# Windows NT® Server 4 FOR DUMMIES®

**by Ed Tittel with Mary Madden & James Michael Stewart**

Hungry Minds™

HUNGRY MINDS, INC.

New York, NY ◆ Indianapolis, IN ◆ Cleveland, OH

**Windows NT® Server 4 For Dummies®**

Published by
**Hungry Minds, Inc.**
909 Third Avenue
New York, NY 10022
www.hungryminds.com
www.dummies.com (Dummies Press Web site)



Library of Congress Catalog Card No.: 99-60727

ISBN: 0-7645-0524-6

Printed in the United States of America

10 9 8 7 6 5

1B/QV/QU/QR/IN

Distributed in the United States by Hungry Minds, Inc.

Distributed by CDG Books Canada Inc. for Canada; by Transworld Publishers Limited in the United Kingdom; by IDG Norge Books for Norway; by IDG Sweden Books for Sweden; by IDG Books Australia Publishing Corporation Pty. Ltd. for Australia and New Zealand; by TransQuest Publishers Pte Ltd. for Singapore, Malaysia, Thailand, Indonesia, and Hong Kong; by Gotop Information Inc. for Taiwan; by ICG Muse, Inc. for Japan; by Intersoft for South Africa; by Eyrolles for France; by International Thomson Publishing for Germany, Austria and Switzerland; by Distribuidora Cuspide for Argentina; by LR International for Brazil; by Galileo Libros for Chile; by Ediciones ZETA S.C.R. Ltda. for Peru; by WS Computer Publishing Corporation, Inc., for the Philippines; by Contemporanea de Ediciones for Venezuela; by Express Computer Distributors for the Caribbean and West Indies; by Micronesia Media Distributor, Inc. for Micronesia; by Chips Computadoras S.A. de C.V. for Mexico; by Editorial Norma de Panama S.A. for Panama; by American Bookshops for Finland.

For general information on Hungry Minds' products and services please contact our Customer Care Department within the U.S. at 800-762-2974, outside the U.S. at 317-572-3993 or fax 317-572-4002.

For sales inquiries and reseller information, including discounts, premium and bulk quantity sales, and foreign-language translations, please contact our Customer Care Department at 800-434-3422, fax 317-572-4002, or write to Hungry Minds, Inc., Attn: Customer Care Department, 10475 Crosspoint Boulevard, Indianapolis, IN 46256.

For information on licensing foreign or domestic rights, please contact our Sub-Rights Customer Care Department at 212-884-5000.

For authorization to photocopy items for corporate, personal, or educational use, please contact Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, or fax 978-750-4470.

For information on using Hungry Minds' products and services in the classroom or for ordering examination copies, please contact our Educational Sales Department at 800-434-2086 or fax 317-572-4005.

Please contact our Public Relations Department at 212-884-5163 for press review copies or 212-884-5000 for author interviews and other publicity information or fax 212-884-5400.



# Chapter 12

# Next on Montel — IP Addresses and the Nerds Who Love Them

*In This Chapter*

- Working with TCP/IP and NetBIOS names
- Understanding IP addressing, nets, and subnets
- Obtaining Internet-ready IP addresses
- Using private IP addresses
- Using proxy servers and address translation
- Working with DHCP
- Knowing when to use WINS
- Working with DNS

TCP/IP drives the Internet and makes it accessible around the world. TCP/IP, however, is a lot more than just a collection of protocols: Many elements in the TCP/IP marry protocols to related services to provide more complete capabilities. Important examples include dynamic address allocation and management, known as DHCP, plus domain name to address resolution services, known as DNS. You find out about TCP/IP names, addresses, and related standard services in this chapter, as well as some other services that are unique to Windows NT.

## Name-Calling with TCP/IP and NetBIOS

Whenever you issue a command in Windows NT, you're expected to use the proper syntax. Otherwise, your efforts might not produce the desired results. For example, when you issue a NET USE command from a command prompt, you must enter the server name and a share name, as well as the drive you wish to map. Thus, a simple command like `NET USE G: \\LANWRIGHTS\APPS` associates the drive letter G: with a share named APPS

on the LANWRIGHTS server. If you're using the TCP/IP protocol to convey the data involved, the protocol doesn't know how to interpret the name LANWRIGHTS as the server. Instead, it understands IP addresses, such as 172.16.1.7.

If you use TCP/IP on your network, you need some way to convert IP addresses into names, and vice versa. Just as the United Nations requires translators so everyone can communicate, so does Windows NT! That's why understanding naming conventions and name-to-address resolution is such an important part of working with TCP/IP on Windows NT.

