At-a-Glance: OSI Model

Why Should I Care About the OSI Model?

The Open Systems Interconnection (OSI) model is a conceptual framework that defines network functions and schemes. The framework simplifies complex network interactions by breaking them into simple modular elements. This open-standards approach allows many independent developers to work on separate network functions, which can then be combined in a "plug-and-play" manner.

The OSI model serves as a guideline for creating and implementing network standards, devices, and internetworking schemes. Advantages of using the OSI model include the following:

- It breaks interrelated aspects of network operation into less-complex elements.
- It enables companies and individual engineers to specialize design and development efforts on modular functions.
- It provides standard interfaces for plug-and-play compatibility and multivendor integration.
- It abstracts different layers of the network from each other to provide easier adoption of new technologies within a layer.

OSI Layers and Definitions

The OSI layers are defined as follows:

Layer 1: Physical

Layer 2: Data link

Layer 3: Network

Layer 4: Transport

Layer 5: Session

Layer 6: Presentation

Layer 7: Application

The four lower layers (called the data flow layers) define connection protocols and methods for exchanging data.

The three upper layers (called the application layers) define how the applications within the end stations communicate with each other and with users.

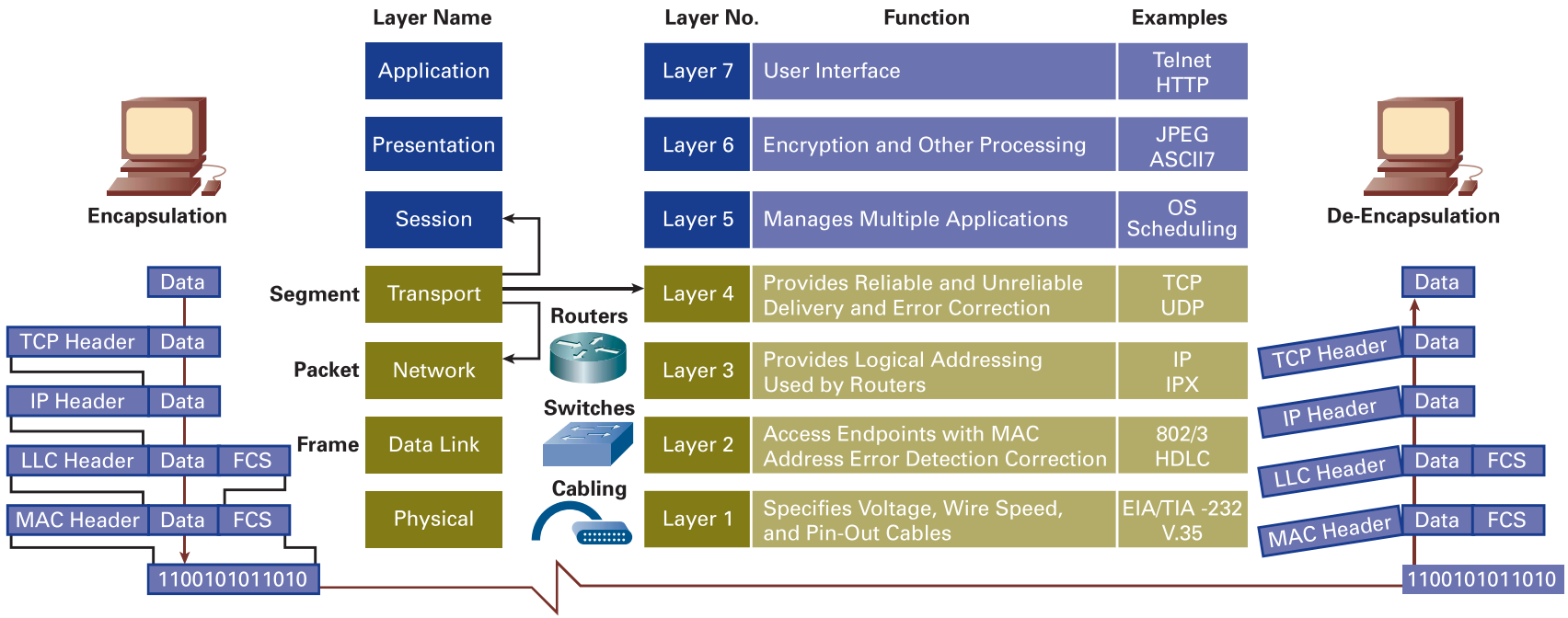
Several mnemonics have been developed to help you memorize the layers and their order. Here's one:

Please Do Not Throw Sausage Pizza Away

What Problems Need to Be Solved?

An OSI layer can communicate only with the layers immediately above and below it on the stack, and with its peer layer on another device. A process must be used so that information (including data and stack instructions) can be passed down the stack, across the network, and back up the stack on the peer device.

