PROTOCOL ARCHITECTURE, TCP/IP & INTERNET-BASED APPLICATIONS

A protocol architecture is the layered structure of hardware and software that supports the exchange of data between systems. It also supports distributed applications like email & file transfer. At each layer, one/more protocols are implemented to provide set of rules for the exchange of data between systems. The most dominant protocol used is the TCP/IP.

TCP/IP is an Internet-based concept and is a framework for developing a complete range of computer communications standards. Another architecture is the OSI (Open Systems Interconnection) model, that is a standardized architecture often used to describe communications functions.

Why do we need a protocol architecture?

When computers, terminals and data processing devices exchange data, the procedures involved can become complex. For example, transfer of files between two computers involves the following tasks:

-the source computer must activate direct data communication path for the information and notify the network of the desired destination.

-the destination system must be prepared to receive data, which is ascertained by the source computer

-the file transfer application on the source computer must ascertain the file management program on the destination is ready to accept & store the file for its user.

-in case the file format used by the computers are different, there must be a format translation function performed.

In protocol architecture, each subtask is implemented separately and are grouped into modules. These modules are arranged in a vertical

stack. Each layer performs a related subset of functions required for the two systems to communicate with each other. Each layer relies on the next lower layer to perform more primitive functions & to conceal the details of the functions.

Ideally, changes on a single layer do not affect the functions of the other layers. It is important that the two systems have the same set of layers, therefore, communication is achieved by having peer layers. These peer layers communicate by means of formatted blocks of data that obey the protocols.

Key features of protocols include:

- **SYNTAX**; the format of the data blocks
- **SEMANTICS**; includes the control information for coordination & error handling
- **TIMING**; includes speed matching and sequencing

THE TCP/IP PROTOCOL ARCHITECTURE

It was developed as a result of experimental packet-switched network and consists of a large collection of protocols.

TCP/IP LAYERS

In general, communications involve three agents: applications, computers & network. The applications are distributed in nature, and execute on computers. The computers are connected to the networks and the data to be exchanges are transferred via the network. The layers are divided into 5 independent layers:

- **PHYSICAL LAYER**: covers the physical interface between a data transmission device & a transmission medium. It deals with characteristics of the transmission medium, the nature of the signals, data rate etc.

-**NETWORK ACCESS LAYER**: concerns with the exchange of data between the communicating systems and the network. The source system must provide the correct destination address in order for the network to route the data appropriately.

-**INTERNET LAYER**: the Internet Protocol(IP) is used in this layer to provide routing functions across multiple networks. a router is a processor that connects two networks & relays data from one network to another.

-**HOST-TO-HOST/ TRANSPORT LAYER**: to ensure reliability, all data should arrive at the destination application and in the same order in which they were sent. the TCP (Transmission Control Protocol) is used here to provide this functionality.

-**APPLICATION LAYER**: contains logic needed to support the various user applications.

OPERATION OF TCP & IP

Subnetworks are constituent networks within networks. Network access protocols are used to connect a computer to a subnetwork. These protocols enable the host to send data across the subnetwork. IP is implemented in all the end systems and the routers, it acts as a relay to move a block of data from one host through routers then to another host. TCP in implemented only in end systems and keeps track of blocks of data ensuring reliability of delivery to appropriate application.

Successful communication is ensured if every entity in the overall system has a unique address. There are two levels of addressing: the unique global Internet address that allows data delivery to the proper host & the ports which is an address unique within the host and allows TCP to deliver data to the proper process.

TCP breaks the blocks of data into smaller pieces to make them more manageable. To each pieces, the TCP appends control information called the TCP header thus forming a TCP segment. The header include:

-**DESTINATION PORT**: this shows TCP whom the data is to be delivered to.

-**SEQUENCE NUMBER**: TCP numbers the segments that it sends to a particular destination port sequentially.

-**CHECKSUM**: the sending TCP includes a code that is a function of the contents of the remainder of the segment. the receiving TCP performs the same calculation and compares the result with the incoming code. Errors can be detected.