## NetBIOS names

If you're like most folks, you freeze like a deer in the headlights when you hear the word NetBIOS. Don't worry. Only a small number of people really understand NetBIOS in detail, but figuring out what you need to know without stressing out is easy.

A NetBIOS name is often called a computer name. When you install Windows NT onto a network, each computer that runs Windows NT requires a unique computer name. This permits all NetBIOS-based utilities to identify each machine by its name. Any time you enter a command that includes a computer name, Windows NT knows which computer you're talking about.

If you try to give two devices the same name, you run into trouble — like trying to use the same Social Security number for two people. Each time a computer joins the network, it registers its name with a browser service that keeps track of such things. When the second computer with the same name tries to register, it is rejected because that name is already "taken." In fact, that machine will be unable to join the network until its name is changed to something unique.

When creating NetBIOS names, you need to work within their limitations:

- NetBIOS names must be between 1 and 15 characters long. (If you have DOS or Windows 3.x machines on your network, they can't recognize NetBIOS names with more than 8 characters.)
- NetBIOS names may not contain any of the following characters: " (double quotation mark), / (right slash), \ (left slash), [ (left square bracket), ] (right square bracket), : (colon), ; (semicolon), | (vertical slash), = (equal sign), + (plus sign), * (asterisk), ? (question mark), < (left angle bracket), and > (right angle bracket). Dollar signs are not recommended because they have a special meaning. (A NetBIOS name that ends in $ does not display in a browse list.)

- Don't use lengthy names, or put spaces in names. Windows NT doesn't care if you use longer names or include embedded spaces, but other networking clients or systems may not be able to handle such usages.
- Pick names that make sense to users that are short and to the point. Don't name machines after their users or locations, especially if users come and go regularly, or if machines move around a lot. When it comes to servers, name them to indicate organizational role or affiliation (for example, Sales).

What's in a NetBIOS name, you ask? A short, clear indication of what's being named so that users can recognize what they see. At best, this kind of naming convention will make sense without requiring further explanation. At the least, you can do what we do and put a sticker with the machine's name on each monitor for self-identification purposes. Figure 12-1 shows a list of NetBIOS names in our network's Network Neighborhood (names that begin with Nts indicate Windows NT Servers, Ntw indicates Windows NT Workstations, and Win98 . . . well, you see what we mean; we also add numbers to identify each machine's IP address).

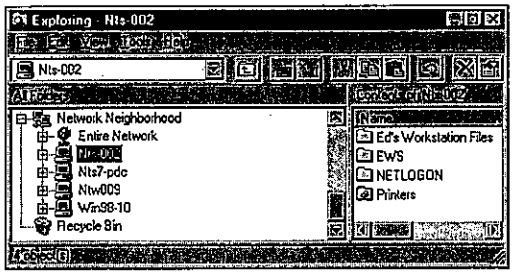



**Figure 12-1:** NetBIOS computer names show up for machines under the NT Explorer Network Neighborhood.

## TCP/IP names and addresses

TCP/IP uses a different scheme for names than does NetBIOS. TCP/IP uses 32-bit numbers to construct IP addresses (for example, 172.16.1.11). Each host or node on a TCP/IP network must have a unique IP address.

IP addresses are not meaningful to most humans and are difficult to remember. Thus, it's helpful to have some way to convert IP addresses into meaningful names. On an Windows NT network, you use computer names (also known as NetBIOS names). The Internet community uses a different naming convention called domain names. Translation methods, such as WINS and DNS maintain databases for converting an IP address to a computer name (WINS) or a domain name (DNS).

If you've ever used a Web browser on the Internet, you know that you can type a URL (Uniform Resource Locator) such as `http://206.224.65.194/` or `http://www.lanw.com/` to obtain access to a Web page. You can do so because the Internet uses the Domain Name Service, also known as DNS, to resolve IP addresses to domain names and vice versa. If you type the IP address, the Web browser jumps straight to the named address; if you type a domain name, your request goes through a DNS server that resolves the name to an IP address, and then the browser jumps to the named address thereafter.

In the IP world, the naming scheme you can use is limited if you plan to connect your network directly to the Internet. An organization known as the Internet Network Information Center, or InterNIC, is in charge of approving and maintaining the database of "legal" Internet domain names. You can request any domain name you want, but if someone else is using it or has a legitimate claim to a trade or brand name, you won't be able to use it. For example, you probably won't be able to use `mcdonalds.com` or `cocacola.com` as domain names; likewise, if somebody else has already registered `xyzcorp.com`, you wouldn't be able to use that name, even if your company is named XYZ Corporation.

