

Suppose your network connection supports both IPX/SPX and TCP/IP protocols; however, your primary interface is using TCP/IP. In that case, you would move that protocol to the top for best performance. When connecting to a server, the client defines the protocol being used. Setting the order of the protocols on a server therefore does not enhance performance, whereas changing the order of the protocols on your Windows 2000 Professional client will impact performance. Deselecting a network protocol will make it temporarily unavailable without uninstalling it.

CONFIGURING THE TCP/IP PROTOCOL

Configure and troubleshoot the TCP/IP protocol.

TCP/IP is the default protocol for Windows 2000 Professional and is supported by most common operating systems. TCP/IP is a suite of protocols used to provide connectivity within an enterprise network (LAN) in addition to providing connectivity to the Internet. When you manually configure a computer with a TCP/IP network adapter, you must enter the appropriate settings for connectivity with your network.

IP Addressing

Before delving into IP addressing schemes, it is appropriate to review binary-to-decimal conversions. IP addresses are 32-bit integers that are usually depicted as four 8-bit numbers. This can be thought of as a series of 1s or 0s, with eight taken together to be a number. Each position in the 8 bits (from right to left) is twice the value of the field before it. (In decimal notation, the same rule would state that each column is worth 10 times the value of the previous column—100s versus 10s versus 1s.) The smallest integer number that can be represented with 8 bits is 0 0 0 0 0 0 0 0 (2^0-1), or 0. The largest integer that can be represented by 8 bits is 1 1 1 1 1 1 1 1 (2^8-1), or 255. Because of this, you will always see IP addresses as four numbers ranging from 0 to 255.

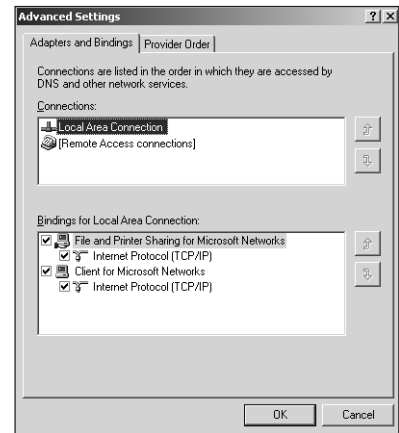


FIGURE 6.10

Typical protocols available.

NOTE

Unnecessary Protocols For maximum performance, remove any unnecessary protocols and always make sure that your most frequently used protocol is configured to be the first one accessed.

Each TCP/IP connection must be identified by an address. The address is a 32-bit number that is used to uniquely identify a host on a network. The TCP/IP address has no dependence on the Data-Link layer address such as the MAC address of a Network adapter.

Although the IP address is 32 bits, it is customary to break it into four 8-bit numbers expressed in decimal and separated by dots. This can be referred to in dotted decimal format and is expressed as $w.x.y.z$. The value of breaking down this address into four 8-bit values can be seen in the following example. Suppose you have an address that is 192.168.8.4. If you had to remember that as a binary 32-bit number it would be 1100000010101000000010000000100 or, converted to a decimal number, it would be the sum of $2^3+2^{12}+2^{20}+2^{22}+2^{24}+2^{31}+2^{32}$, which is $8+4,096+1,048,576+4,194,304+16,777,216+2,147,483,648+4,294,967,296$, or 6,464,475,144. 192.168.8.4 is definitely easier to remember.

This addressing scheme is again broken down into two halves: a network ID (also known as the network address) and the host ID (also known as the host address). The network ID must be unique in the Internet or intranet, and the host ID must be unique to the network ID. The network portion of the $w.x.y.z$ notation is separated from the host through the use of the subnet mask. See the section entitled “Subnet Mask” later in this chapter.

The Internet community was originally divided into five address classes. Microsoft TCP/IP supports class A, B, and C addresses assigned to hosts.

The class of address defines which bits are used for the network ID and which bits are used for the host ID. It also defines the possible number of networks and the number of hosts per network. Here is a rundown of the five classes:

- ◆ **Class A addresses.** The high-order bit is always binary 0 and the next seven bits complete the network ID. The next three octets define the host ID. This represents 126 networks with 16,777,214 hosts per network.
- ◆ **Class B addresses.** The top two bits in a class B address are always set to binary 1 0. The next 14 bits complete the network ID. The remaining two octets define the host ID. This represents 16,384 networks with 65,534 hosts per network.