TCP then hands over each segment to IP with instructions to transmit to destination. The transmission requires use of control information thus IP appends a control information to each segment to form an IP datagram. The IP datagram is presented to the network layer for transmission across the first subnetwork then to destination. The network access layer appends its own header, creating a packet/frame. This packet header has information that the subnetwork needs to transfer data across it. The header includes: Destination Subnetwork Address to know which attached device the packet is delivered to & Facilities Requests such as priority.

When the data is received at destination, the reverse process occurs, at each layer the corresponding header is removed as it is passed on to the next higher layer until the original user data are delivered at the destination process.

Bellow is the illustration:

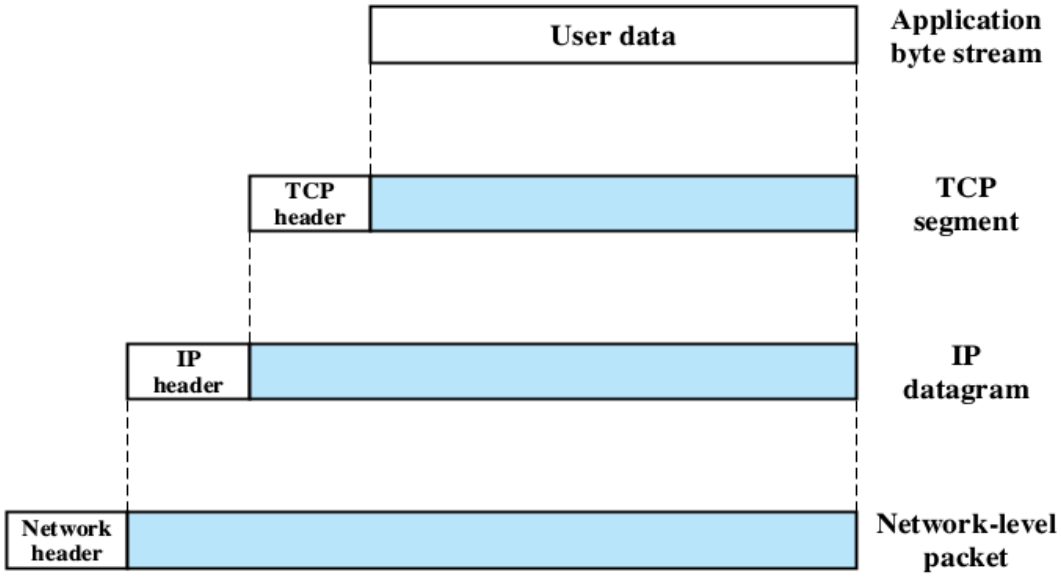

Figure 2.2 Protocol Data Units (PDUs) in the TCP/IP Architecture

TCP AND UDP

A connection is simply a temporary logical association between two entities in different systems. A logical connection refers to a given pair of port values. During the connection, each entity keeps track of TCP segments coming and going in order to regulate the flow of segments & to recover from lost/damaged segments.

The TCP header is a minimum of 20 octets/160 bits and includes:

-**SOURCE PORT & DESTINATION PORT**: these fields identify the applications at the source and destination systems using the connection

-**SEQUENCE NUMBER, ACKNOWLEDGE NUMBER & WINDOW**: these fields provide flow control and error control

-**CHECKSUM**: a 16-bit frame check sequence used to detect errors

UDP(User Datagram Protocol) is a transport-level protocol that does not guarantee delivery, preservation of sequence or protection against duplication. It uses minimum protocol mechanism to send messages. It is connectionless and has a checksum to verify that no error occurs in data. SNMP (Simple Network Management Protocol) uses UDP.