At-a-Glance: OSI Model



Communicating Between Layers

Each layer of the OSI model uses its own protocol to communicate with its peer layer in the destination device. The OSI model specifies how each layer communicates with the layers above and below it, allowing vendors to focus on specific layers that will work with any other vendor's adjacent layers.

Information is exchanged between layers using protocol data units (PDU). PDUs include control information (in the form of headers and trailers) and user data. PDUs include different types of information as they go up or down the layers (called "the stack"). To clarify where the PDU is on the stack, it is given a distinct name at each of the lower levels.

In other words, a PDU that is a segment (Layer 4) includes all the application layer's information. A packet (Layer 3) includes network layer control information in addition to the data and control information contained at the transport layer. Similarly, a frame (Layer 2) is a PDU that includes data link layer control information in addition to the upper layer control information and data. Finally, PDUs at the physical layer (Layer 1) are called bits.

At-a-Glance: OSI Model

Encapsulation

The process of passing data down the stack using PDUs is called data encapsulation. Encapsulation works as follows: When a layer receives a PDU from the layer above it, it encapsulates the PDU with a header and trailer and then passes the PDU down to the next layer. The control information that is added to the PDU is read by the peer layer on the remote device. Think of this as like putting a letter in an envelope, which has the destination address on it. The envelope is then put in a mailbag with a zip code on it. The bag is then placed in large box with a city name on it. The box is then put on a plane for transport to the city.

De-encapsulation

De-encapsulation, the opposite of encapsulation, is the process of passing information up the stack. When a layer receives a PDU from the layer below, it does the following:

1. It reads the control information provided by the peer source device.
2. The layer strips the control information (header) from the frame.
3. It processes the data (usually passing it up the stack).

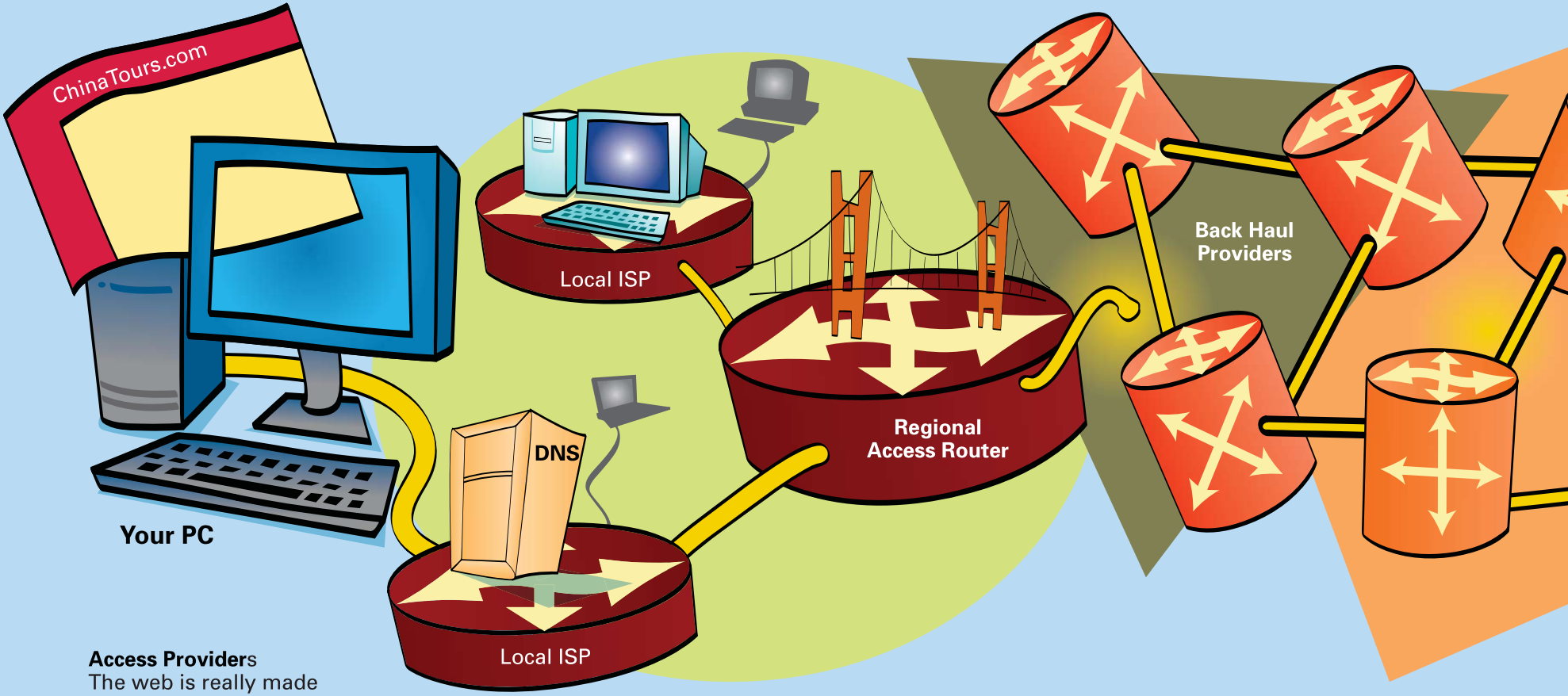
Each subsequent layer performs this same de-encapsulation process. To continue the preceding example, when the plane arrives, the box of mail is removed from the plane. The mailbags are taken out of the boxes and are sent to the correct post office. The letters are removed from the mailbags and are delivered to the correct address. The intended recipient opens the envelope and reads the letter.

