University of Thi-Qar
College of Education for Pure Science
Department of Computer Science

TCP/IP PROTOCOL SUITE

Dr. Ali Basim Al-Khafaji

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, 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.
- However, when TCP/IP is compared to OSI, we can say 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 topmost layers in the OSI model, however, are represented in TCP/IP by a single layer called the application layer (see Figure 2.16).

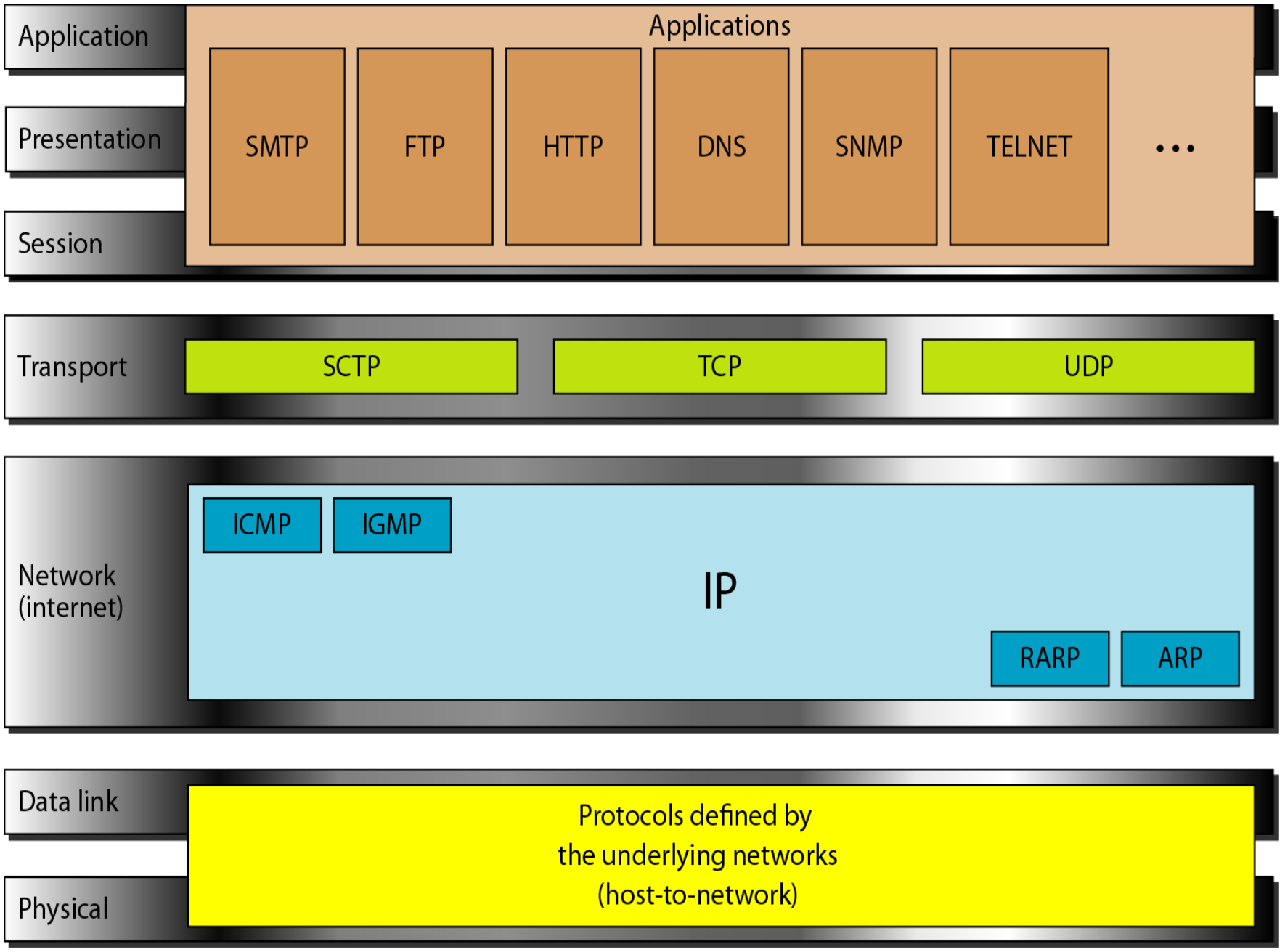

Figure (2.16) TCP/IP and OSI model

TCP/IP PROTOCOL SUITE

- 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.