- ◆ **Class C addresses.** The top three bits in a class C address are always set to binary 1 1 0. The next 21 bits define the network ID. The remaining octet defines the host ID. This represents 2,097,152 networks with 254 hosts per network.
- ◆ **Class D addresses.** Class D addresses are used for multicasting to a number of hosts. Packets are passed to a selected subset of hosts on a network. Only those hosts registered for the multicast address accept the packet. The four high-order bits in a class D address are always set to binary 1 1 1 0. The remaining bits are for the address that interested hosts will recognize.
- ◆ **Class E addresses.** Class E is an experimental address that is reserved for future use. The high-order bits in a class E address are set to 1 1 1 1.

Table 6.2 indicates how the three classes supported by Microsoft TCP/IP divide up network IDs and host IDs.

| |
|------------------|
| TABLE 6.2 |
|------------------|

CLASS ADDRESS RANGES

| <i>Class</i> | <i>Network ID</i> | <i>Network Portion</i> | <i>Host Portion</i> | <i>Number of Networks</i> | <i>Number of Hosts</i> |
|--------------|-------------------|------------------------|---------------------|---------------------------|------------------------|
| A | 1.126 | w. | x.y.z | 126 | 16,777,214 |
| B | 128.191 | w.x | y.z | 16,384 | 65,534 |
| C | 192.223 | w.x.y | z | 2,097,152 | 254 |

Subnet Mask

Once an IP address from a particular class has been decided upon, it is possible to divide it into smaller segments to better utilize the addresses available. Each segment is bounded by an IP router and assigned a new subnetted network ID that is a subset of the original class-based network ID.

A subnet mask (also known as an address mask) is defined as a 32-bit value that is used to distinguish the network ID from the host ID in an IP address. The bits of the subnet mask are defined as follows:

- ◆ All bits that correspond to the network ID are set to 1.
- ◆ All bits that correspond to the host ID are set to 0.

NOTE

Hosts Need a Subnet Mask Each host on a TCP/IP network requires a subnet mask even if it is on a single-segment network. Although the subnet mask is expressed in dotted decimal notation, a subnet mask is not an IP address.

The subnet mask is broken down into four 8-bit octets in the same fashion as the class addresses.

Table 6.3 lists the default subnet masks using dotted decimal notation.

TABLE 6.3

DEFAULT SUBNET MASKS

| <i>Address Class</i> | <i>Bits for Subnet Mask</i> | <i>Subnet Mask</i> |
|----------------------|--|--------------------|
| Class A | 11111111 00000000 00000000 00000000 | 255.0.0.0 |
| Class B | 11111111 11111111 00000000 00000000 | 255.255.0.0 |
| Class C | 11111111 11111111 11111111 00000000 | 255.255.255.0 |

Default Gateway (Router)

This optional setting is the IP address of the router for this subnet segment. Each subnet segment is bounded by a router that will direct packets destined for segments outside the local one to the correct segment or to another router that can complete the connection. Routers, therefore, have connections to more than one network segment, and this address points to the router's network adapter on the same segment as your computer. If this address is left blank, this computer will only be able to communicate with other computers on the same network segment.

Figure 6.11 shows a hypothetical network that is using a subnet mask of 255.255.224.0. This provides for a maximum of six subnets

of 8,190 hosts each; however, in this example only three connections are being used. The hosts on the subnet 192.168.32.0 would each have the default gateway address set to the address of the router port connecting that subnet, 192.168.32.1. The IP address of each router connection is local to the subnet it serves, allowing the hosts on that subnet to communicate with it directly.

