COMPUTER NETWORKS AND INTERNET PROTOCOLS

Protocol Stack – Layered Services

SOUMYA K GHOSH
COMPUTER SCIENCE AND ENGINEERING
IIT KHARAGPUR

SANDIP CHAKRABORTY
COMPUTER SCIENCE AND ENGINEERING
IIT KHARAGPUR

Network Protocols

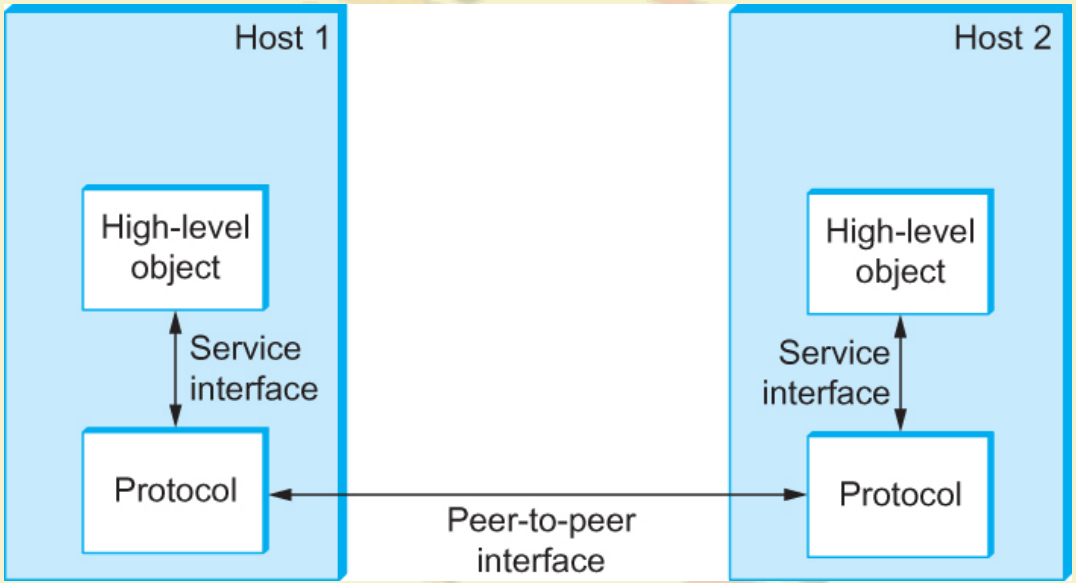
- Protocol defines the interfaces between the layers in the same system and with the layers of peer system
- Building blocks of a network architecture
- Each protocol object has two different interfaces
  - service interface: operations on this protocol
  - peer-to-peer interface: messages exchanged with peer
- "Protocol" includes
  - specification of peer-to-peer interface
  - module that implements this interface
- Features:
  - Protocol Specification: prose, pseudo-code, state transition diagram
  - Interoperable: when two or more protocols that implement the specification accurately
  - IETF: Internet Engineering Task Force

Ref: Computer Networks: A Systems Approach, by Larry L. Peterson and Bruce S. Davie

Key Elements of a Protocol

- Syntax
  - Data formats
  - Signal levels
- Semantics
  - Control information
  - Error handling
- Timing
  - Speed matching
  - Sequencing

