 shows the relationship of the transport layer to the network and session layers.

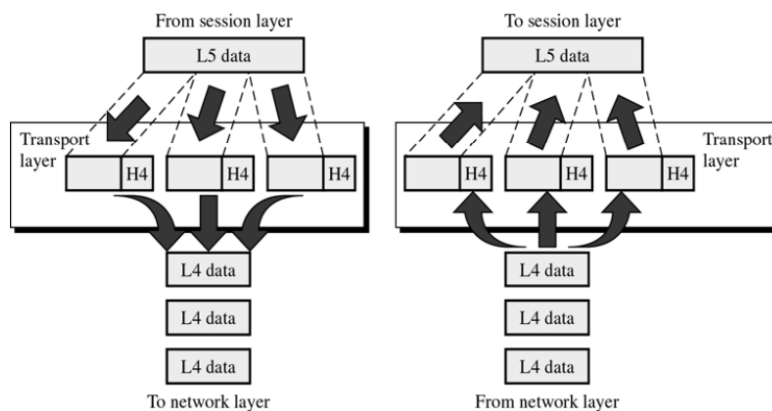


Figure 5.6 Transport Layer

The transport layer is responsible for the delivery of a message from one process to another.

Other responsibilities of the transport layer include the following:

- Service-point addressing: Computers often run several programs at the same time. For this reason, source-to-destination delivery means delivery not only from one computer to the next but also from a specific process (running program) on one computer to a specific process (running program) on the other. The transport layer header must therefore include a type of address called a *service-point address* (or port address). The network layer gets each packet to the correct computer; the transport layer gets the entire message to the correct process on that computer.
- Segmentation and reassembly: A message is divided into transmittable segments, with each segment containing a sequence number. These numbers enable the transport layer to reassemble the message correctly upon arriving at the destination and to identify and replace packets that were lost in transmission.
- Connection control: The transport layer can be either connectionless or connection-oriented. A connectionless transport layer treats each segment as an independent packet and delivers it to the transport layer at the destination machine. A connection-oriented transport layer makes a connection with the transport layer at the destination machine first before delivering the packets. After all the data are transferred, the connection is terminated.
- Flow control: Like the data link layer, the transport layer is responsible for flow control. However, flow control at this layer is performed end to end rather than across a single link.
- Error control: Like the data link layer, the transport layer is responsible for error control. However, error control at this layer is performed process-to-process rather than across a single link. The sending transport layer makes sure that the entire message arrives at the receiving transport layer without error (damage, loss, or duplication). Error correction is usually achieved through retransmission.

#### v. Session Layer

The services provided by the first three layers (physical, data link, and network) are not sufficient for some processes. The session layer is the network *dialog controller*. It establishes, maintains, and synchronizes the interaction among communicating systems.

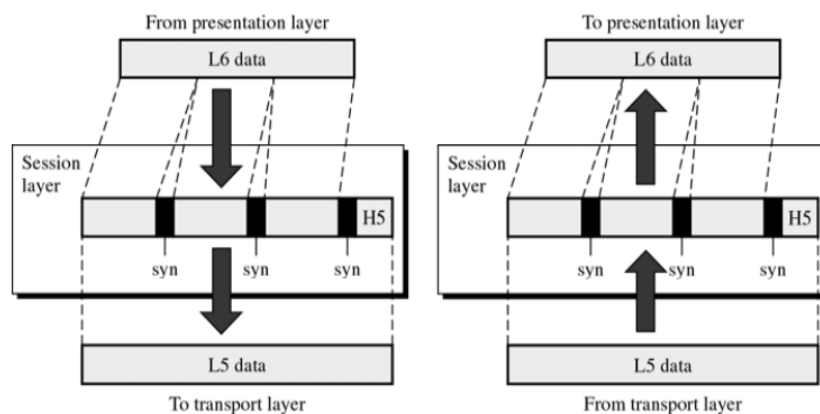


Figure 5.7 Session Layer

The session layer is responsible for dialog control and synchronization.