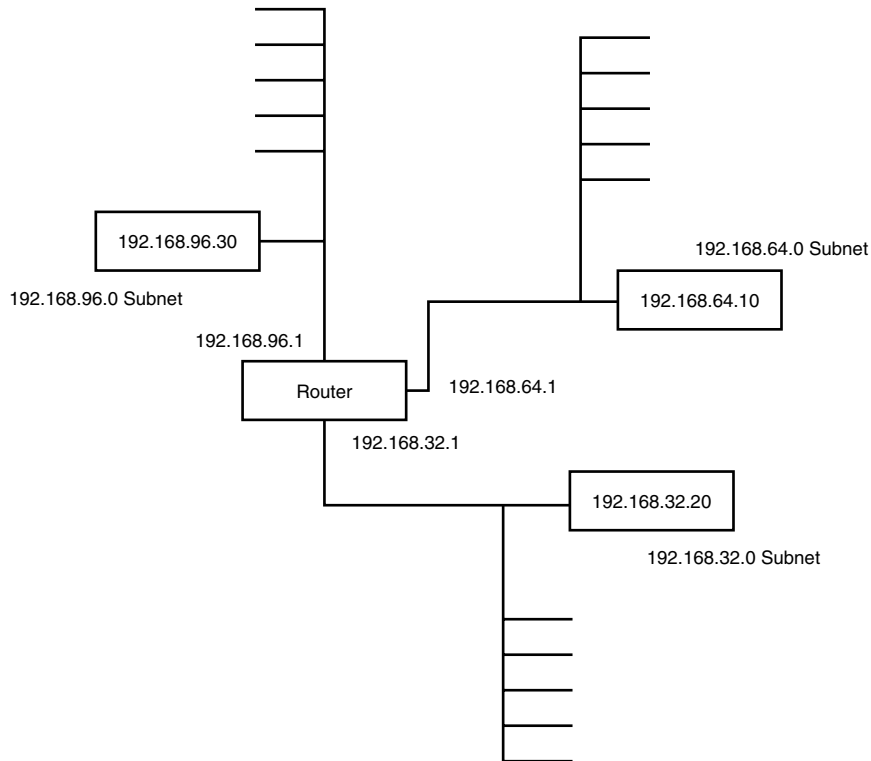


FIGURE 6.11

Default gateways on a subnetworked network.

Windows Internet Name Service (WINS)

Computers may use IP addresses to identify one another, but users generally prefer to use computer names. Windows 2000 Professional allows Windows 9x and Windows NT 4 clients to use NetBIOS names to communicate and therefore requires a means to resolve

NOTE

The Dynamic Nature of WINS WINS eliminates the need for an LMHOSTS file, which is a static alternative to WINS. Maintaining an LMHOSTS file requires much more administrative overhead than using WINS.

NOTE

Name Resolution Name resolution is the process of translating fully qualified domain names (FQDN) to IP addresses.

WARNING

TCP/IP Setting If the settings for the TCP/IP protocol are incorrectly specified, you will experience problems that may keep your computer from establishing communications with other TCP/IP hosts in your network. In extreme cases, communications on your entire subnet can be disrupted.

NetBIOS names to IP addresses. The Windows Internet Name Service is an enhanced NetBIOS name server that registers NetBIOS computer names and resolves them to IP addresses. WINS provides a dynamic database that maintains mappings of computer names to IP addresses.

Domain Name Systems (DNS) Server Address

DNS is an industry-standard distributed database that provides name resolution and a hierarchical naming system for identifying TCP/IP hosts on Internets and private networks. A DNS address must be specified to enable connectivity with the Internet or with UNIX TCP/IP hosts. You can specify more than one DNS address and the search order in which they should be used.

The IPCONFIG command can be used to display information recently obtained from the DNS service.

The IPCONFIG /DISPLAYDNS command displays the contents of the DNS client resolver cache, which includes entries preloaded from the local HOSTS file, as well as any recently obtained resource records for name queries recently resolved by the system. This is used by the DNS Client service to quickly resolve frequently queried names.

The resolver cache can also support negative caching of unresolved or invalid DNS names. These entries are added by the DNS Client service when it receives a negative answer from a DNS server for a queried name. The negative result is then cached for a brief period of time so that this name is not queried repeatedly by the system.

The IPCONFIG /FLUSHDNS command will flush and reset the DNS resolver cache. Once this option is used, the computer must query DNS servers again for any names previously used on the computer.