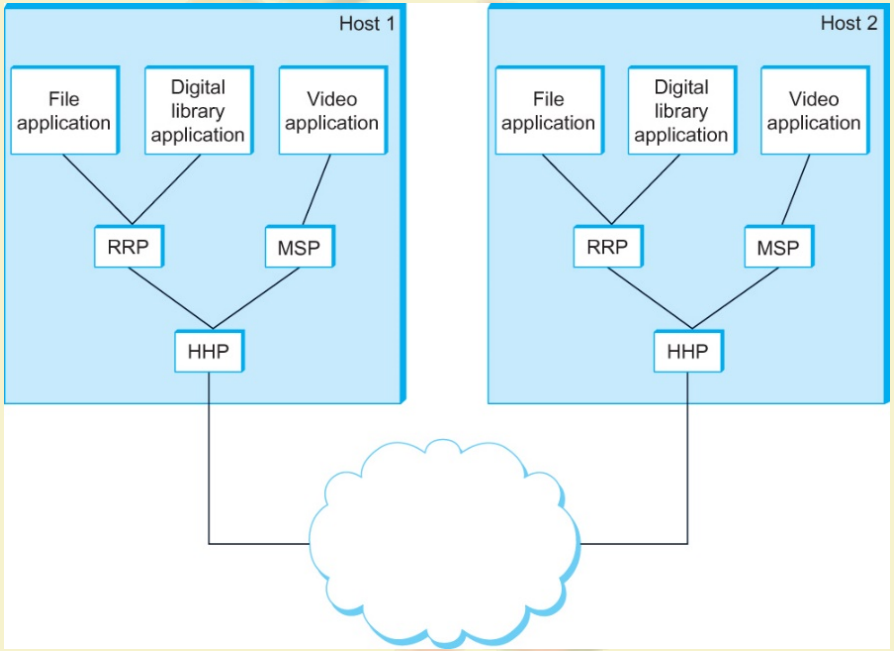
Interfaces



Service and Peer Interfaces

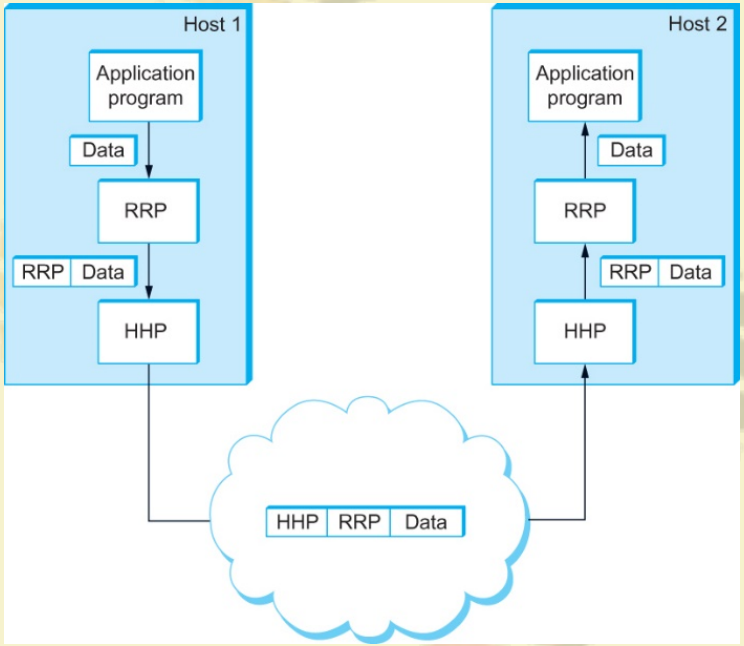
Ref: Computer Networks: A Systems Approach, by Larry L. Peterson and Bruce S. Davie

Protocol Hierarchy



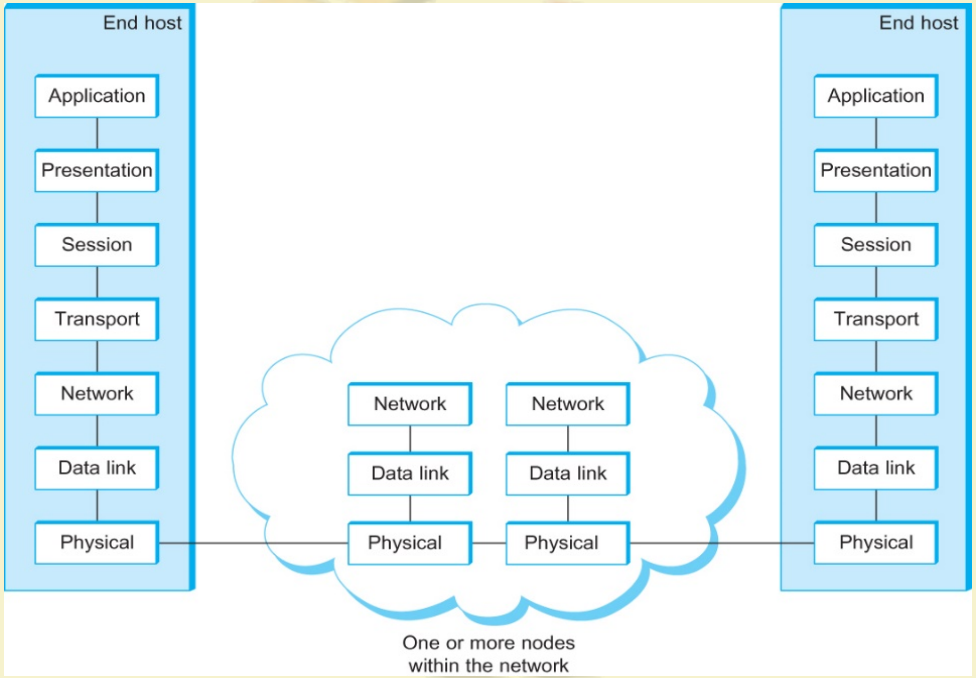
Ref: Computer Networks: A Systems Approach, by Larry L. Peterson and Bruce S. Davie