Layer 1: Physical Layer

The physical layer defines the mechanical, electrical, optical, radio, procedural, and functional standards to enable the transmission of data-link (Layer 2) data frames over a certain transmission medium. The physical layer is defined in TCP/IP by the IEEE 802.1X (Ethernet) standard.

Some Layer 1 protocols

* ADSL (Asymmetric digital subscriber line).
* ISDN (Integrated Services Digital Network).
* FDDI (Fiber Distributed Data Interface) is a set of ANSI and ISO standards for data transmission on fiber optic lines in a local area network (LAN).
* RS-232, a serial line interface originally developed to connect modems and computer terminals.
* SDH (Synchronous Digital Hierarchy).

Layer 2: Data Link Layer

The data link layer transmits data on a physical medium. This layer also routes data locally to the next hop on the physical network medium. The data link layer uses physical addresses (MAC addresses) assigned to each physical network device in the local network to route data from one physical device to another. These addresses are called Media Access Control (MAC) addresses in TCP/IP. MAC addresses uniquely identify a specific network device, such as a switch or a router, or a network interface card (NIC) in a computer host device. The data link layer is defined in TCP/IP by the IEEE 802.2 (Ethernet) standard.

The data link layer receives each packet from the network layer on the sending host and wraps it up in a data frame along with local routing data. The data frame is sent down to the physical layer to code an electrical or optical signal to transmit it over a wire, or over the air (wireless transmission). On the receiving host, the data link layer unwraps the data frame received to extract the packet and sends it up to the network layer.

Some TCP/IP protocols at Layer 2

- **ARP** (**Address Resolution Protocol**) is used to resolve (find) the physical (MAC) address of a host or network device, when only its logical (IP) address is known.

- **RARP** (**Reverse Address Resolution Protocol**) is used to resolve (find) the logical (IP) address of a host or network device, when only its physical (MAC) address is known.

- **CSMA/CD** (**Carrier sense multiple access collision detect protocol**) is used to allow the host and network device to share the bandwidth of a given interconnection medium.

Layer 3: Network Layer

One of the most important functions of network layer devices and protocols is choosing the best route to send packets between hosts. This is called routing. The network layer also assigns logical addresses (IP addresses) to all devices in the network to be able to identify each source host, each destination host, and each network through which packets need to be routed. Logical addresses are assigned at the network protocol level as opposed to physical addresses, which are assigned on a physical device, such as network card.

Some TCP/IP protocols at Layer 3

- **IP** (**Internet Protocol**) is used to deliver data packets over a packet switched network from a source host to a destination host based on their respective IP addresses. IP comes in two versions: IP version 4 (IPv4) and IP version 6 (IPv6). IPv4 is currently the most widely used version.
- **ICMP** (**Internet Control Message Protocol**) is used to send error and status messages about network operations and available services, mostly by host and network devices. The most typical use of ICMP is the ping command, which allows you to verify whether a host or network device is reachable over the IP network from another host or network device.

- **IPsec** (**Internet Protocol Security**) is used to secure IP data packet deliveries.

The Internet Protocol (IP) is the most important TCP/IP protocol that operates at the network layer. IP addresses are logical addresses provided by the IP in TCP/IP. Two types of protocols operate at the network layer: routed protocols and routing protocols:

- **Routed protocols** are used to route data packets. For example, IP (IPv4) is a routed protocol, and so are IPv6, AppleTalk, IPX, and SNA.
- **Routing protocols** are used to send route update packets. Route update packets carry information about new networks and new routes. Routers send each other route update packets whenever a new network is created or a new route is enabled. Some of the most common routing protocols are Routing Information Protocol (RIP), RIPv2, Enhanced Interior Gateway Routing Protocol (EIGRP), and Open Shortest Path First (OSPF). Different routing protocols use different metrics to decide which routes are better than others for example, the number of hop counts. The number of hop counts is the number of networks a data packet has to go through before reaching the destination network.