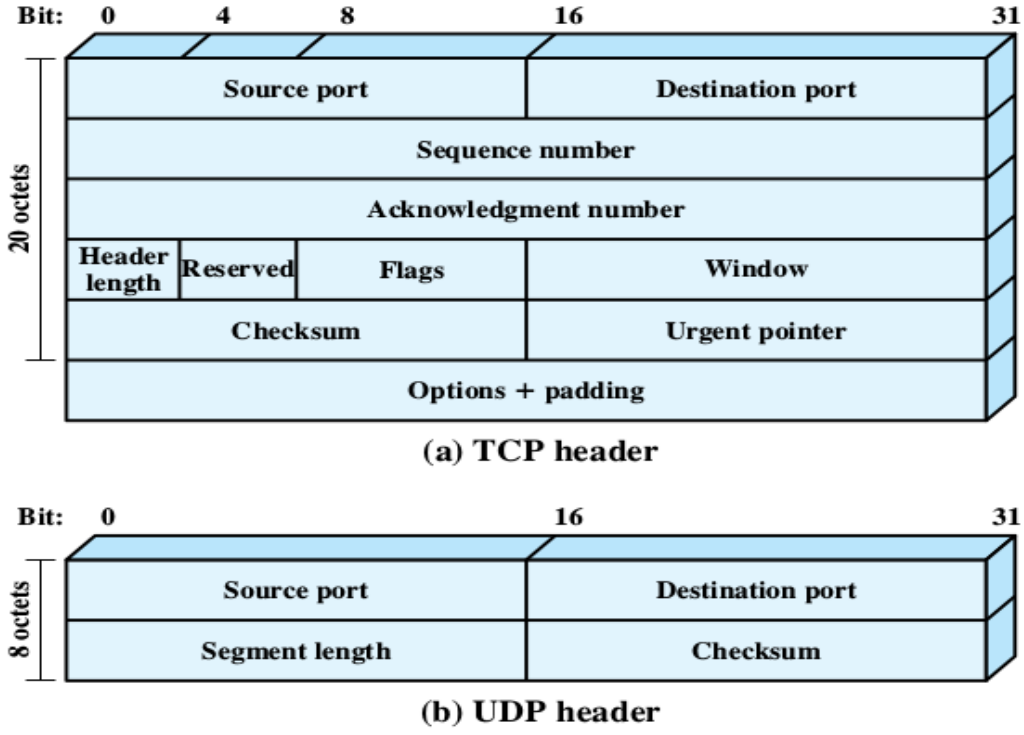

Figure 2.3 TCP and UDP Headers

IPv4 & IPv6

IP is the keystone of the TCP/IP protocol. The IP header together with the segment from the transport layer form the IP-level PDU (Protocol Data Unit). The IP header includes 32-bit source & destination address, header checksum to detect errors and avoid

misdelivery, protocol field indicating IP and the ID,Flags & Fragment Offset fields used in fragmentation and reassembly process.
IPv6 is a next generation IP that has enhancements over IPv4, it is designed to accommodate the higher speeds of today's networks and the mix of data streams(graphic + video). The current IPv4 uses 32-bit address to specify a source/destination.

With the explosive growth of the Internet the IPv4 addresses will be insufficient to accommodate all systems.

IPv6 includes an 128-bit source & destination address fields. It is divided into 8 groups of 16 bits. Uses hexadecimal notation separated by colon. The 16 bits are further subdivided into groups of 4, then changed into hexadecimal. Leading zero's are omitted while one/more consecutive zero's are replaced by double colon.

Computers with IPv4 must go through an intermediate node(proxy) to communicate with computers with IPv6. Computers that have both IPv4 and IPv6 use dual-stack to communicate.

TCP/IP APPLICATIONS

Applications standardized to operate on top of TCP are:

-**SMTP (Simple Mail Transfer Protocol)**: provides basic email transport facility. has a mechanism for transferring messages among separate hosts. it includes mailing lists, return receipts & forwarding. local editing and native email facility is required. Once a message is created, SMTP accepts the message & makes use of TCP to send to a SMTP module/other host. target SMTP module will make use of a local email package to store incoming messages in user's mailbox.

-**FTP (File Transfer Protocol)**: used to send files from one system to another under user command. Accommodates both text and binary files and provides features for controlling user access. FTP sets up a TCP connection to the target system when user wants to engage in file transfer. The connection allows user ID & password to be transmitted, allows user to specify the file + actions. Once file transfer is approved, a second TCP connection is set up for data transfer. On completion of data transfer, the connection is used to signal the completion and to accept new file transfer commands.