Encapsulation



High-level messages are encapsulated inside of low-level messages

OSI (Open Systems Interconnection) Model



Protocol Layers - Functions

- Physical Layer
  - Handles the transmission of raw bits over a communication link
- Data Link Layer
  - Collects a stream of bits into a larger aggregate called a *frame*
  - Network adaptor along with device driver in OS implement the protocol in this layer
  - Frames are actually delivered to hosts
- Network Layer
  - Handles routing among nodes within a packet-switched network
  - Unit of data exchanged between nodes in this layer is called a *packet*

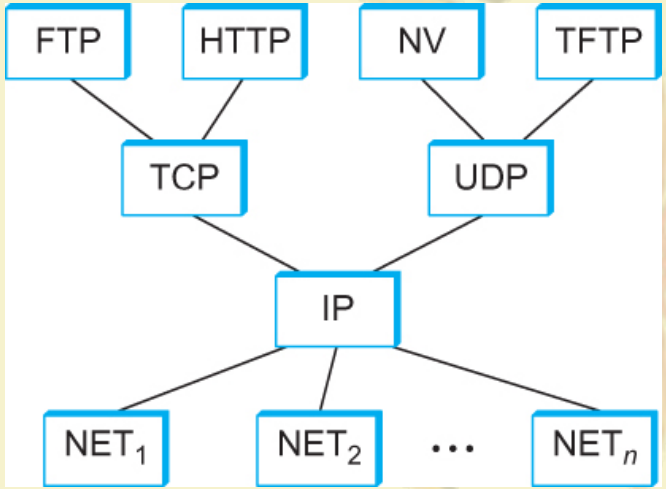
Lower three layers are typically implemented on all network nodes

Protocol Layers - Functions

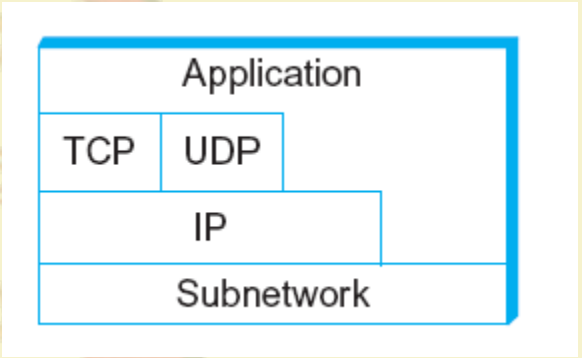
- Transport Layer
  - Implements a process-to-process channel
  - Unit of data exchanges in this layer is called a *message*
- Session Layer
  - Provides a name space that is used to tie together the potentially different transport streams that are part of a single application
- Presentation Layer
  - Concerned about the format of data exchanged between peers
- Application Layer
  - Standardize common type of exchanges

Transport layer and the higher layers typically run only on end-hosts and not on the intermediate switches and routers