The format for a typical IP name is host.domainname.suffix. The domain name is something you can't guarantee, but typically represents your organization. The suffix sometimes identifies the country of origin (for example, .ca is Canada, .de is Germany) or the type of organization (.gov is government, .edu is education, .com is a commercial business, .org is a nonprofit organization, and so forth).

Some domain names are more complex; they can take a form like host.subdomain.domainname.suffix, as in `jello.eng.sun.com`, where the host name is `jello`, the subdomain is `eng` (for engineering), and the domain name is `sun` (the domain name for Sun Microsystems, Inc.) which is a commercial (.com) entity. The only part of the name that's under control of the InterNIC is the domain name part, and the suffix, but every domain name must be unique in its entirety to be recognized properly.

Names that include the host part and the domain name and suffix (plus any other subdomain information that may apply) are called Fully Qualified Domain Names or FQDNs. To be valid, any FQDN must have a corresponding

entry in some DNS server's database that permits it to be translated into a unique numeric TCP/IP address. For example, your authors' Web server is named `http://www.lanw.com`, which resolves into a numeric address of 206.224.65.194.

As long as you're completely isolated from the Internet and intend to stay that way, you can assign any names and IP addresses you might like on your network. But if you ever connect your network to the Internet, you'll have to go back and change everything! If your network will be — or simply *might be* — connecting to the Internet, you have one of two options for assigning addresses:

1. **You can obtain and install valid public IP addresses and domain names now.**

   You can obtain these directly from the InterNIC at some difficulty and expense, or you can pay your Internet Service Provider (ISP) to do this for you. We recommend the latter course. When you obtain a range of IP addresses for your network — remember, each computer needs its own unique address, and some computers or devices need multiple addresses (one for each interface) — make sure you get enough to leave some room to grow.

2. **You can (and should) obtain a valid domain name from the InterNIC, but you can use any of a range of reserved IP addresses called private IP addresses to number your networks.**

   These addresses may not be used directly on the Internet, but have been set aside for private use. When used in concert with a type of software called Network Address Translation (or NAT for short), this approach requires you to obtain only a small number of public IP addresses but still permits Internet access for every computer on your network. This topic is discussed in more detail later in this chapter in the section "The magic of proxy servers and address translation."

To find out more about the process of obtaining a domain name, visit the InterNIC's Web site at `http://internic.net` and click the hyperlink that reads "domain name registration services." You'll find details on name registration services and well as the directory and database services that support the Internet's distributed collection of DNS servers.


If you're thinking about registering a domain name, check the existing name database at the InterNIC Web site to make sure that name's not already assigned to somebody else. Why ask for something you can't have?

# An Address for Every Node

A unique numeric identification tag, called an *IP address,* is assigned to each interface on a TCP/IP network. Every IP address within a TCP/IP network must be unique. Each device on a TCP/IP network is known as a *host.* Each host has at least one network interface with an assigned IP address. However, a host can have multiple network interface cards (NICs), and even multiple IP addresses assigned to each NIC.

## Of network and host IDs

An IP address consists of two components: a *network ID* and a *host ID.* The network ID identifies the network segment to which the host belongs. The host ID identifies an individual host on some specific network segment. A host can only communicate directly with other hosts on the same network segment. A network segment is a logical division of a network into unique numeric network Ids called *subnets.* A host must use a router to communicate with hosts on other subnets.

A *router* moves packets from one subnet to another. A router reads the network ID for a packet's destination address and determines if that packet should remain on the current subnet or be routed to a different subnet. When a router delivers a packet to the correct subnet, the router then uses the host ID portion of the destination address to deliver the packet to its final destination.

A typical IP address looks like 207.46.131.137 (which matches the domain name `http://www.microsoft.com`). This numeric IP address format is known as *dotted-decimal notation.* But computers "see" IP addresses as binary numbers. This same IP address is 11001111 00101110 10000011 10001001 in binary form and written in collections of eight bits called *octets.* Each octet is converted to a decimal number and then separated by periods to form the dotted-decimal notation format shown at the beginning of this paragraph. The decimal version of IP addresses is more human friendly than binary. As you may already know, domain names and NetBIOS names are still more friendly because they use symbolic names that make sense to humans.