Extra Layers?

Discussions among technical purists can often lead to philosophical or budgetary debates that can quickly derail otherwise-productive meetings. These discussions are often referred to as Layer 8 (political) and Layer 9 (financial) debates. Although these layers are not really part of the OSI model, they are usually the underlying cause of heated technology arguments.

Another common joke among networking professionals is the type of networking problem referred to as a "Layer 8 issue." Because the network, computers, and applications stop at Layer 7, Layer 8 sometimes represents the end user actually using the system. So if you hear your IT person snicker to his colleagues that your IT trouble ticket is closed and it was a "Layer 8 issue," the IT person is referring to you.

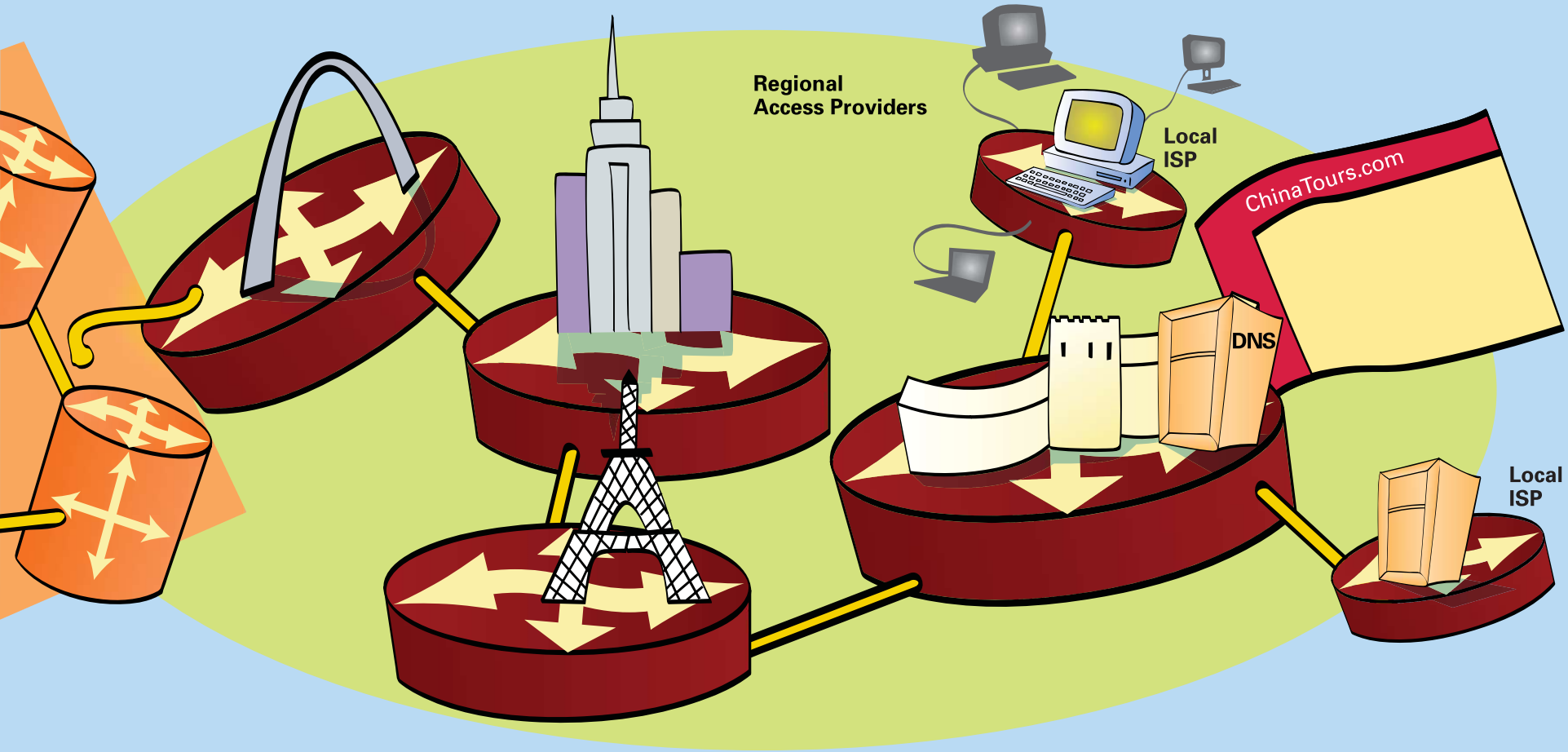
Internet Infrastructure: How It All Connects



Access Providers
The web is really made of many networks connected in a hierarchy. Local Internet service providers (ISPs) typically give residential and small business access to the Internet. Regional providers typically connect several local ISPs to each other and to back haul providers that connect with other regional providers.