-**TELNET**: provides remote logon capability which enables a user to use a terminal/PC to logon to a remote computer & function as if it is directly connected to that computer. Designed to work with simple scroll-mode terminals. USER TELNET interacts with the terminal I/O module to communicate with a local terminal. It converts the characteristics of real terminals to communicate with a standard and vice versa. SERVER TELNET interacts with the application acting as surrogate terminal handler so that remote terminals appear as local to the application. traffic between the user & server telnet are carried over TCP connection.

PROTOCOL INTERFACE

Each layer in the TCP/IP protocol interacts with its immediate adjacent layers. At the source, the application layer makes use of the services of the end-to-end layer and provides data down to that layer. Same occurs at interface of end-to-end & Internet layers, also at the interface of Internet & network access layers. At the destination, each layer delivers data up to the next higher layer.

It is possible to develop applications that directly invoke the services of any of the layers. Most applications like BGP(border gateway protocol),FTP,HTTP,SMTP & TELNET use TCP, some applications are specific-purposed like the MIME(multipurpose Internet Mail Extension). Some use UDP like SNMP and others like ICMP(Internet Control Message Protocol),IGMP(Internet Group Management Protocol),OSPF(Open Shortest Path First) & RSVP(Resource ReSerVation Protocol) use IP directly.

TCP/IP MODEL

The TCP/IP model is mostly used over the OSI model because it is robust, widely available on almost every hardware & OS platform and is also used by the Internet. It consists of 4 layers:

-**APPLICATION LAYER**: defines application protocols and how host programs interface with transport layer. protocols includes HTTP,FTP,DNS & SMTP.

-**TRANSPORT LAYER**: provides communication session management between computers. protocols include TCP,UDP & RTP.

-**INTERNET LAYER**: information and data is packaged into IP datagrams that contain source & destination address, performs routing of the datagrams. protocols include IP, ICMP & ARP.

-**NETWORK INTERFACE LAYER**: specifies in detail how data is sent through network, including hoe bits are electrically signaled. protocols include Ethernet, token-ring & frame relay.

The Application layer is a combination of the session + presentation + application layers of OSI.

The Networking Interface layer is a combination of data link + physical layers of the OSI model.

OSI MODEL

The OSI (Open Systems Interconnection) reference model was developed to model a computer protocol architecture and as a framework for developing protocol standards. Protocols are developed to perform the functions of each layer. It consists of 7 layers:

