The Internet is a collection of separate and distinct networks, each one operating under a common

framework consisting of globally unique IP addressing and using IP routing and global Border Gateway

Routing (BGP) protocols.1 AnIP address is a string of integers uniquely identifying every host connected

to the Internet; the IP address allows the network to identify first the destination network and then the

host in that network to which a datagram should be delivered.

In addition to the IP or logical address, each network interface, the hardware connecting a host with a

network, has a unique physical or MAC address. Although theMAC address is permanently assigned to

a network interface of the device, the IP address may be dynamically assigned and changes depending

on the network. For example, a laptop will get a different IP addresswhen it is connected to different networks

and a phone will get different IP addresses in differentWi-Fi networks. A host may have multiple

network interfaces and, consequently, multiple IP addresses when it is connected to multiple networks.

The Internet is based on a hourglass network architecture and the TCP/IP protocol family (see

Figure 7.2). The hourglass model captures the fact that all packets transported through the Internet use

Internet Protocol (IP) to reach the destination. IP ensures datagrams’ delivery from the source host to

the destination host based on their IP addresses. A host could be a supercomputer, a workstation, a

laptop, a mobile phone, a network printer, or any other physical device with a network interface.

IP only provides best-effort delivery and its service is characterized as unreliable. Best-effort delivery

means that any router on the path from the source to the destination may drop a packet when it is

overloaded.

The Internet uses two transport protocols: a connectionless datagram protocol, User Datagram

Protocol (UDP), and a connection-oriented protocol, Transport Control Protocol (TCP). The header

of a datagram contains information sufficient for routing through the network from the source to the

destination.

To ensure efficient communication, the UDP transport protocol assumes that error checking and

error correction are either not necessary or performed by the application. Datagrams may arrive out of

order or duplicated, or may not arrive at all. Applications using UDP include the Domain Name System

(DNS), Voice over IP (VoIP), Trivial File Transfer Protocol (TFTP), streaming media applications such

as Internet Protocol television (IPTV), and online games.

Once a packet reaches the destination host, it is delivered to the proper transport protocol daemon,

which, in turn, delivers it to the application that listens to an abstraction of the endpoint of a logical

communication channel called a port (see Figure 7.3). The processes or threads running an application use an abstraction called socket to send and receive data through the network. A socket manages a queue

of incoming messages and one of outgoing messages.

TCP provides reliable, ordered delivery of a stream of bytes from an application on one system to

its peer on the destination system. An application sends/receives data units called segments to/from

a specific port, an abstraction of and endpoint of a logical communication link. TCP is the transport

protocol used by the World Wide Web, email, file transfer, remote administration, and many other

important applications.

TCP uses an end-to-end flow control mechanism based on a sliding window, a range of packets the

sender can send before receiving an acknowledgment from the receiver. This mechanism allows the

receiver to control the rate of segments sent and process them reliably.

A network has a finite capacity to transport data, and when its load is approaching this capacity, we

witness undesirable effects: The routers start dropping packets, and the delays and the jitter increase. An

obvious analogy is a highway on which the time to travel from point A to point B increases dramatically

when the highway becomes congested; a solution for traffic management is to introduce traffic lights

limiting the rate at which new traffic is allowed to enter the highway, and this is precisely what the TCP

emulates.

TCP uses several mechanisms for congestion control (see Section 7.5). These mechanisms control

the rate of data entering the network, keeping the data flow below a rate that would lead to a network collapse and enforce a max/min fair allocation between flows. Section acknowledgment coupled to

timers is used to infer network conditions between the sender and receiver.

TCP congestion control policies are based on four algorithms: slow-start, congestion avoidance, fast

retransmit, and fast recovery. These algorithms use local information, such as the retransmission timeout

(RTO) based on the estimated round-trip time (RTT) between the sender and receiver as well as the

variance in this round-trip time to implement the congestion control policies. UDP is a connectionless

protocol, so there are no means to control the UDP traffic.

The review of basic networking concepts in this section shows why process-to-process communication

incurs a significant overhead. Though raw speed of fiber optic channels can reach Tbps,2 the actual

transmission rate for end-to-end communication over a wide area network can only be of the order of

tens of Mbps and the latency is of the order of milliseconds. This has important consequences for the

development of computer clouds.