Overview of Microsoft TCP/IP for Windows for Workgroups

TCP/IP is a networking protocol that provides communication across interconnected networks, between computers with diverse hardware architectures and various operating systems. TCP/IP can be used with Windows for Workgroups or to connect to Microsoft LAN Manager or non-Microsoft (for example, UNIX®) hosts.

What Is TCP/IP for Windows for Workgroups?

How Does TCP/IP Work?

What Is TCP/IP for Windows for Workgroups?

The TCP/IP protocol family is a standard set of networking protocols, or rules, that govern how data is passed between systems on a network. Microsoft TCP/IP for Windows for Workgroups enables enterprise networking and connectivity on your Windows for Workgroups-based desktop system. Adding TCP/IP to a Windows for Workgroups configuration offers the following advantages:

- A standard, routable, enterprise networking protocol for Windows for Workgroup services
- An architecture that facilitates connection to foreign systems
- A robust, cross-platform client-server framework

How Does TCP/IP Work?

The name TCP/IP is somewhat confusing since TCP and IP are really only two protocols in the family of Internet protocols. Over time, however, TCP/IP has been used in industry to denote the family of common Internet protocols.

How do these protocols work and what do they do? The following sections briefly explain how the TCP and IP protocols work.

How TCP Works

How IP Works

Example

How TCP Works

TCP is a reliable, *connection-oriented* protocol. Connection-oriented implies that TCP first establishes a connection between the two systems that intend to exchange data. Since most networks are built on shared media, it is necessary to break chunks of data into packets so that no two communicating systems monopolize the network. When an application sends a message to TCP for transmission, TCP breaks the message into packets, sized appropriately for the network, and sends them over the network.

Sequence Numbers, Checksum, and Port ID

Because a single message is often broken into many packets, TCP marks these packets with *sequence numbers* before sending them. The sequence numbers allow the receiving system to properly reassemble the packets into the original message.

Being able to reassemble the original message is not enough--the accuracy of the data must also be verified. TCP does this by computing a checksum. The recipient then does the same calculation on the received data and compares the result with the checksum that the sender computed. If the results match, the recipient sends an acknowledgment (ACK). If the results do *not* match, the recipient asks the sender to resend the packet.

Finally, TCP uses *port IDs* to specify which application running on the system is sending or receiving the data.

TCP Headers

The port ID, checksum, and sequence number are inserted into the TCP packet in a special section called the *header*. The header is at the beginning of the packet containing this and other control information for TCP.

How IP Works

IP is the *messenger protocol* of TCP/IP. The IP protocol, much simpler than TCP, basically addresses and sends packets. IP relies on three pieces of information, which you provide, to receive and deliver packets successfully:

- IP address
- Subnet mask
- Default gateway

IP Addresses

The IP address identifies your system on the TCP/IP network. Although an IP address is a single value, it really contains two pieces of information:

- Your systems network ID
- Your systems host (or system) ID

Subnet Mask

The *subnet mask*, represented in dotted decimal notation, is used to extract the network ID and host ID from your IP address. The value of the subnet mask is determined by setting the network ID bits of the IP address to 1s and the host ID bits to 0s. The result allows TCP/IP to determine the host and network IDs of the local workstation. For example:

When the IP address is:	`102.54.94.97`
And the subnet mask is:	`255.255.0.0`
The network ID is:	`102.54`
And the host ID is:	`94.97`

Default Gateway

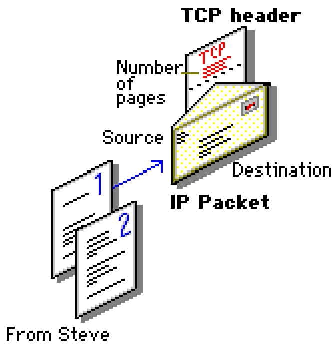
The default gateway is needed only for systems that are part of an internet. When IP gets ready to send a packet on the wire, it inserts the local (source) IP address and destination address of the packet in the IP header, and verifies that the network ID of the destination matches the source. If they match, the packet is sent directly to the destination system on the local network.