Domain Name Server (DNS)
This server maps domain names to their IP addresses. One of the reasons that the Internet has taken off in use and popularity is because www.cisco.com is much easier to remember than 25.156.10.4.

Internet Infrastructure: How It All Connects



Web Servers
All web pages are stored on computers called web servers. Thousands of these servers can be dedicated servers for companies, hosting servers that house many personal pages, or even single computers housing individual pages.

Back Haul Providers
A few back haul providers comprise the high-speed backbone of the Internet. Only a handful of these providers are capable of handling the massive amounts of Internet traffic that continues to grow. Many parts of the back haul providers overlap with each other, which improves both the speed and reliability of the network.

Computers Speaking the Same Language

The Internet protocols comprise the most popular, nonproprietary data-networking protocol suite in the world. The Internet protocols are communication protocols used by electronic devices to talk to each other. Initially, computers were the primary clients of IP protocols, but other types of electronic devices can connect to IP networks, including printers, cellular phones, and MP3 players. Today, even common devices such as vending machines, dishwashers, and cars are being connected to IP networks.

The two best-known Internet protocols are Transmission Control Protocol (TCP) and Internet Protocol (IP). The Defense Advanced Research Projects Agency (DARPA) developed the Internet protocols in the mid-1970s. DARPA funded Stanford University and Bolt, Beranek, and Newman (BBN) to develop a set of protocols that would allow different types of computers at various research locations to communicate over a common packet-switched network. The result of this research produced the Internet protocol suite, which was later distributed for free with the Berkeley Software Distribution (BSD) UNIX operating system.

From there, IP became the primary networking protocol, serving as the basis for the World Wide Web (WWW) and the Internet in general. Internet protocols are discussed and adopted in the public domain. Technical bulletins called Requests for Comments (RFC) documents proposed protocols and practices. These documents are reviewed, edited, published, and analyzed, and then are accepted by the Internet community (this process takes years).

The Internet protocol suite also comprises application-based protocols, including definitions for the following:

- Electronic mail (Simple Mail Transfer Protocol [SMTP])
- Terminal emulation (Telnet)
- File transfer (File Transfer Protocol [FTP])
- HTTP

IP is considered a Layer 3 protocol according to the OSI model, and TCP is a Layer 4 protocol.

What Is an Address?

For computers to send and receive information to each other, they must have some form of addressing so that each end device on the network knows what information to read and what information to ignore. This capability is important both for the computers that ultimately use the information and for the devices that deliver information to end stations, such as switches and routers. Every computer on a network has two addresses:

- **MAC address:** A manufacturer-allocated ID number (such as a global serial number) that is permanent and unique to every network device on Earth. MAC addresses are analogous to a social security number or other national identification number. You have only one, it stays the same wherever you go, and no two people (devices) have the same number. MAC address are formatted using six pairs of hexadecimal numbers, such as 01-23-45-67-89-AB.

  Hexadecimal or "hex" is a base 16 numbering scheme that uses the numbers 0 through 9 and the letters A through F to count from 0 to 15. This might seem odd, but it provides an easy translation from binary (which uses only 1s and 0s), which is the language of all computers.

- **IP address:** This address is what matters most to basic networking. Unlike a MAC address, the IP address of any device is temporary and can be changed. It is often assigned by the network itself and is analogous to your street address. It only needs to be unique within a network. Someone else's network might use the same IP address, much like another town might have the same street (for example, 101 Main Street). Every device on an IP network is given an IP address, which looks like this: 192.168.1.100.

The format of this address is called dotted-decimal notation. The period separators are pronounced "dot," as in one ninety two dot one sixty eight dot...." Because of some rules with binary, the largest number in each section is 255.

In addition to breaking up the number, the dots that appear in IP addresses allow us to break the address into parts that represent networks and hosts. In this case, the "network" portion refers to a company, university, government agency, or your private network. The hosts would be the addresses of all the computers on the individual network. If you think of the network portion of

the address as a street, the hosts would be all the houses on that street. If you could see the IP addresses of everyone who is on the same network segment as you, you would notice that the network portion of the address is the same for all computers, and the host portion changes from computer to computer. An example will probably help. Think of an IP address as being like your home address for the post office: state.city.street.house-number.

Each number in the IP address provides a more and more specific location so that the Internet can find your computer among millions of other computers. The Internet is not organized geographically like the postal system, though. The components of the address (intentionally oversimplified) are major-network.minor-network.local-network.device.