Windows 2000 Server contains a dynamically updated DNS service. This service is updated with records obtained from either a DHCP server or the DHCP client service on a Windows 2000 Professional workstation. Normally, this is done when the DHCP address is

assigned to the Windows 2000 Professional workstation. The `IPCONFIG /REGISTERDNS` command provides a mechanism to manually initiate dynamic registration for the DNS names and IP addresses configured at a computer. This option would normally only be used in troubleshooting DNS name resolution problems.

By default, the `IPCONFIG /REGISTERDNS` command refreshes all DHCP address leases and registers all related DNS names configured and used by the client computer.

You can specify all the settings for the TCP/IP protocol manually, or they can be automatically configured through a network service called *Dynamic Host Configuration Protocol* (DHCP).

Understanding DHCP

One way to avoid the possible problems of administrative overhead and incorrect settings for the TCP/IP protocol (which are usually caused by manual configurations) is to set up your network so that all your clients receive their TCP/IP configuration information automatically through DHCP. DHCP centralizes and manages the allocation of the TCP/IP settings required for proper network functionality for computers that have been configured as DHCP clients. One major advantage of DHCP is that most of the configuration of your network settings need happen only once, at the DHCP server. Also, the TCP/IP settings that the DHCP client receives from the *DHCP server* are only leased, and must be periodically renewed. This lease and renewal sequence enables a network administrator to change client TCP/IP settings, if needed.

Using DHCP

To configure a computer as a DHCP client, all you must do is to specify an IP address automatically in the TCP/IP properties box (see Figure 6.12). Exercise 6.2 at the end of the chapter contains complete instructions.

Testing DHCP

To determine the network settings that a DHCP server has leased to your computer, type the following command at a command prompt:

```
IPCONFIG /all
```

NOTE

Dynamic DNS Only The `registerdns` option on `IPCONFIG` will function correctly only if the DNS allows the client to register directly (it is configurable by the Network Administrator) or if the DNS can accept configuration records. This is only true with the Microsoft DNS service and recent version of BIND.

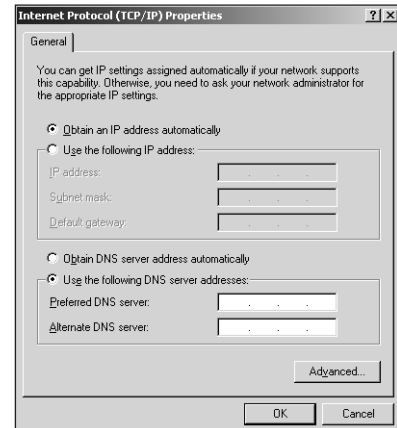


FIGURE 6.12
Specifying that TCP/IP configuration comes from a DHCP server.

The following is a sample output from the IPCONFIG command.

```
Windows 2000 IP Configuration
Host Name . . . . . : NTW1
Primary DNS Suffix . . . . . : Private.Barknet
Node Type . . . . . : Mixed
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
DNS Suffix Search List. . . . . : Private.Barknet
Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix . :
Description . . . . . : NE2000

➤Compatible
Physical Address. . . . . : 00-40-05-3E-C7-

➤BF
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IP Address. . . . . : 192.168.0.113
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.0.1
DHCP Server . . . . . : 192.168.0.1
DNS Servers . . . . . : 192.168.0.1
Lease Obtained. . . . . : June 4, 1999
➤11:40:48 PM
Lease Expires . . . . . : June 12, 1999
➤11:40:48 PM
```

Note that IPCONFIG also gives you full details on the duration of your current lease. You can verify whether a DHCP client has connectivity to a DHCP server by releasing the IP address and requesting a new lease. You can conduct this test by typing the following commands in a command window:

```
IPCONFIG /release
IPCONFIG /renew
```

Manually Configuring TCP/IP

You can manually configure your TCP/IP settings by entering the required values into the TCP/IP properties sheet (see Figure 6.13). For complete details, see Exercise 6.1 at the end of the chapter.

Name Resolution with TCP/IP

DNS and WINS are not the only name resolution methods available for Windows 2000 TCP/IP hosts. Microsoft also provides two different lookup files: LMHOSTS and HOSTS. You can find samples of these files in the `\winnt_root\SYSTEM32\DRIVERS\ETC` folder. Read the contents of each sample file for instructions on how to use them.

