Network Communication Architecture and Protocols

A **network architecture** is a blueprint of the complete computer communication network, which provides a framework and technology foundation for designing, building and managing a communication network. It typically has a layered structure. Layering is a modern network design principle which divides the communication tasks into a number of smaller parts, each part accomplishing a particular sub-task and interacting with the other parts in a small number of well-defined ways.

If network architecture is **open**, no single vendor owns the technology and controls its definition and development. Anyone is free to design hardware and software based on the network architecture. The **TCP/IP network architecture,** which the Internet is based on, is such open network architecture and it is adopted as a worldwide network standard and widely deployed in local area network (LAN), wide area network (WAN), small and large enterprises, and last but not the least, the Internet.

Open Systems Interconnection (OSI) network architecture, developed by International Organization for Standardization, is an open standard for communication in the network across different equipment and applications by different vendors. Though not widely deployed, the OSI 7 layer model is considered the primary network architectural model for inter-computing and inter-networking communications.

In addition to the OSI network architecture model, there exist other network architecture models by many vendors, such as IBM SNA (Systems Network Architecture), Digital Equipment Corporation (DEC; now part of HP) DNA (Digital Network Architecture), Apple computer's AppleTalk, and Novell's NetWare. Actually, the TCP/IP architecture does not exactly match the OSI model. Unfortunately, there is no universal agreement regarding how to describe TCP/IP with a layered model. It is generally agreed that TCP/IP has fewer levels (from three to five layers) than the seven layers of the OSI model.

Network architecture provides only a conceptual framework for communications between computers. The model itself does not provide specific methods of communication. Actual communication is defined by various communication **protocols**.

## OSI Network Architecture 7 Layers Model

Open Systems Interconnection (OSI) model is a reference model developed by ISO (International Organization for Standardization) in 1984, as a conceptual framework of standards for communication in the network across different equipment and applications by different vendors. It is now considered the primary architectural model for inter-computing and internetworking communications. **Most of the network communication protocols used today have a structure based on the OSI model.** The OSI model defines the communications process into 7 layers, dividing the tasks involved with moving information between networked computers into seven smaller, more manageable task groups. A task or group of tasks is then assigned to each of the seven OSI layers. Each layer is reasonably self-contained so that the tasks assigned to each layer can be implemented independently. This enables the solutions offered by one layer to be updated without adversely affecting the other layers.

The OSI 7 layers model has clear characteristics at each layer. Basically, layers 7 through 4 deal with end to end communications between data source and destinations, while layers 3 to 1 deal with communications between network devices. On the other hand, the seven layers of the OSI model can be divided into two groups: upper layers (layers 7, 6 & 5) and lower layers (layers 4, 3, 2, 1). The upper layers of the OSI model deal with application issues and generally are implemented only in software. The highest layer, the application layer, is closest to the end user. The lower layers of the OSI model handle data transport issues. The physical layer and the data link layer are implemented in hardware and software. The lowest layer, the physical layer, is closest to the physical network medium (the wires, for example) and is responsible for placing data on the medium.

The specific description for each layer is as follows:

**Layer 7: Application Layer**

Defines interface to user processes for communication and data transfer in network

Provides standardized services such as virtual terminal, file and job transfer and operations

**Layer 6: Presentation Layer**

Masks the differences of data formats between dissimilar systems Specifies architecture-independent data transfer format. Encodes and decodes data; encrypts and decrypts data; compress­ es and decompresses data

**Layer 5: Session Layer**

Manages user sessions and dialogues. Controls establishment and termination of logic links between users Reports upper layer errors

**Layer 4: Transport Layer**

Manages end-to-end message delivery in network Provides reliable and sequential packet delivery through error recovery and flow control mechanisms. Provides connectionless oriented packet delivery

**Layer 3: Network Layer**

Determines how data are transferred between network devices Routes packets according to unique network device addresses Provides flow and congestion control to prevent network resource depletion

**Layer 2: Data Link Layer**

Defines procedures for operating the communication links Frames packets Detects and corrects packets transmit errors

**Layer 1: Physical Layer**

Defines physical means of sending data over network devices Interfaces between network medium and devices Defines optical, electrical and mechanical characteristics

## TCP/IP Four Layers Architecture Model

TCP/IP architecture does not exactly follow the OSI model. Unfortunately, there is no universal agreement regarding how to describe TCP/IP with a layered model. It is generally agreed that TCP/IP has fewer levels (from three to five layers) than the seven layers of the OSI model. We adopt a four layers model for the TCP/IP architecture.

TCP/IP architecture omits some features found under the OSI model, combines the features of some adjacent OSI layers and splits other layers apart. The 4-layer structure of TCP/IP is built as information is passed down from applications to the physical network layer. When data is sent, each layer treats all of the information it receives from the upper layer as data, adds control information (header) to the front of that data and then pass it to the lower layer. When data is received, the opposite procedure takes place as each layer processes and removes its header before passing the data to the upper layer.