Dynamically Allocated IP Addresses

A network administrator is responsible for assigning which devices receive which IP addresses in a corporate network. The admin assigns an IP address to a device in one of two ways: by configuring the device with a specific address or by letting the device automatically learn its address from the network. Dynamic Host Configuration Protocol (DHCP) is the protocol used for automatic IP address assignment. Dynamic addressing saves considerable administrative effort and conserves IP addressing space. It can be difficult to manually administer IP addresses for every computer and device on a network. Most networks use DHCP to automatically assign an available IP address to a device when it connects to the network. Generally, devices that don't move around receive fixed addresses, known as static addressing. For example, servers, routers, and switches usually receive static IP addresses. The rest use dynamic addressing. For home networks you do not need a network administrator to set up your address; instead, a home broadband router allocates IP addresses via DHCP.

Domain Names and Relationship to IP Addresses

Because IP addresses are difficult to remember in their dotted-decimal notation, a naming convention called domain names was established that's more natural for people to use. Domain names such as www.cisco.com are registered and associated with a particular public IP address. The Domain Name System (DNS) maps a readable name to an IP address. For example, when you enter http://www.cisco.com into a browser, the PC uses the DNS protocol to contact a DNS name server. The name server translates the name http://www.cisco.com into the actual IP address for that host.

At-a-Glance: TCP/IP

Why Should I Care About TCP/IP?

TCP/IP is the best-known and most popular protocol suite used today. Its ease of use and widespread adoption are some of the best reasons for the Internet explosion that is taking place.

Encompassed within the TCP/IP protocol is the capability to offer reliable, connection-based packet transfer (sometimes called synchronous) as well as less reliable, connectionless transfers (also called asynchronous).

What Problems Need to Be Solved?

TCP is a connection-oriented, reliable protocol that breaks messages into segments and reassembles them at the destination station (it also resends packets not received at the destination). TCP also provides virtual circuits between applications.

A connection-oriented protocol establishes and maintains a connection during a transmission. The protocol must establish the connection before sending data. As soon as the data transfer is complete, the session is torn down.

User Datagram Protocol (UDP) is an alternative protocol to TCP that also operates at Layer 4. UDP is considered an "unreliable," connectionless protocol. Although "unreliable" may have a negative connotation, in cases where real-time information is being exchanged (such as a voice conversation), taking the time to set up a connection and resend dropped packets can do more harm than good.

Endpoints in TCP/IP are identified by IP addresses. IP addressing is covered in the next At-a-Glance.

TCP/IP Datagrams

TCP/IP information is sent via datagrams. A single message may be broken into a series of datagrams that must be reassembled at their destination. Three layers are associated with the TCP/IP protocol stack:

- **Application layer:** This layer specifies protocols for e-mail, file transfer, remote login, and other applications. Network management is also supported.
- **Transport layer:** This layer allows multiple upper-layer applications to use the same data stream. TCP and UDP protocols provide flow control and reliability.
- **Network layer:** Several protocols operate at the network layer, including IP, ICMP, ARP, and RARP.

IP provides connectionless, best-effort routing of datagrams.

TCP/IP hosts use Internet Control Message Protocol (ICMP) to carry error and control messages with IP datagrams. For example, a process called ping allows one station to discover a host on another network.

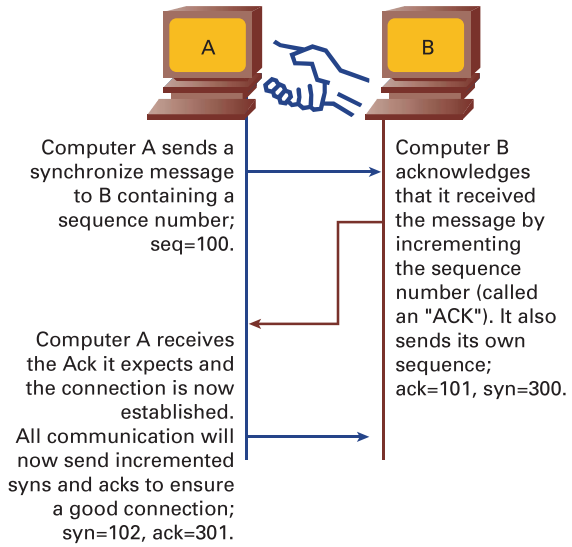
Address Resolution Protocol (ARP) allows communication on a multiaccess medium such as Ethernet by mapping known IP addresses to MAC addresses.

Reverse Address Resolution Protocol (RARP) is used to map a known MAC address to an IP address.

How TCP Connections Are Established

End stations exchange control bits called SYN (for synchronize) and Initial Sequence Numbers (ISN) to synchronize during connection establishment. TCP/IP uses what is known as a three-way handshake to establish connections.

To synchronize the connection, each side sends its own initial sequence number and expects to receive a confirmation in an acknowledgment (ACK) from the other side. The following figure shows an example.