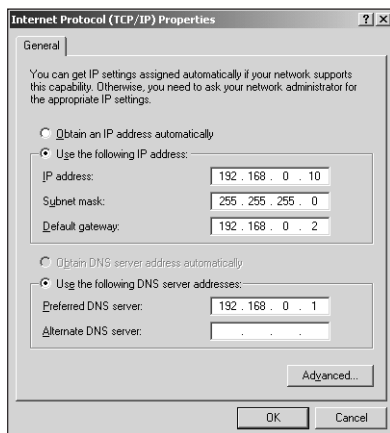


FIGURE 6.13
Manual configuration of a TCP/IP host.

NOTE

LMHOSTS File Although the sample LMHOSTS file in the `\winnt_root\SYSTEM32\DRIVERS\ETC` folder is named LMHOSTS.SAM, it must be renamed LMHOSTS with no file extension. Otherwise, it will not be used for name resolution.

Advanced TCP/IP Configuration

As the complexity of networks grows, the requirement for more sophisticated access and control of information flowing over the networks grows as well. Two recent additions to the TCP/IP configuration options in Windows 2000 are Virtual Private Networks (VPN) and IP Security (IPSec). These automatic methods encrypt the information flowing between two computer systems even if it is using the public Internet network.

Virtual Private Networks (VPN)

A *Virtual Private Network (VPN)* allows the computers in one network to connect to the computers in another network by the use of a tunnel through the Internet or other public network. The VPN provides the same security and features formerly available only in private networks.

A VPN connection allows you to connect to a server on your corporate network from home or when traveling using the routing facilities of the Internet. The connection appears to be a private point-to-point network connection between your computer and the corporate server.

Additionally, VPNs can be used to connect remote office LANs to the corporate LAN or to other remote LANs to share resources and information using direct connect or dial-up access.

The basic functions managed by VPNs are the following:

- ◆ **User authentication.** Verifies the user's identity and restricts VPN access to authorized users only.
- ◆ **Address management.** Assigns the client's address on the private net and ensures that private addresses are kept private.
- ◆ **Data encryption.** Data carried on the public network must be unreadable to unauthorized clients on the network.
- ◆ **Key management.** Encryption keys must be refreshed for both the client and the server.
- ◆ **Multiprotocol support.** The most common protocols used in the public network are supported.

A VPN is not a protocol in itself, but rather the encapsulation of existing protocols and the encryption of the data being transmitted.

Windows 2000 Professional provides two encapsulation methods for creating VPN connections.

Point-to-Point Tunneling Protocol (PPTP)

This protocol enables the secure transfer of data from your computer to a remote computer on TCP/IP networks. PPTP tunnels, or encapsulates, IP, IPX, or NetBEUI protocols inside PPP datagrams. PPTP can work over dedicated Internet connections or over dial-up connections; however, it does require IP connectivity between your computer and the server to which it is authenticating before the tunnel can be established.

In Point-to-Point Tunneling, a PPP frame (containing an IP datagram or an IPX datagram) is wrapped with a Generic Routing Encapsulation (GRE) header and an IP header. In the IP header are the source and destination IP address that correspond to the VPN client and VPN server.

Figure 6.14 shows the PPTP encapsulation of a PPP payload.

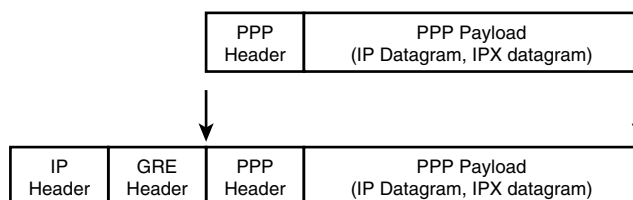


FIGURE 6.14

PPTP encapsulation of an encrypted datagram.

PPTP Encryption

The PPP frame is encrypted with Microsoft Point-to-Point Encryption (MPPE) by using encryption keys generated from the MS-CHAP or EAP-TLS authentication process. Virtual private networking clients must use either the MS-CHAP or EAP-TLS authentication protocol in order to encrypt PPP payloads. PPTP does not provide encryption services. PPTP encapsulates a previously encrypted PPP frame.