Internet Architecture



Internet Protocol Graph



Internet Architecture

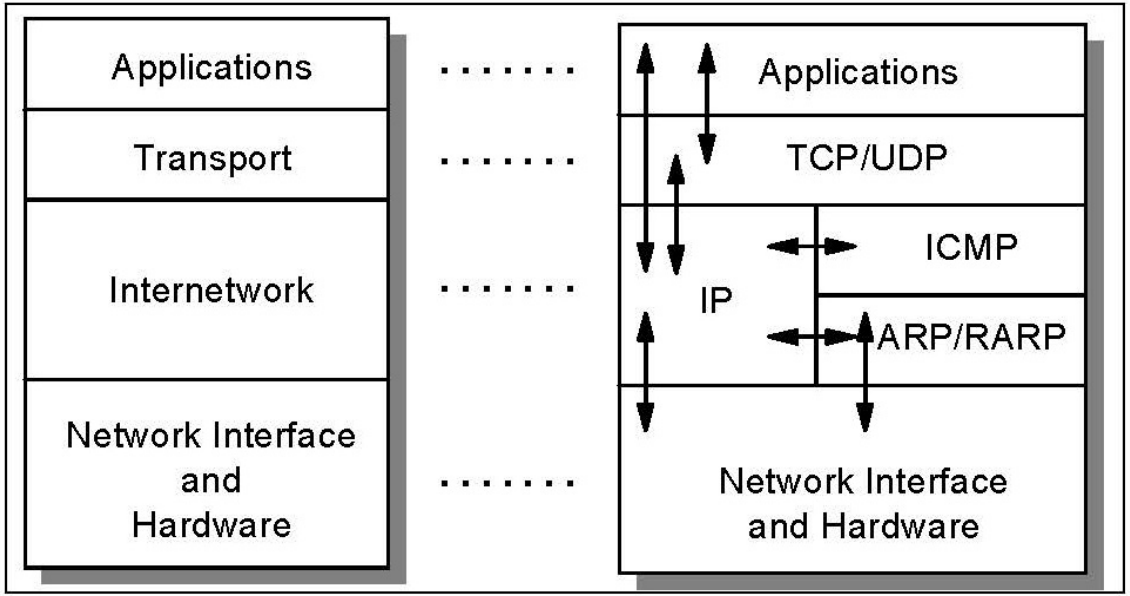
- Defined by IETF
- Three main features
  - Does not imply strict layering. The application is free to bypass the defined transport layers and to directly use IP or other underlying networks
  - An hour-glass shape – wide at the top, narrow in the middle and wide at the bottom. IP serves as the focal point for the architecture
  - In order for a new protocol to be officially included in the architecture, there needs to be both a protocol specification and at least one (and preferably two) representative implementations of the specification

Ref: Computer Networks: A Systems Approach, by Larry L. Peterson and Bruce S. Davie

Network Application Programming Interface (API)

- Interface exported by the network
- Since most network protocols are implemented (those in the high protocol stack) in software and nearly all computer systems implement their network protocols as part of the operating system
- The interface is called the *network Application Programming Interface* (API)

TCP/IP Protocol Stack



TCP/IP Layers – Group of functions in each layer

Ref: IBM Redbooks - TCP/IP Tutorial and Technical Overview

TCP/IP Protocol Stack

- **Application layer**
  - The application layer is provided by the program that uses TCP/IP for communication.
  - An application is a user process cooperating with another process usually on a different host (there is also a benefit to application communication within a single host).
  - Examples of applications include Telnet and the File Transfer Protocol (FTP).
  - The interface between the application and transport layers is defined by port numbers and “sockets”

TCP/IP Protocol Stack

- **Transport layer**
  - Transport layer provides the end-to-end data transfer by delivering data from an application to its remote peer. Multiple applications can be supported simultaneously.
  - Most-used transport layer protocol is the Transmission Control Protocol (TCP), which provides connection-oriented reliable data delivery, duplicate data suppression, congestion control, and flow control.
  - Another transport layer protocol: User Datagram Protocol (UDP)
  - It provides connectionless, unreliable, best-effort service.
  - As a result, applications using UDP as the transport protocol have to provide their own end-to-end integrity, flow control, and congestion control, if desired.
  - Usually, UDP is used by applications that need a fast transport mechanism and can tolerate the loss of some data.