If the network IDs do *not* match, the packet is forwarded to the default gateway for delivery. Since the default gateway has knowledge of the network IDs of the other networks in the internet, it forwards the packet to other gateways until the packet is eventually delivered to a gateway connected to the specified destination. This process is known as *routing*.

Here is an example to show how TCP/IP might deliver a message. To keep things simple, the example uses an analogy with the U.S. postal system, to describe how these protocols work.

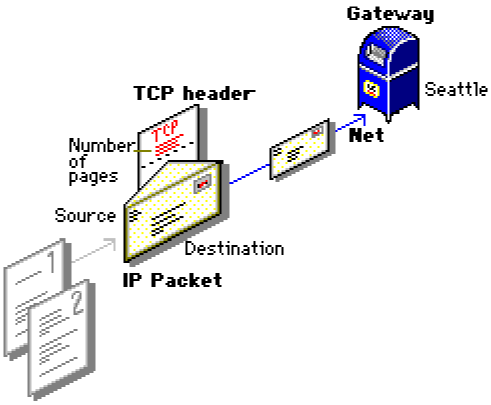
Steve (source host ID) in Seattle (source network ID) wants to send a two page letter (message) to Dana (destination host ID) in Dartmouth (destination network ID). There is a limit to the length of the message that can be sent in a single letter (maximum transmission unit, or MTU).

Continue



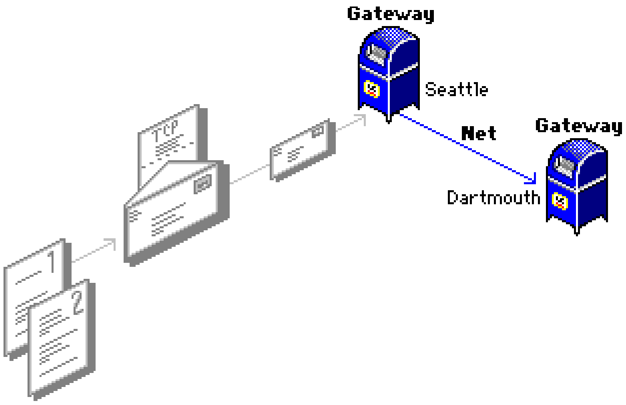
Steve realizes that his letter (message) is too long to fit in a single envelope (IP packet), tears off the first page (TCP packet) and writes *1 of 2* in the margin (TCP header), puts it into an envelope (IP packet), addresses it to Dana in Dartmouth (destination IP address), and mails it.

Continue



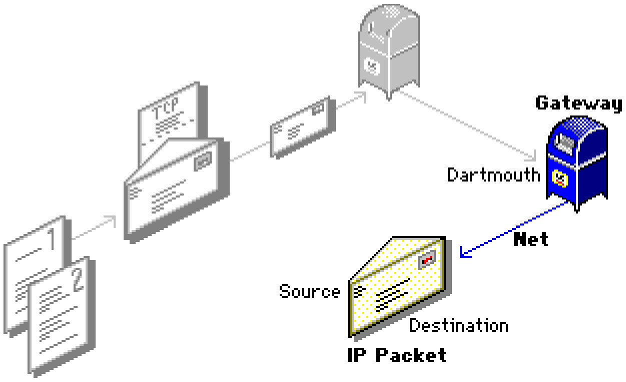
Steves mail carrier (Steves default gateway) picks up the letter, doesnt know where Dartmouth is, and forwards it to the Seattle post office (gateway).