Specific responsibilities of the session layer include the following:

- Dialog control: The session layer allows two systems to enter into a dialog. It allows the communication between two processes to take place in either half-duplex (one way at a time) or full-duplex (two ways at a time) mode.
- Synchronization: The session layer allows a process to add checkpoints, or synchronization points, to a stream of data. For example, if a system is sending a file of 2000 pages, it is advisable to insert checkpoints after every 100 pages to ensure that each 100-page unit is received and acknowledged

independently. In this case, if a crash happens during the transmission of page 523, the only pages that need to be resent after system recovery are pages 501 to 523. Pages previous to 501 need not be resent.

### vi. Presentation Layer

The presentation layer is concerned with the syntax and semantics of the information exchanged between two systems. Figure 5.8 shows the relationship between the presentation layer and the application and session layers.

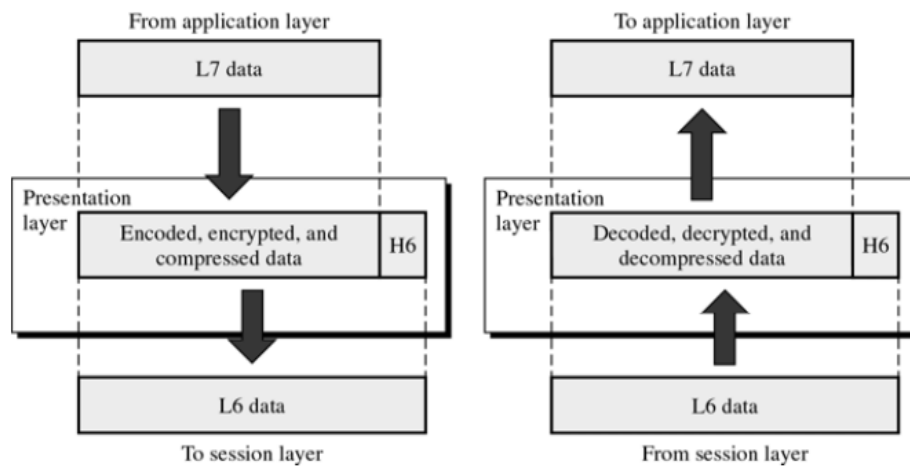


Figure 5.8 Presentation Layer

The presentation layer is responsible for translation, compression, and encryption.

Specific responsibilities of the presentation layer include the following:

- Translation: The processes (running programs) in two systems are usually exchanging information in the form of character strings, numbers, and so on. The information must be changed to bit streams before being transmitted. Because different computers use different encoding systems, the presentation layer is responsible for interoperability between these different encoding methods. The presentation layer at the sender changes the information from its sender-dependent format into a common format. The presentation layer at the receiving machine changes the common format into its receiver-dependent format.
- Encryption: To carry sensitive information, a system must be able to ensure privacy. Encryption means that the sender transforms the original information to another form and sends the resulting message out over the network. Decryption reverses the original process to transform the message back to its original form.
- Compression: Data compression reduces the number of bits contained in the information. Data compression becomes particularly important in the transmission of multimedia such as text, audio, and video.

### vii. Application Layer

The application layer enables the user, whether human or software, to access the network. It provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared database management, and other types of distributed information services.

Figure 5.9 shows the relationship of the application layer to the user and the presentation layer. Of the many application services available, the figure shows only three: X.400 (message-handling services), X.500 (directory services), and file transfer, access, and management (FTAM). The user in this example employs X.400 to send an e-mail message.



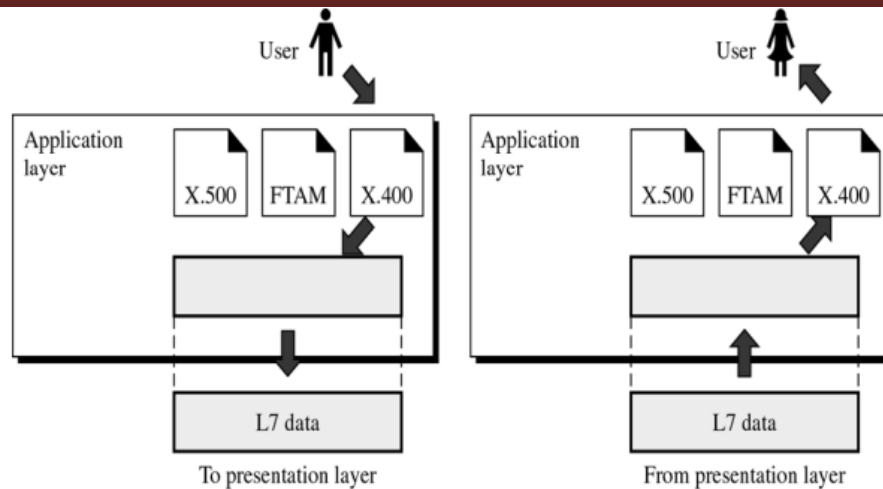


Figure 5.9 Application Layer

The application layer is responsible for providing services to the user.

