# 1.1. Network Packets and Protocols

- A protocol is an agreement about the packets exchanged by communicating programs and what they mean. A protocol tells how packets are structured.
- It turns out to be useful to organize protocols into layers; TCP/IP and virtually all other protocol suites are organized this way. Applications access the services provided by UDP and TCP through the sockets API.
- In TCP/IP, the bottom layer consists of the underlying communication channels. The single network layer protocol in the TCP/IP suite is the Internet Protocol; it solves the problem of making the sequence of channels and routers between any two hosts look like a single host-to-host channel. The Internet Protocol provides a datagram service: every packet is handled and delivered by the network independently. The layer above IP is called the transport layer. It offers a choice between two protocols: TCP and UDP. Both TCP and UDP use addresses, called port numbers, to identify applications within hosts. TCP and UDP are called end-to-end transport protocols because they carry data all the way from one program to another. TCP provides a reliable byte-stream channel so that applications do not have to deal with these problems.

# 1.2. About Addresses

- Internet addresses are binary numbers. They come in two flavors, corresponding to the two versions of the Internet Protocol that have been standardized. The most common type is version 4 (“IPv4,” [14]); the other is version 6 (“IPv6,” [7]), which is just beginning to be deployed.
- Technically, each Internet address refers to the connection between a host and an underlying communication channel—in other words, a network interface. A host may have several interfaces. The port number in TCP or UDP is always interpreted relative to an Internet address.
- Another group of IPv4 addresses reserved for a special purpose includes those reserved for “private use.” This group includes all IPv4 addresses that start with 10 or 192.168, as well as those whose first number is 172 and whose second number is between 16 and 31.
- These addresses were originally designated for use in private networks that are not part of the global Internet. Today they are often used in homes and small offices that are connected to the Internet through a network address translation (NAT) device.

# 1.3. About Names

- Most likely you are accustomed to referring to hosts by name (e.g., host.example.com). However, the Internet protocols deal with addresses (binary numbers), not names.
- The name-resolution service can access information from a wide variety of sources. Two of the primary sources are the Domain Name System (DNS) and local configuration databases.
- The DNS [10] is a distributed database that maps domain names such as www.mkp.com to Internet addresses and other information; the DNS protocol allows hosts connected to the Internet to retrieve information from that database using TCP or UDP. Local configuration

databases are generally OS-specific mechanisms for local name-to-Internet address mappings.

# 1.4. Clients and Servers

- The terms client and server refer to these roles: the client program initiates communication, while the server program waits passively for and then responds to clients that contact it. Together, the client and server compose the application. The terms client and server are descriptive of the typical situation in which the server makes a particular capability—for example, a database service—available to any client that is able to communicate with it.

# 1.5. What Is a Socket?

- A socket is an abstraction through which an application may send and receive data, in much the same way as an open file handle allows an application to read and write data to stable storage. A socket allows an application to plug in to the network and communicate with other applications that are plugged in to the same network. Information written to the socket by an application on one machine can be read by an application on a different machine and vice versa.
- Different types of sockets correspond to different underlying protocol suites and different stacks of protocols within a suite.
- The main types of sockets in TCP/IP today are stream sockets and datagram sockets. Stream sockets use TCP as the end-to-end protocol (with IP underneath) and thus provide a reliable byte-stream service. A TCP/IP stream socket represents one end of a TCP connection. Datagram sockets use UDP (again, with IP underneath) and thus provide a best-effort datagram service that applications can use to send individual messages up to about 65,500 bytes in length. Stream and datagram sockets are also supported by other protocol suites. A TCP/IP socket is uniquely identified by an Internet address, an end-to-end protocol (TCP or UDP), and a port number