Layer 2 Tunneling Protocol (L2TP)

L2TP is an Internet tunneling protocol with roughly the same functionality as PPTP. The Windows 2000 implementation of L2TP is designed to run natively over IP networks.

Encapsulation for L2TP consists of two separate layers:

- ◆ A PPP frame (containing an IP datagram or an IPX datagram) is wrapped with an L2TP header and a UDP header.
- ◆ The resulting L2TP message is then wrapped with an IPsec Encapsulating Security Payload (ESP) header and trailer, an IPsec Authentication trailer that provides message integrity and authentication, and a final IP header. In the IP header are the source and destination IP addresses that correspond to the VPN client and VPN server.

Figure 6.15 shows the L2TP encapsulation of a PPP payload.

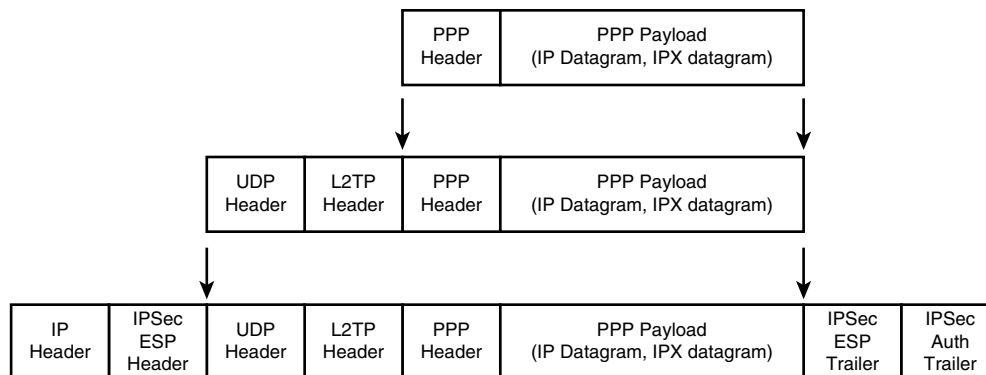


FIGURE 6.15

L2TP encapsulation of an encrypted PPP payload.

Encryption

The L2TP message is encrypted with IPsec encryption mechanisms by using encryption keys generated from the IPsec authentication process. The portion of the packet from the UDP header to the IPsec ESP Trailer inclusive is encrypted by IPsec.

NOTE

Non-Encrypted L2TP packets It is possible to have a non-IPSec-based (non-encrypted) L2TP connection, where the PPP payload is sent in plain text. However, a non-encrypted L2TP connection is not recommended for virtual private network connections over the Internet because communications of this type are not secure.

Installing VPN Connections

A VPN connection is configured running the Network Connection Wizard (Make New Connection) in the Network and Dial-Up Connection applet in the Control Panel. A VPN is created by selecting the option Connects to a Private Network Through the Internet. The connection can be configured to dial to an Internet Service Provider (ISP) or to connect directly to a VPN server if your computer is directly connected to the Internet.

The type of connection (PPTP or L2TP) is defined by the server you are connecting to. In any case, the connection and security are negotiated automatically.

Configuring IPSec

IPSec is applied according to a global policy. Select Network and Dial-Up Connections from the Control Panel and right-click a connection to find the properties page. Select the TCP/IP protocol and click the Advanced button. Click the Options tab and then the Properties button. This will bring you to the screen that allows you to select an IPSec policy. These are system-wide and defined by the system administrator.

CONNECTING TO COMPUTERS BY USING DIAL-UP NETWORKING

Connect to computers by using dial-up networking.

Dial-Up Networking enables you to extend your network to unlimited locations. A dial-up connection connects your computer to a private network or the Internet (through an Internet Service Provider), to a private network through the public Internet using a secure Virtual Private Network (VPN) connection, or to a RAS server using the Microsoft RAS Protocol. The Microsoft RAS protocol is a proprietary protocol that supports the NetBIOS standard.

The dial-up connection can be made by a modem over the public switched network (also known as the Plain Old Telephone System, or POTS), or through a cable modem, xDSL service, X.25