Layer 4: Transport Layer

The transport layer slices up the data to be transmitted into small chunks called data segments that can be easily sent over the network medium. The segments may end up taking different routes to get to their destination. Consequently, they may arrive in different order. The transport layer on the receiving host reorders the data segments. The transport layer also provides some error-detection mechanisms. It also insulates the upper layers from network implementation details below, by providing a generic data transfer protocol to upper layers, no matter how the network is implemented underneath.

Some TCP/IP protocols at Layer 4

- **TCP** (**Transmission Control Protocol**) is a connection-oriented transport protocol. TCP guarantees reliable transmission.

- **UDP** (**User Datagram Protocol**) is a connectionless transport protocol. UDP does not guarantee reliable transmission.

Connectionless transport

Data can be sent between two hosts without establishing a logical connection between sending and receiving hosts. Connectionless transport protocols do not guarantee reliable delivery of data segments. However, they are a bit faster than connection-oriented transport protocols, because they do not need to spend time to establish and maintain connections. User Datagram Protocol (UDP) is a connectionless transport protocol.

Connection-oriented transport

A transport protocol that establishes a logical connection between the sending and the receiving hosts is called a connection-oriented transport protocol. Connection-oriented transport protocols usually guarantee reliable delivery of data segments. They are a bit slower than connectionless transport protocols, because they need to spend some time to establish and maintain the connection. Transmission Control Protocol (TCP) is a connection-oriented transport protocol. Connection-oriented transport involves both creating a logical connection between the sending and the receiving hosts, and an exchange of acknowledgments between the hosts. Data segments are sequenced, allowing them to be sent in any order and reassembled on the receiving host.

Flow control is also part of connection-oriented reliable data transport. Flow control involves the sender and the receiver coordinating to sustain an optimal data transfer flow: As the receiver processes the data segments, it acknowledges reception to the sender. The sender then sends more segments.

TCP Flow Control

The TCP transport protocol is a connection-oriented protocol that can control the flow of data transmission to guarantee reliable transmissions. TCP on the sending host establishes a logical connection to TCP on the receiving host. This step is called three-way handshake, call setup, or virtual circuit setup. The sending host and the receiving host use this connection, or virtual circuit, to coordinate their data transfer.

The connection is terminated when no more data needs to be transferred. Any host can initiate TCP connections. The host that initiates the TCP connection becomes the sending host. The other host is the receiving host. TCP connections allow both hosts to send and receive TCP segments. TCP controls the flow of segments in each direction of a connection independently using sender and receiver sequence numbers.

Three-way handshake

The first step to establish a TCP connection involves a three-way handshake (also called "call setup" or "virtual circuit setup").

How the three-way handshake process works:

1. The host that initiates the network communication sends a TCP "Synchronize" (SYN) message to the receiving host to notify it that it wants to establish a TCP connection. This message contains, among other things, the sender starting sequence number for the TCP transmission.
2. The starting sequence number is the sequence number of the first TCP segment to transfer from sender to receiver. The sending and the receiving host then negotiate connection parameters.
3. The receiving host replies with a TCP "Synchronize" (SYN) message that contains the receiver starting sequence number. This message also sends an acknowledgment (ACK) to the sending host, indicating that the receiving host did receive the first TCP "Synchronize" message.
4. The sending host sends back an acknowledgment (ACK) to the receiving host to let it know that it did receive the receiver starting sequence number and that it is ready to send.

- At this point, the bidirectional TCP connection is established. TCP connections are bidirectional, because both hosts send SYN and ACK messages to each other to synchronize and guarantee a reliable data transfer.

Sequencing and acknowledgments

TCP controls the flow of segments in each direction of a connection independently using sender and receiver sequence numbers. TCP connections maintain two sets of sequence numbers: sender sequence numbers and receiver sequence numbers. Each number is used to control the flow of segments sent by the sending host and by the receiving host. Each segment that needs to be sent in either direction is sequenced (numbered) within the sender or receiver sequencing set, depending in which direction the segment travels.