TCP/IP Protocol Stack

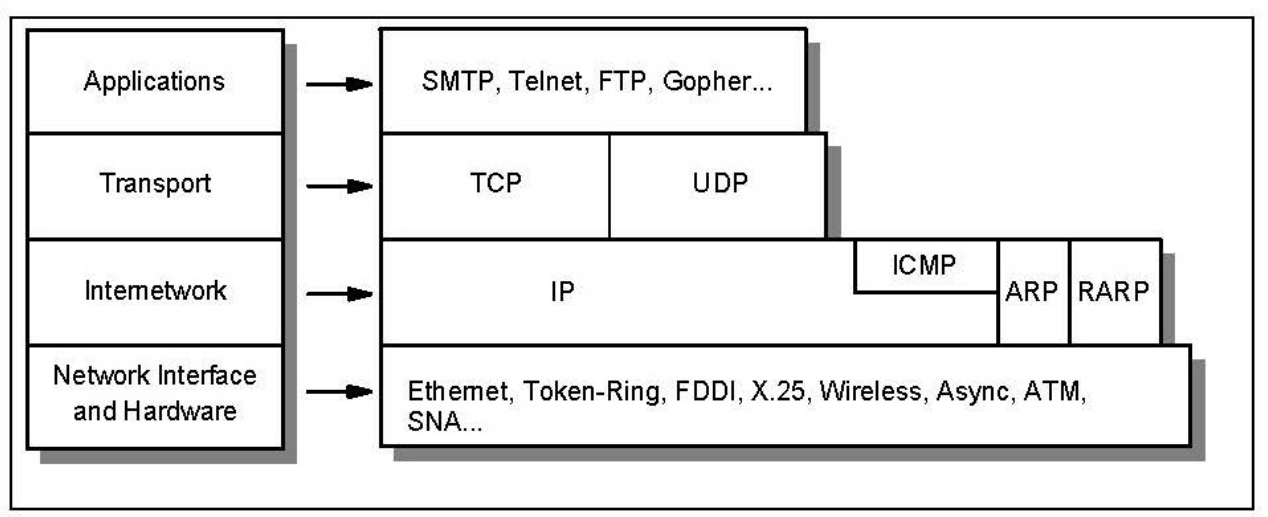
- **Internetwork layer (IP / Network Layer)**
  - The internetwork layer, also called the *internet layer* or the *network layer*, provides the "virtual network" image of an internet (this layer shields the higher levels from the physical network architecture below it).
  - Internet Protocol (IP) is the most important protocol in this layer. It is a connectionless protocol that does not assume reliability from lower layers.
  - IP does *not* provide reliability, flow control, or error recovery. These functions must be provided at a higher level.
  - IP provides a routing function that attempts to deliver transmitted messages to their destination.
  - A message unit in an IP network is called an *IP datagram*. This is the basic unit of information transmitted across TCP/IP networks.
  - Typical internetwork-layer protocols are IP, ICMP, IGMP, ARP, and RARP.

TCP/IP Protocol Stack

- **Network interface layer**
  - The network interface layer, also called the *link layer* or the *data-link layer*, is the interface to the actual network hardware.
  - This interface may or may not provide reliable delivery, and may be packet or stream oriented.
  - In fact, TCP/IP does not specify any protocol here, but can use almost any network interface available, which illustrates the flexibility of the IP layer.
  - Examples are IEEE 802.2, X.25 (which is reliable in itself), ATM, FDDI, and even SNA.
  - There should be some underlying physical networks and interfaces