Continue



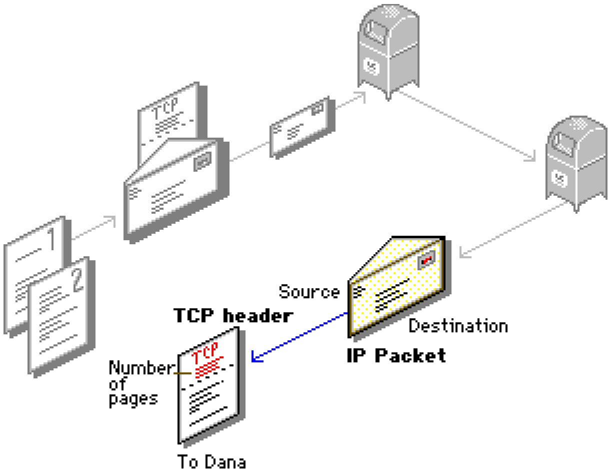
From Seattle, the messages goes (routes) to the Dartmouth post office (Danas default gateway).

Continue



The Dartmouth mail carrier delivers it to Dana.

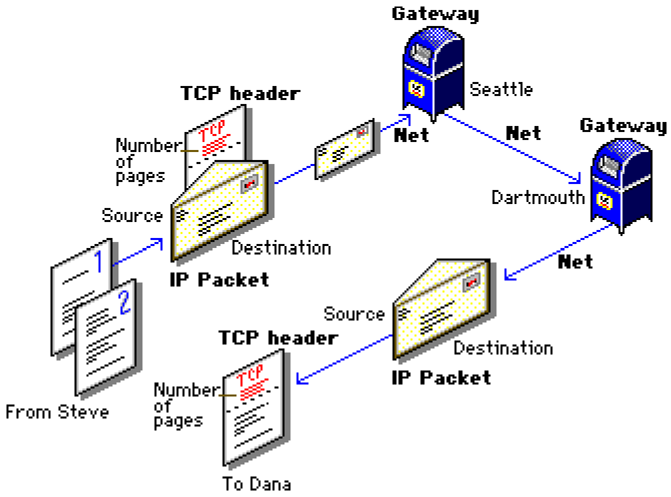
Continue



Dana opens the envelope and reads the contents.

The story is so compelling that she gets impatient after waiting for the second message and mails a quick note (ACK) to Steve: *Steve, I received the first page of the message, but I havent seen any since. Are you okay?* Her note (ACK) travels back the same way Steve's message came.

Continue



Steve reads Danas letter and immediately takes the second page (TCP packet), writes *2 of 2* in the margin (TCP header), puts it in an envelope addressed to Dana (IP packet), and mails it.

The second note is also successfully routed to Dana, so she replies (ACKs) with one final note: *Good story, Steve. Thanks for the laughs.*

Done

Transmission Control Protocol

Internet Protocol

Application Programming Interface (API)

File Transfer Protocol (FTP)

Terminal Emulation Protocol (Telnet)

TCP/IP is viewed as the most complete and accepted networking protocol available. Virtually all modern operating systems offer TCP/IP support, and most large networks rely on TCP/IP for all their network traffic.

Because most operating systems offer TCP/IP, many standard utilities have been designed to access and transfer data between heterogeneous environments. The Windows Sockets interface offers compatibility with many foreign host connectivity products. Several applications vendors support this API (Application Programming Interface) standard.

TCP/IP for Windows for Workgroups offers the Windows Sockets interface, which is ideal for developing client-server applications. A Windows Sockets application developed to be used with Microsoft TCP/IP will be able to run other vendors Windows Sockets-compliant stacks as well.

(for example, several systems sharing the same cabling)

Packets

Manageable pieces of data.

Checksum

A checksum is a simple mathematical computation applied, by the sender, to the data contained in the TCP packet.

Network ID

Identifies a group of systems that are all located on the same physical network.

Host ID

Identifies your system within a particular network ID.

Gateway

In *internetworks* (networks formed by a collection of networks), there are as many unique network IDs as there are networks. TCP/IP networks are connected by *gateways* (or *routers*), which have knowledge of the networks that are connected in the internet.

IP Address

An IP address is a unique 32-bit address, represented in dotted decimal notation, that identifies a computer on a network. An IP address looks like this:

```
102.54.94.97
```

A **subnet mask** is used to extract the host ID and the network ID from an IP address.

Transmission Control Protocol/Internet Protocol (TCP/IP)