Sequencing is also used to determine the order of the data segments. Data segments need to be reassembled in the correct order when they arrive at the destination on the receiving host, because they can get there in any order, depending on network conditions. TCP on the receiving host uses the sequence number of each data segment to determine its order during reassembly.

During transmission, errors can occur due to electrical interference, collisions, or link failure. TCP's use of sequencing and acknowledgments allows not only the control of the bidirectional transfer flow but also the correction of transmission errors by retransmitting segments that are lost or damaged. After a TCP connection is established using the three-way handshake process, TCP uses the positive acknowledgment and retransmission (PAR) process to ensure that all segments are received within a certain time period.

How PAR works:

1. Sending host starts a timer when it sends a segment. The sending host retransmits the segment if it does not receive a reception acknowledgment after a certain timeout period.
2. Sending host keeps track of the sequence number of each segment it transmits and expects reception acknowledgments for each one of them.
3. Receiving host sends acknowledgments back to the sending host for each segment it receives. The acknowledgment contains the sequence number of the next segment expected by the receiving host.

UDP simplicity

The User Datagram Protocol (UDP) is a connectionless transport protocol that does not guarantee reliable transmission. UDP is not as chatty as TCP. Hosts that transfer data using TCP need to exchange many segments just to open a connection during the three-way handshake process. They need to exchange many more segments to acknowledge reception of every single data segment. These flow control data segments add some overhead to TCP transmissions.

UDP does not add flow control overhead because

- ✦ UDP is connectionless, so there's no need to send segments to do a three-way handshake to establish a connection.
- ✦ UDP makes no use of sequencing.
- ✦ UDP does not send acknowledgments.
- ✦ UDP does not guarantee reception of data segments.

UDP is faster than TCP and can be good enough in some data-transfer scenarios such as DNS lookups and TFTP transfers. However, despite being chatty, TCP is by far the most widely used transport protocol in TCP/IP networks.

TCP/IP ports

Both TCP and UDP use ports to identify the source and destination network applications that are involved in data transmission. Every host has a logical (IP) address and a physical (MAC) address. On the other hand, more than one network application may be running on each host.

Example: you can have an e-mail program and a Web browser open at the same time on your host. So, how does your Web browser connect to a Web server, considering that the Web server host has only one IP address and may also be running an e-mail server application?

Answer: By using standard TCP/IP ports. A standard TCP/IP port exists for HTTP (the protocol used by Web browsers), a standard TCP/IP port exists for SMTP (the protocol used by some e-mail readers).

All network applications use a TCP/ IP port to allow the sending application to connect to the receiving application. So, even if you run multiple network applications on the same host, as long as each network application has its own TCP/IP port, a TCP or UDP data transmission can be accomplished.

TCP/IP ports are defined by the IANA (Internet Assigned Numbers Authority)

Some well-known reserved TCP/IP ports:

Port	Protocol	Type	Description
20,21	FTP	TCP	File Transfer Protocol
22	SSH	TCP/UDP	Secure Shell
23	Telnet	TCP	login into a remote host
25	SMTP	TCP	Simple Mail Transfer Protocol
53	DNS	TCP/UDP	Domain Name System
67/68	BOOTP	UDP	Bootstrap Protocol
67/68	DHCP	UDP	Dynamic Host Configuration Protocol
80	HTTP	TCP	Hypertext Transfer Protocol
110	POP	TCP	Post Office Protocol
123	NTP	UDP	Network Time Protocol
161,162	SNMP	UDP	Simple Network Management Protocol
443	HTTPS	TCP	Hypertext Transfer Protocol Secure
445	SMP	UDP	Simple Management Protocol
520	RIP	UDP	Routing Information Protocol

Layer 5: Session Layer

The session layer maintains a logical communication channel between a network application running on the sending host and a network application running on the receiving host. Sometimes the session layer also provides authentication services when sessions are established.