Specific services provided by the application layer include the following:

- Network virtual terminal: A network virtual terminal is a software version of a physical terminal, and it allows a user to log on to a remote host. To do so, the application creates a software emulation of a terminal at the remote host. The user's computer talks to the software terminal which, in turn, talks to the host, and vice versa. The remote host believes it is communicating with one of its own terminals and allows the user to log on.
- File transfer, access, and management: This application allows a user to access files in a remote host (to make changes or read data), to retrieve files from a remote computer for use in the local computer, and to manage or control files in a remote computer locally.
- Mail services: This application provides the basis for e-mail forwarding and storage.
- Directory services: This application provides distributed database sources and access for global information about various objects and services.

## 5.2 TCP/IP PROTOCOL SUITE

The TCP/IP protocol suite was developed prior to the OSI model. Therefore, the layers in the TCP/IP protocol suite do not exactly match those in the OSI model. The original TCP/IP protocol suite was defined as having four layers: host-to-network (Hardware), internet, transport, and application. However, when TCP/IP is compared to OSI, we can say that the host-to-network layer is equivalent to the combination of the physical and data link layers. The internet layer is equivalent to the network layer, and the application layer is roughly doing the job of the session, presentation, and application layers with the transport layer in TCP/IP taking care of part of the duties of the session layer. So in this course, we assume that the TCP/IP protocol suite is made of five layers: physical, data link, network, transport, and application. The first four layers provide physical standards, network interfaces, internetworking, and transport functions that correspond to the first four layers of the OSI model. The three top most layers in the OSI model, however, are represented in TCP/IP by a single layer called the application layer.