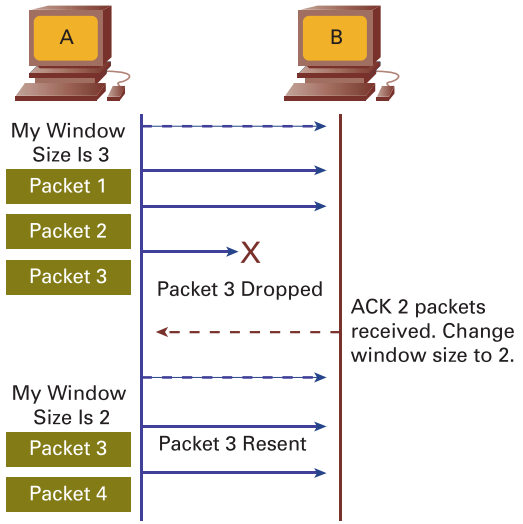


At-a-Glance: TCP/IP

TCP Windowing

One way to structure a communications protocol is to have the receiver acknowledge every packet received from a sender. Although this is the most reliable method, it can add unnecessary overhead, especially on fairly reliable connection media. Windowing is a compromise that reduces overhead by acknowledging packets only after a specified number have been received.

The window size from one end station informs the other side of the connection how much it can accept at one time. With a window size of 1, each segment must be acknowledged before another segment is sent. This is the least efficient use of bandwidth. A window size of 7 means that an acknowledgment needs to be sent after the receipt of seven segments; this allows better utilization of bandwidth. A windowing example is shown in the figure.



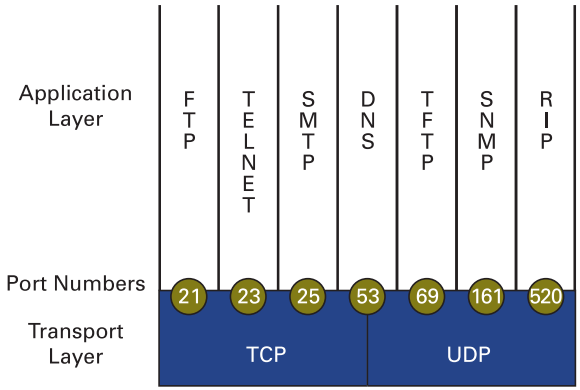
UDP

UDP is a connectionless, unreliable Layer 4 protocol. Unreliable in this sense means that the protocol does not ensure that every packet will reach its destination. UDP is used for applications that provide their own error recovery process or when retransmission does not make sense. UDP is simple and efficient, trading reliability for speed.

Why not resend? It may not be obvious why you would not resend dropped packets if you had the option to do so. However, real-time applications such as voice and video could be disrupted by receiving old packets out of order. For example, suppose a packet containing a portion of speech is received 2 seconds later than the rest of the conversation. Playing the sound out into the earpiece probably will sound like poor audio quality to the user, because the user is listening further into the conversation. In these cases, the application usually can conceal the dropped packets from the end user so long as they account for a small percentage of the total.

Port Numbers

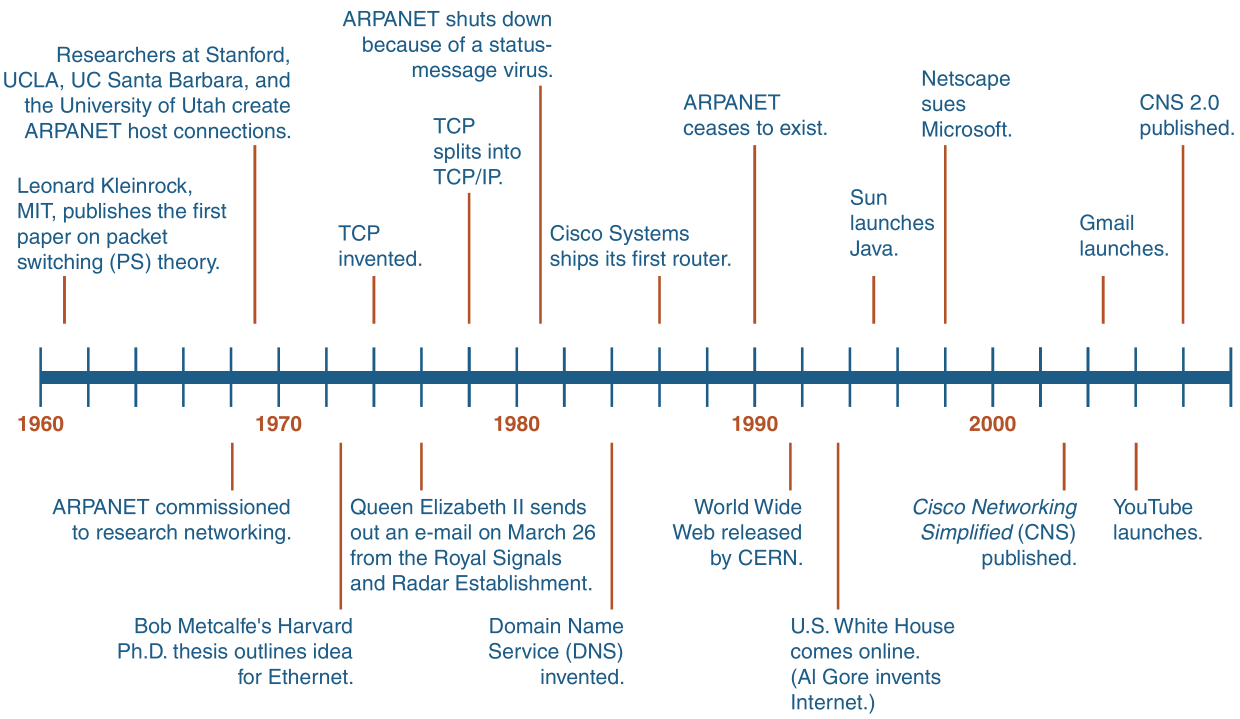
TCP and UDP can send data from several upper-layer applications on the same datagram. Port numbers (also called socket numbers) are used to keep track of different conversations crossing the network at any given time. Some of the more well-known port numbers are controlled by the Internet Assigned Numbers Authority (IANA). For example, Telnet is always defined by port 23. Applications that do not use well-known port numbers have numbers randomly assigned from a specific range.