Some TCP/IP protocols at Layer 5

- **Telnet:** A protocol used to open login sessions on a computer host.

- **RPC** (**Remote-procedure call**) protocol is used to allow the execution of procedures (programs) on remote hosts.

- **ISCSI** (**The Internet small computer system interface**) protocol allows you to send SCSI commands over a TCP/IP network. iSCSI is used to interconnect specialized storage devices and computer hosts using a TCP/IP network.

Layer 6: Presentation Layer

The presentation layer is mostly concerned with data format. It converts the data between different formats so that both the sender and the receiver can use heterogeneous data. Layer 6 protocols and Layer 6 software applications exist. For example, MIME is a Layer 6 protocol that is used by e-mail software programs and Web browsers (Layer 6 applications) to convert e-mail contents that are not text into a data format that can be viewed, rendered, or otherwise processed on the computer host.

Some TCP/IP protocols at Layer 6

- **MIME** (**Multipurpose Internet Mail Extensions**) are used to allow e-mail applications to convert e-mail message contents other than text into a data format that is supported on the receiving host. MIME is also used to code non text data into an outgoing mail message.
- **Unicode**: Modern e-mail applications and Web browsers use Unicode at the presentation layer to convert characters between the character set of the sender and the character set of the receiver. Unicode provides a standard way to code characters in different character sets, including multi byte characters for some languages.

Some TCP/IP software applications at Layer 6

- **E-mail application:** E-mail applications use the MIME protocol to convert audio, video, picture, graphical, and even software application contents in e-mail messages.

- **Web browser**: Browsers also use the MIME protocol to convert non-HTML contents in Web pages.

Layer 7: Application Layer

The application layer represents the various network applications such as e-mail reader and Web browser. It is important to distinguish between Layer 7 protocols and Layer 7 software applications. For example, you use Web-browsing software to view Web pages that are transferred to your computer using the Hypertext Transfer Protocol (HTTP). Web pages are coded in Hypertext Markup Language (HTML) text format. The Web browser is a Layer 7 network application. The HTTP protocol is a Layer 7 protocol.

Some TCP/IP protocols at Layer 7

- **SMTP** (**Simple Mail Transfer Protocol**) is used to transfer, edit, and display e-mail messages.
- **HTTP** (**Hypertext Transfer Protocol**) is used to transfer text in HTML format from one host to another. HTML is the Hypertext Markup Language that marks up text with hyperlinks to allow jumping from one text document to another. The Web is based on HTTP and HTML.
- **FTP** (**File Transfer Protocol**) is used to transfer files between hosts.
- **NFS** (**Network File System**) is used to share file systems over the network.

- **SNMP** (**Simple Network Management Protocol**) is used to provide a distributed network management framework to monitor and manage host and network devices over the network.
- **DNS** (**Domain Name System**) is a protocol that helps keep track of host names and logical (IP) addresses in a network.
- **DHCP** (**Dynamic Host Configuration Protocol**) is used to assign dynamic logical addresses (IP addresses) to hosts in a network

Some TCP/IP software applications at Layer 7

- **E-mail application:** This application is used to read, edit, archive, and otherwise manage e-mail messages. E-mail applications typically use SMTP to send and receive e-mails to and from remote hosts. E-mail applications also work at Layer 6, the presentation layer. For example, e-mail applications use the Multipurpose Internet Mail Extensions (MIME) protocol to convert audio, video, picture, graphical, and even software application contents in e-mail messages into a format that can be displayed, rendered, or played on the receiving host. Whenever you send audio or video, your e-mail application also uses MIME to code the audio/video contents within the e-mail message in a format that is easily transferable over the network. Remember that Layer 6 is doing the data conversion.

- **Web browser:** A browser is used to view Web pages. Web browsers use HTTP to transfer Web pages to and from your computer. Web browsers also work at the presentation layer because they need to convert and render non-HTML format that may be embedded in an HTML Web page. For instance, when you browse a Web page that contains a video-streaming window, the Web page contains code embedded into the HTML text to instruct the Web browser on how to play that video stream. Remember that Layer 6 is doing the data conversion.