The TCP/IP 4-layer model and the key functions of each layer are described below:

**Application Layer**

The Application Layer in TCP/IP groups the functions of OSI Application, Presentation Layer and Session Layer. Therefore any process above the transport layer is called an Application in the TCP/IP architecture. In TCP/IP socket and port are used to describe the path over which applications communicate. Most application level protocols are associated with one or more port number.

**Transport Layer**

In TCP/IP architecture, there are two Transport Layer protocols. The Transmission Control Protocol (TCP) guarantees information transmission. The User Datagram Protocol (UDP) transports datagrams without end-to-end reliability checking. Both protocols are useful for different applications.

**Network Layer**

The Internet Protocol (IP) is the primary protocol in the TCP/IP Network Layer. All upper and lower layer communications must travel through IP as they are passed through the TCP/IP protocol stack. In addition, there are many supporting protocols in the Network Layer, such as ICMP, to facilitate and manage the routing process.

**Network Access Layer**

In the TCP/IP architecture, the Data Link Layer and Physical Layer are normally grouped together to become the Network Access layer. TCP/IP makes use of existing Data Link and Physical Layer standards rather than defining its own. Many RFCs describe how IP utilizes and interfaces with the existing data link protocols such as Ethernet, Token Ring, FDDI, HSSI, and ATM. The physical layer, which defines the hardware communication properties, is not often directly interfaced with the TCP/IP protocols in the network layer and above.

## 

## Network Protocol: Definition and Overview

The OSI model, and any other network communication model, provides only a conceptual framework for communication between computers, but the model itself does not provide specific methods of communication. Actual communication is defined by various **communication protocols**. In the context of data communication, a protocol is a formal set of rules, conventions and data structure that governs how computers and other network devices exchange information over a network. In other words, a protocol is a standard procedure and format that two data communication devices must understand, accept and use to be able to talk to each other.

In modern protocol design, protocols are "layered" according to the OSI 7 layer model or a similar layered model. Layering is a design principle which divides the protocol design into a number of smaller parts, each part accomplishing a particular sub-task, and interacting with the other parts of the protocol only in a small number of well-defined ways. Layering allows the parts of a protocol to be designed and tested without a combinatorial explosion of cases, keeping each design relatively simple. Layering also permits familiar protocols to be adapted to unusual circumstances.

The header and/or trailer at each layer reflect the structure of the protocol. Detailed rules and procedures of a protocol or protocol stack are often defined by a lengthy document. For example, IETF uses RFCs (Request for Comments) to define protocols and updates to the protocols.

A wide variety of communication protocols exist. These protocols are defined by many standard organizations throughout the world and by technology vendors over years of technology evolution and development. **One of the most popular protocol suites is TCP/IP**, which is the heart of lnternetworking communications. The IP, the Internet Protocol, is responsible for exchanging information between routers so that the routers can select the proper path for network traffic, while TCP is responsible for ensuring the data packets are transmitted across the network reliably and error free. LAN and WAN protocols are also critical protocols in network communications. The LAN protocols suite is for the physical and data link layers communications over various LAN media such as Ethernet wires and wireless waves. The WAN protocol suite is for the lowest three layers and defines communication over various wide-area media, such as fiber optic and copper cable.

Network communication has gradually evolved - Today's new technologies are based on accumulation over years of technologies, which may be still existing or obsolete. Because of this, the protocols which define the network communication are highly inter-related. Many protocols rely on others for operation. For example, many routing protocols use other network protocols to exchange information between routers.

In addition to standards for individual protocols in transmission, there are now also interface standards for different layers to talk to the ones above or below (usually operating-system-specific). For example: Winsock and Berkeley sockets between layers 4 and 5, NDIS and ODI between layers 2 and 3.

The protocols for data communication cover all areas as defined in the OSI model. However, the OSI model is only loosely defined. A protocol may perform the functions of one or more of the OSI layers, which introduces complexity to understand protocols relevant to the OSI 7 layer model. In real-world protocols, there is some argument as to where the distinctions between layers are drawn; there is no one black and white answer.

To develop a complete technology that is useful for the industry, very often a group of protocols is required in the same layer or across many different layers. Different protocols often describe different aspects of a single communication; taken together, these form a protocol suite. For example, Voice over IP (VOiP), a group of protocols developed by many vendors and standard organizations, has many protocols across the 4 top layers in the OSI model.

Protocols can be implemented either in hardware or software, or a mixture of both. Typically, the lower layers are implemented in hardware, with the higher layers being implemented in software.

Protocols could be grouped into suites (or families, or stacks) by their technical functions, or origin of the protocol introduction, or both. A protocol may belong to one or multiple protocol suites, depending on how you categorize it. For example, the Gigabit Ethernet protocol IEEE 802.32 is a LAN (Local Area Network) protocol and it can also be used in MAN (Metropolitan Area Network) communications.

Most recent protocols are designed by the IETF for Internet­ working communications, and the IEEE for local area networking (LAN) and metropolitan area networking (MAN). The ITU-T contributes mostly to wide area networking (WAN) and telecommunications protocols. ISO has its own suite of protocols for internetworking communications, which is mainly deployed in European countries.