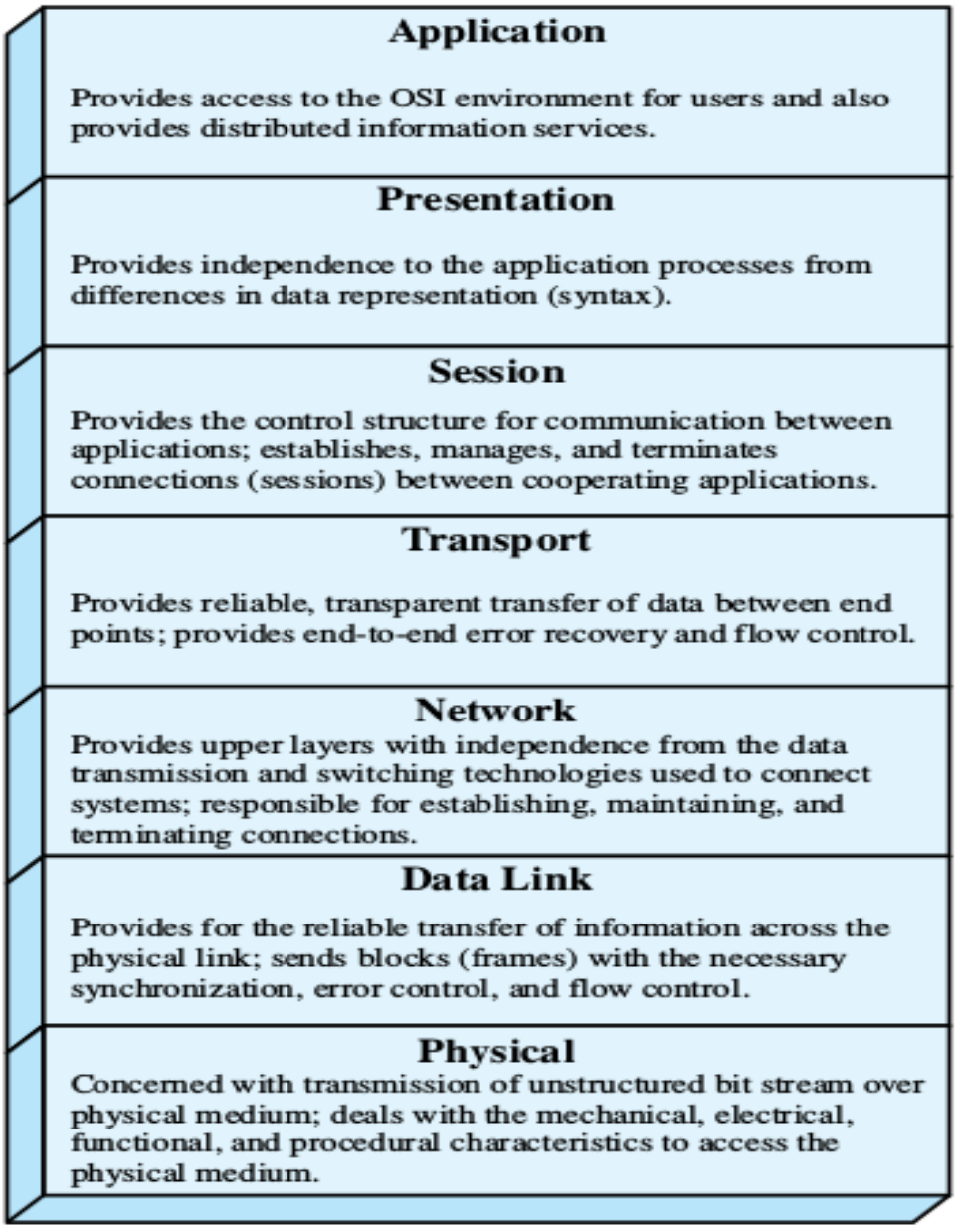

Figure 2.6 The OSI Layers

TCP HANDSHAKE

The handshake illustrates how to setup connection between client and server. It happens prior to sending information.

—the client sends a synchronization request to the server.

—server listens and receives the request

—server sends an acknowledgment + synchronization message to the client

—client receives the message and sends an acknowledgement of receipt of the message.

IP ADDRESSING SCHEME

IP addresses are used to locate individual hosts for communication. This scheme provides 232 possible addresses with over 4.2 thousand million individual addresses. To accommodate different sized organizations that require different number of hosts addresses, we split the IP addresses into different classes: Class A, Class B, Class C, Class D and Class E.

Subnetting is the process that allows large chunks of addresses to be further split into network and host components. A subnet mask is used to change part of the address that represents host and network. Supernetting is the way of joining subtends to make a supernet.

MAC ADDRESSING SCHEME

MAC (Media Access Control) address is a unique 6 number address separated by colon and are hard coded into the Ethernet card for each

device. These are used at the lower levels of the protocol stack, that is layer 1&2 and differ depending on media used. MAC addresses allow different networking protocols to be carried over Ethernet. ARP (Address Resolution Protocol) is used to translate IP addresses into MAC addresses

SOCKETS & PORT NUMBERS

IP addresses provide connection to the right machine, though it can't distinguish the different service that is required.

PORTS are used to distinguish applications and its value ranges from 0 – 65535.

SOCKET is a combination of port + protocol, which is unique for every service.

Port numbers are available for TCP & UDP. The first 1000 port numbers are reserved for specific applications. In Linux they are listed in the /etc/services directory.

Examples of port numbers:

20,21 = FTP
23 = Telnet
25 = SMTP
53 = DNS
80 = WWW
110 = POP3(post office protocol)
144 = News
6000 = X-Windows

ROUTING

Routing is required for computers to communicate over the Internet and other networks. Routers direct traffic by taking incoming packets, based upon destination address it sends it through a different interface to another router, then to the end destination. A default gateway is a router directly attached to the same LAN segment as the host & knows how to route packets. When a packet passes through a router it is called a HOP.

Types of routing protocols include: RIP(Routing Information Protocol), RIP2 & OSPF(Open Shortest Path First).