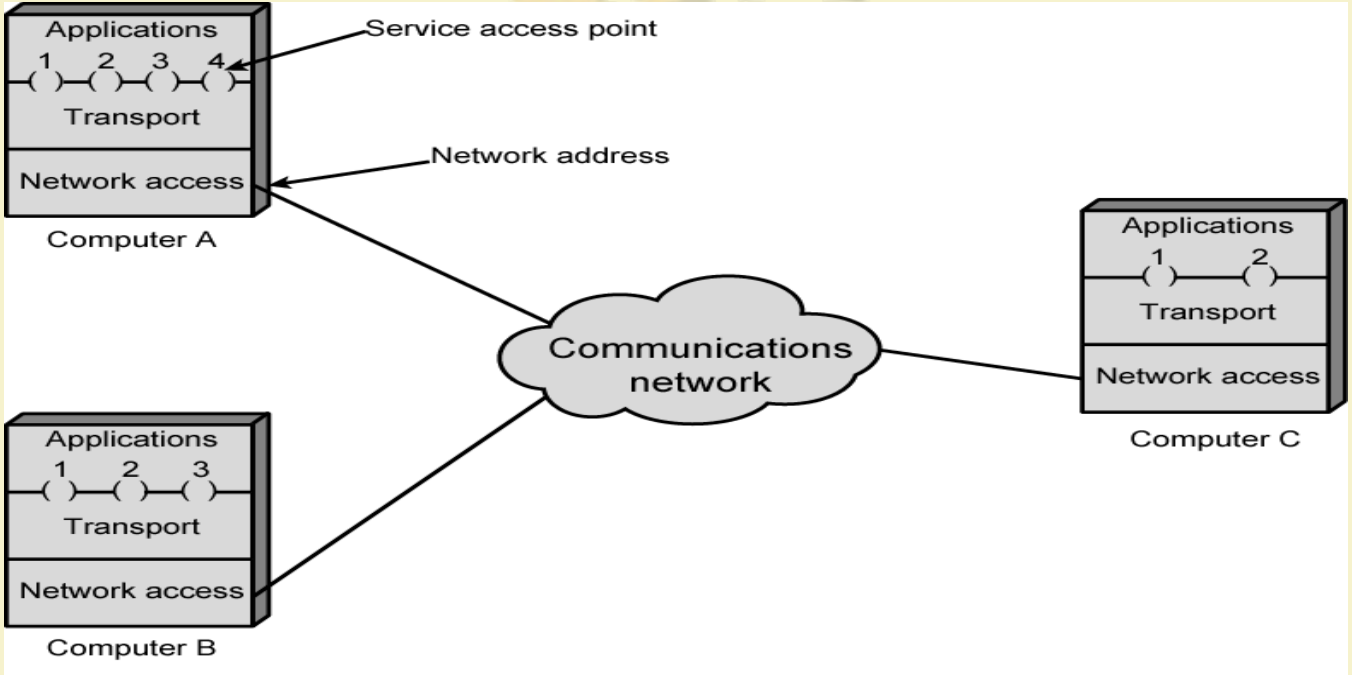
TCP/IP Protocol Stack

TCP/IP specifications do not describe or standardize any network-layer protocols per se; they only standardize ways of accessing those protocols from the internetwork layer.



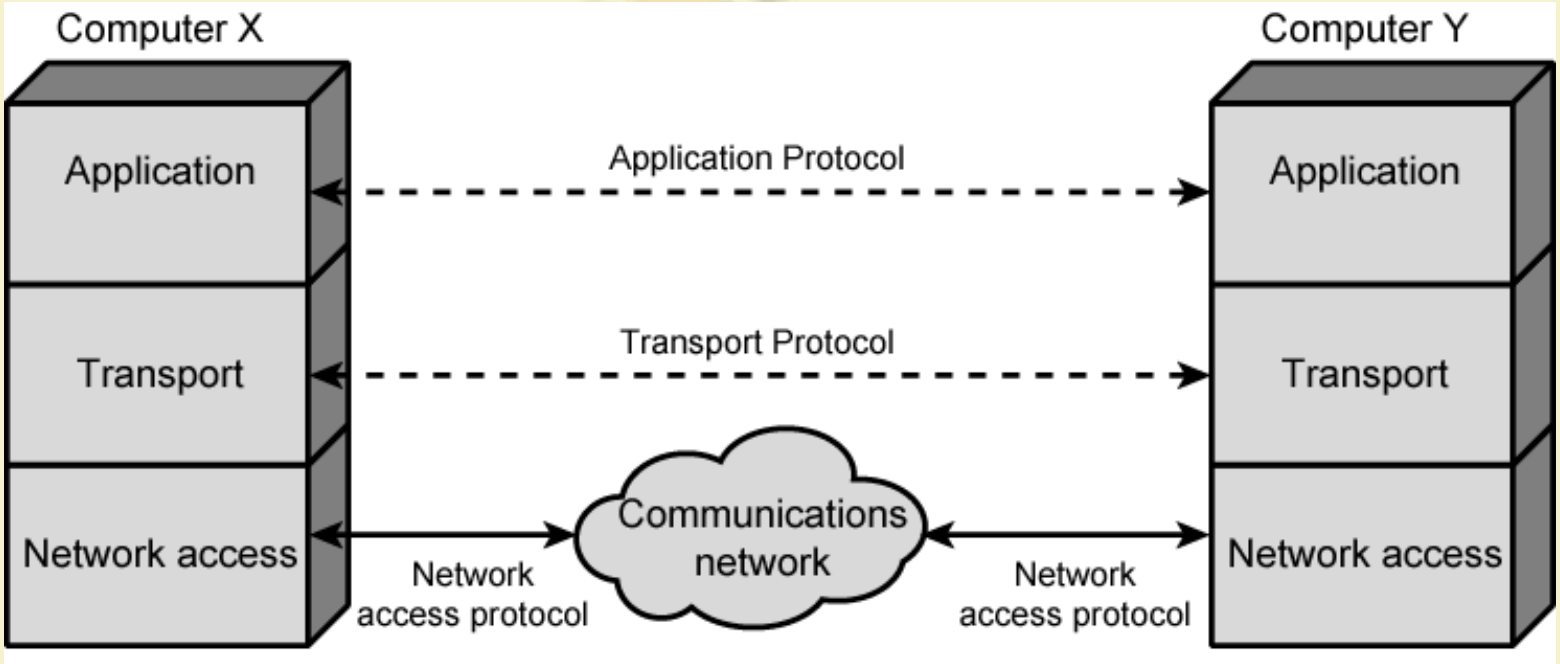
TCP/IP Architecture

TCP/IP : Protocol Architecture and Communication Network



Ref: Data and Computer Communications, by William Stallings

TCP/IP : Protocol Architecture and Communication Network



Ref: Data and Computer Communications, by William Stallings


thank you!