The use of port numbers is what allows you to watch streaming video on your computer while checking e-mails and downloading documents from a web page all at the same time. All three may use TCP/IP, but use of a port number allows the applications to distinguish which are video and which are e-mail packets.

At-a-Glance: TCP/IP

History of TCP/IP



At-a-Glance: IP Addressing

Why Should I Care About IP Addressing?

Behind every website, Universal Resource Locator (URL), and computer or other device connected to the Internet is a number that uniquely identifies that device. This unique identifier is called an IP address. These addresses are the key components of the routing schemes used over the Internet. For example, if you are downloading a data sheet from www.cisco.com to your computer, the header of the packets comprising the document includes both the host address (in this case, the IP address of Cisco's public server) and the destination address (your PC).

What Problems Need to Be Solved?

Each IP address is a 32-bit number, which means that there are about 4.3 trillion address combinations. These addresses must be allocated in a way that balances the need for administrative and routing efficiency with the need to retain as many usable addresses as possible.

Dotted decimal: The most common notation for describing an IP address is dotted decimal. Dotted decimal breaks a 32-bit binary number into four 8-bit numbers (represented in decimal form), which is called an *octet*. Each octet is separated by a period, which aids in the organizational scheme to be discussed. For example, the binary address 00001010100000001011001000101110 can be represented in dotted decimal as 10.128.178.46.

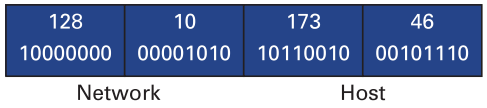
Logical Versus Physical

MAC addresses are considered physical addresses because they are assigned to pieces of hardware by the manufacturer and cannot be reassigned.

IP addresses are assigned by a network administrator and have meaning only in a TCP/IP network. These addresses are used solely for routing purposes and can be reassigned.

Host and network: Rather than assigning numbers at random to various endpoints (which would be extremely difficult to manage), every company and organization listed on the Internet is given a block of public address numbers to use. This is accomplished by using a two-part addressing scheme that identifies a network and host. This two-part scheme allows the following:

- All the endpoints within a network share the same network number.
- The remaining bits identify each host within that network.



In the figure, the first two octets (128.10) identify a company with an Internet presence (it's the address of the router that accesses the Internet). All computers and servers within the company's network share the same network address. The next two octets identify a specific endpoint (computer, server, printer, and so on). In this example the company has 65,536 addresses it can assign (16 bits, or 2^{16}). Therefore, all devices in this network would have an address between 128.10.0.1 and 128.10.255.255.

Address Classes

When the IP address scheme was developed, only the first octet was used to identify the network portion of the address. At the time it was assumed that 254 networks would be more than enough to cover the research groups and universities using this protocol. As usage grew, however, it became clear that more network designations would be needed (each with fewer hosts). This issue led to the development of address classes.

Addresses are segmented into five classes (A through E). Classes A, B, and C are the most common. Class A has 8 network bits and 24 host bits. Class B has 16 network bits and 16 host bits, and Class C has 24 network bits and 8 host bits. This scheme was based on the assumption that there would be many more small networks (each with fewer endpoints) than large networks in the world. Class D is used for multicast, and Class E is reserved for research. The following table breaks down the three main classes. Note that the Class A address starting with 127 is reserved.