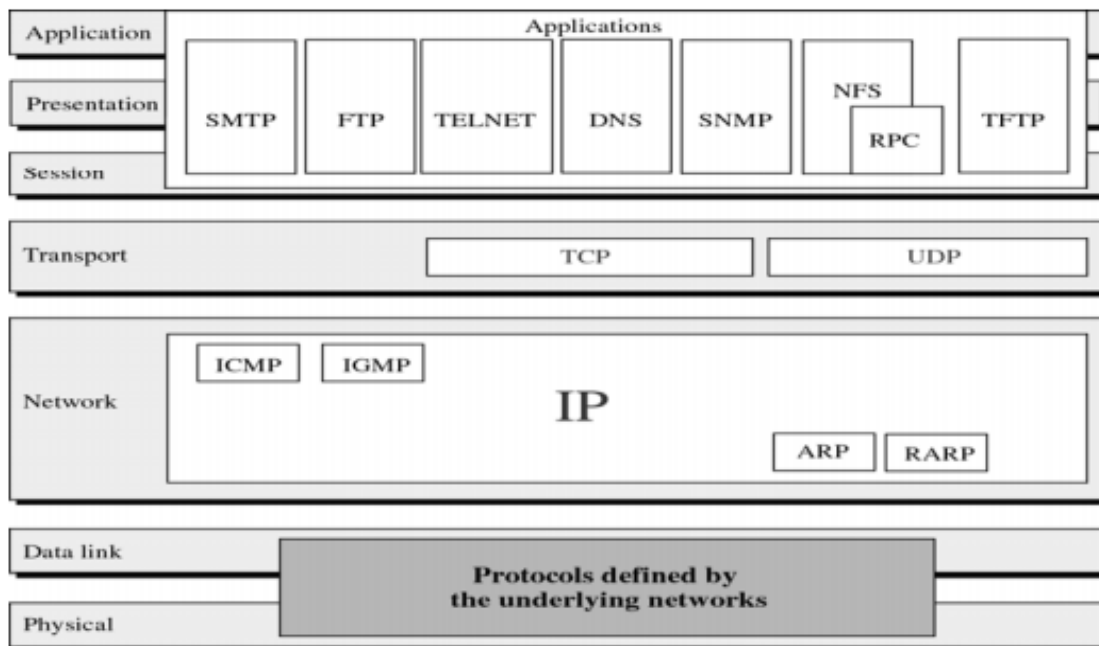


Figure 5.10 TCP/IP Protocol suites

TCP/IP is a hierarchical protocol made up of interactive modules, each of which provides a specific functionality; however, the modules are not necessarily interdependent. Whereas the OSI model specifies which functions belong to each of its layers, the layers of the *TCP/IP* protocol suite contain relatively independent protocols that can be mixed and matched depending on the needs of the system. The term *hierarchical* means that each upper-level protocol is supported by one or more lower-level protocols.

At the transport layer, *TCP/IP* defines three protocols: Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Stream Control Transmission Protocol (SCTP). At the network layer, the main protocol defined by TCP/IP is the Internetworking Protocol (IP); there are also some other protocols that support data movement in this layer. Now, let us see a brief description of each Layers of TCP/IP model.

### 1. Physical and Data Link Layers

At the physical and data link layers, *TCP/IP* does not define any specific protocol. It supports all the standard and proprietary protocols. A network in a *TCP/IP* internetwork can be a local-area network or a wide-area network.

### 2. Network Layer

At the network layer (or, more accurately, the internetwork layer), *TCP/IP* supports the Internetworking Protocol. IP, in turn, uses four supporting protocols: ARP, RARP, ICMP, and IGMP. Each of these protocols is described in the previous chapter (Chapter Four).

#### *Internetworking Protocol (IP)*

The Internetworking Protocol (IP) is the transmission mechanism used by the TCP/IP protocols. It is an unreliable and connectionless protocol—a best-effort delivery service. The term *best effort* means that IP provides no error checking or tracking. IP assumes the unreliability of the underlying layers and does its best to get a transmission through to its destination, but with no guarantees.

IP transports data in packets called *datagrams*, each of which is transported separately. Datagrams can travel along different routes and can arrive out of sequence or be duplicated. IP does not keep track of the routes and has no facility for reordering datagrams once they arrive at their destination.

The limited functionality of IP should not be considered a weakness, however. IP provides bare-bones transmission functions that free the user to add only those facilities necessary for a given application and

thereby allows for maximum efficiency.

### 3. Transport Layer

Traditionally the transport layer was represented in *TCP/IP* by two protocols: TCP and UDP. IP is a host-to-host protocol, meaning that it can deliver a packet from one physical device to another. UDP and TCP are transport level protocols responsible for delivery of a message from a process (running program) to another process. A new transport layer protocol, SCTP, has been devised to meet the needs of some newer applications.

#### *User Datagram Protocol*

The User Datagram Protocol (UDP) is the simpler of the two standard TCP/IP transport protocols. It is a process-to-process protocol that adds only port addresses, checksum error control, and length information to the data from the upper layer.

#### *Transmission Control Protocol*

The Transmission Control Protocol (TCP) provides full transport-layer services to applications. TCP is a reliable stream transport protocol. The term *stream*, in this context, means connection-oriented: A connection must be established between both ends of a transmission before either can transmit data.

At the sending end of each transmission, TCP divides a stream of data into smaller units called *segments*. Each segment includes a sequence number for reordering after receipt, together with an acknowledgment number for the segments received. Segments are carried across the internet inside of IP datagrams. At the receiving end, TCP collects each datagram as it comes in and reorders the transmission based on sequence numbers.

#### *Stream Control Transmission Protocol*

The Stream Control Transmission Protocol (SCTP) provides support for newer applications such as voice over the Internet. It is a transport layer protocol that combines the best features of UDP and TCP.

### 4. Application Layer

The *application layer* in TCP/IP is equivalent to the combined session, presentation, and application layers in the OSI model. Many protocols are defined at this layer.

### Further reading assignment

1. Critique / lacks of OSI model
2. How transport layer of OSI model manages flow control?
3. Difference between TCP and UDP connection
4. What is a Protocol in data communication? Type of protocols at Data Link layer
5. What is a Protocol Data Unit (PDU)? List name of PDU at each layer of OSI reference model
6. What are standards and type of standards on data communication and computer network? List some organizations that are authorized / responsible to set standards on computer network.