## TCP/IP Protocols

The TCP/IP protocol suite establishes the technical foundation of the Internet. Development of the TCP/IP started as DOD projects. Now, most protocols in the suite are developed by the Internet Engineering Task Force (IETF) under the Internet Architecture Board (IAB), an organization initially sponsored by the US government and now an open and autonomous organization. The IAB provides the coordination for the R&D underlying the TCP/IP protocols and guides the evolution of the Internet. The TCP/IP protocols are well documented in the Request For Comments (RFC), which are drafted, discussed, circulated and approved by the IETF committees. All documents are open and free and can be found online in the IETF site listed in the reference.

TCP/IP architecture does not exactly match the OSI model. Unfortunately, there is no universal agreement regarding how to describe TCP/IP with a layered model. It is generally agreed that TCP/IP has fewer levels (from three to five layers) than the seven layers of the OSI model. In this article, we force TCP/IP protocols into the OSI 7 layers structure for comparison purpose.

The TCP/IP suite's core functions are addressing and routing (IP/IPv6 in the networking layer) and transportation control (TCP, UDP in the transport layer).

**IP - Internet Protocol**

Addressing of network components is a critical issue for information routing and transmission in network communications. Each technology has its own convention for transmitting messages between two machines within the same network. On a LAN, messages are sent between machines by supplying the six bytes unique identifier (the "MAC" address). In an SNA network, every machine has Logical Units with their own network addresses. DEC­ NET, AppleTalk, and Novell lPX all have a scheme for assigning numbers to each local network and to each workstation attached to the network.

On top of these local or vendor specific network addresses, IP assigns a unique number to every network device in the world, which is called an **IP address**. This IP address is a four bytes value in IPv4 that, by convention, is expressed by converting each byte into a decimal number (O to 255) and separating the bytes with a period. In IPv6, the IP address has been increased to 16 bytes.

**TCP - Transmission Control Protocol**

TCP provides a reliable stream delivery and virtual connection service to ap­ plications through the use of sequenced acknowledgment with retransmission of packets when necessary. TCP provides stream data transfer, transportation reliability, efficient flow control, full-duplex operation, and multiplexing.

In the following TCP/IP protocol stack table, **we list several protocols** according to their functions in mapping to the OSI 7 layers network communication reference model. However, the TCP/IP architecture does not follow the OSI model closely, for example, most TCP/IP applications directly run on top of the transport layer protocols, TCP and UDP, without the presentation and session layers in between.

## Application Layer

BOOTP: Bootstrap Protocol

DCAP: Data Link Switching Client Access Protocol

DHCP: Dynamic Host Configuration Protocol

DNS: Domain Name Systems

***FTP: File Transfer Protocol***

Finger: User Information Protocol

***HTTP: Hypertext Transfer Protocol***

***S-HTTP: Secure Hypertext Transfer Protocol (S-HTTP)***

***IMAP & IMAP4: Internet Message Access Protocol*** IPDC: IP Device Control

IRCP (IRC): Internet Relay Chat Protocol

LDAP: Lightweighted Directory Access Protocol

MIME (S-MIME): Multipurpose Internet Mail Extensions (Secure MIME)

NAT: Network Address Translation

NNTP: Network News Transfer Protocol NTP: Network Time Protocol

***POP & POP3: Post Office Protocol (version 3)***

RLOGIN: Remote Login in Unix

RMON: Remote Monitoring MIBs in SNMP SLP: Service Location Protocol

***SMTP: Simple Mail Transfer Protocol***

SNMP: Simple Network Management Protocol SNTP: Simple Network Time Protocol TELNET: TCP/IP Terminal Emulation Protocol TFTP: Trivial File Transfer Protocol

URL: Uniform Resource Locator

X-Window: X Window or X Protocol or X System

## Presentation Layer

LPP: Lightweight Presentation Protocol

## Session Layer

RPG: Remote Procedure Call protocol

## Transport Layer

ITOT: ISO Transport Over TCP/IP

RDP: Reliable Data Protocol

RUDP: Reliable UDP

TALI: Transport Adapter Layer Interface

TCP: Transmission Control Protocol

UDP: User Datagram Protocol

## Network Layer: Routing

BGP/BGP4: Border Gateway Protocol EGP: Exterior Gateway Protocol

***IP: Internet Protocol***

1Pv6: Internet Protocol version 6

ICMP/ICMPv6: Internet Control Message Protocol

IRDP: ICMP Router Discovery Protocol

Mobile IP: IP Mobility Support Protocol for 1Pv4 & 1Pv6

NARP: NBMA Address Resolution Protocol

NHRP: Next Hop Resolution Protocol OSPF: Open Shortest Path First

RIP (RIP2): Routing Information Protocol RIPng: RIP for 1Pv6

RSVP: Resource Reservation Protocol

VRRP: Virtual Router Redundancy Protocol

## Data Link Layer

ARP and lnARP: Address Resolution Protocol and Inverse ARP

IPCP and 1Pv6CP: IP Control Protocol and 1Pv6 Control Protocol

RARP: Reverse Address Resolution Protocol SLIP: Serial Line IP