At-a-Glance: IP Addressing

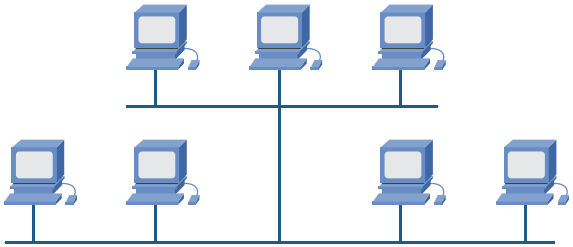
Classes	First Octet Range	Network Bits	Possible Networks	Host Bits	No. of Hosts per Network
A	1–126	8	126	24	16,777,216
B	128–191	16	16,384	16	65,536
C	192–223	24	2,097,152	8	256

The total number of available hosts on a network can be derived by using the formula 2^n–2, where n is the number of host bits. The –2 accounts for an octet with all 0s, which is reserved for network identification, and all 1s, which is reserved for sending a broadcast message to all hosts.

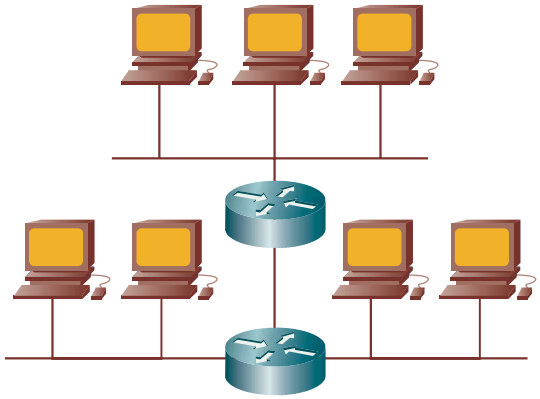
Subnetting

Subnetting is a method of segmenting hosts within a network and providing additional structure. Without subnets, an organization operates as a flat network. These flat topologies result in short routing tables, but as the network grows, the use of bandwidth becomes inefficient.

In the figure, a Class B network is flat, with a single broadcast and collision domain. Collision domains are explained in more detail in the Ethernet chapter. For now, just think of them as a small network segment with a handful of devices. Adding Layer 2 switches to the network creates more collision domains but does not control broadcasts.



In the next figure, the same network has been subdivided into several segments or subnets. This is accomplished by using the third octet (part of the host address space for a Class B network) to segment the network. Note that the outside world sees this network the same as in the previous figure.



Subnetting is a bit complex at first pass. Think of it like a street address. For a house, the street address may provide the needed addressability to reach all the house's occupants. Now consider an apartment building. The street address only gets you to the right building. You need to know in which apartment the occupant you are seeking resides. In this crude example, the apartment number acts a bit like a subnet.

Subnet Masks

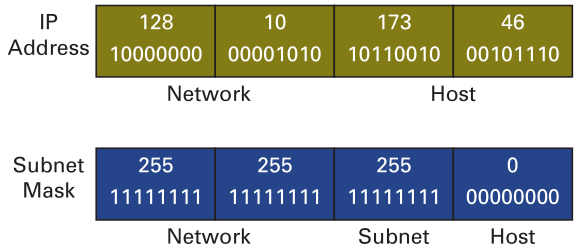
Routers use a subnet mask to determine which parts of the IP address correspond to the network, the subnet, and the host. The mask is a 32-bit number in the same format as the IP address. The mask is a string of consecutive 1s starting from the most-significant bits, representing the network ID, followed by a string of consecutive 0s, representing the host ID portion of the address bits.

At-a-Glance: IP Addressing

Each address class has a default subnet mask (A = /8, B = /16, C = /24). The default subnet masks only the network portion of the address, the effect of which is no subnetting. With each bit of subnetting beyond the default, you can create 2^n–2 subnets. The preceding example has 254 subnets, each with 254 hosts. This counts the address ending with .0, but not the address ending in .255.

Continuing with the preceding analogy, the subnet mask tells the network devices how many apartments are in the building.

Identifying Subnet Addresses



This subnet mask can also be written as "/24", where 24 represents the number of 1s in the subnet mask.

Given an IP address and subnet mask, you can identify the subnet address, broadcast address, and first and last usable addresses within a subnet as follows:

1. Write down the 32-bit address and the subnet mask below that (174.24.4.176/26 is shown in the following figure).
2. Draw a vertical line just after the last 1 bit in the subnet mask.
3. Copy the portion of the IP address to the left of the line. Place all 1s for the remaining free spaces to the right. This is the broadcast address for the subnet.
4. The first and last address can also be found by placing ...0001 and ...1110, respectively, in the remaining free spaces.
5. Copy the portion of the IP address to the left of the line. Place all 0s for the remaining free spaces to the right. This is the subnet number.

```
174.24.4.176     10101110 00011000 00000100 10110000  Host
255.255.255.192  11111111 11111111 11111111 11000000  Mask
174.24.4.128     10101110 00011000 00000100 10000000  Subnet
174.24.4.191     10101110 00011000 00000100 10111111  Broadcast
```