An IP address requires 32 binary digits and defines a 32-bit address space that supports nearly 4.3 billion unique addresses. Although this seems like a lot of addresses, the number of available IP addresses is dwindling. Consequently, several plans exist to expand or change the IP addressing scheme to open up many more addresses. For more information on such plans please visit the Web site at: `http://www.6bone.net/ngtrans.html`.

IP designers carved the entire galaxy of IP addresses into classes, to meet different addressing needs. Today, there are five IP address classes labeled by the letters A through E. Classes A, B, and C are assigned to organizations to allow their networks to connect to the Internet, and Classes D and E are reserved for special uses.

The first three classes of addresses differ by how their network ID is defined:

- Class A addresses use the first octet for the network ID.
- Class B uses the first two octets.
- Class C uses the first three.

Class A addresses support a relatively small number of networks, each with a huge number of possible hosts. Class C addresses support a large number of networks, each with a relatively small number of hosts as shown in Table 12-1 (Class B falls in the middle). Thus, branches of the military, government agencies, and large corporations are likely to need Class A addresses, medium-sized organizations and companies Class B addresses, and small companies and organizations Class C addresses.

When it comes to recognizing address classes A through C, the network ID for Class A addresses always starts its first octet with a zero. Each Class B network ID always starts with 10, while Class C network IDs always start with 110. Consequently, you can determine address class by examining an address, either in binary or decimal form. (See Tables 12-1 and 12-2.)

**Table 12-1 Address Classes and Corresponding Network and Host IDs**

| *Class* | *High-Order Bits* | *First Octet Range* | *#Networks* | *#Hosts* |
|---|---|---|---|---|
| Class A | 0xxxxxxx | 1-126.x.y.z | 126 | 16,777,214 |
| Class B | 10xxxxxx | 128-191.x.y.z | 16,384 | 65,534 |
| Class C | 110xxxxx | 192-223.x.y.z | 2,097,152 | 254 |

**Table 12-2 Division of IP Address Component Octets According to Class**

| *Class* | *IP Address* | *Network ID* | *Host ID* |
|---|---|---|---|
| A | 10.1.1.10 | 10 | 1.1.10 |
| B | 172.16.1.10 | 172.16 | 1.10 |
| C | 192.168.1.10 | 192.168.1 | 10 |


*Note:* Network ID 127 is missing from Table 12-1. That's because 127 is a loopback address (when testing IP transmission, it transmits to itself).

No valid IP address may include an octet that consists entirely of ones or zeros (0 or 255 in decimal), because these addresses are reserved for broadcast addresses (255) and subnet identification (0).

## Subnetting IP addresses

Subnets represent divisions of a single TCP/IP network address into logical subsets. The motivation for subnetting is twofold. First, subnetting reduces the amount of overall traffic on any network segment by collecting systems that communicate often into groups. Second, subnetting makes is easier for networks to grow and expand, and adds an extra layer of security controls. Subnets work by "stealing" bits from the host part of an IP address and using those bits to subdivide a single IP network address into two or more subnets.

Subnet masks are typically used to divide IP address blocks into smaller subnetworks. The base subnet masks for Class A, B, and C networks are 255.0.0.0, 255.255.0.0, and 255.255.255.0 respectively. By adding extra bits set to 1 in the space occupied by the 0 that appears next to the rightmost 255 in any such number, additional subnet masks may be created. This transformation is illustrated in Table 12-3, which shows the some typical values for usable subnet masks.


### Routers move packets among subnets and networks

Only *routers* can transfer packets from one subnet to another, or from one network ID to another, in the TCP/IP world. Routers may be specialized devices that include high-end, high-speed devices from companies like Cisco Systems or Bay Networks. But any computer with two or more NICs installed, where each NIC resides on a different subnet, can be a router, provided that the computer can forward packets from one NIC to another (and thus, from one subnet to another). Right out of the box, in fact, Windows NT Server includes the software and built-in capabilities to function in this way. Computer nerds like to call such machines *multi-homed computers* because the machines are "at home" on two or more subnets.

**Table 12-3 Subnet Masks and Results**

| *Binary Mask* | *Decimal Equivalent* | *Number of New Subnets* | *Number of Hosts* |
|---|---|---|---|
| 00000000 | A: 255.0.0.0<br>B: 255.255.0.0<br>C: 255.255.255.0 | A: 16,777,214<br>B: 65,534<br>C: 254 | 1 |
| 10000000 | A: 255.128.0.0<br>B: 255.255.128.0<br>C: 255.255.255.128 | A: Not valid<br>B: Not valid<br>C: Not valid | Not valid |
| 11000000 | A: 255.192.0.0<br>B: 255.255.192.0<br>C: 255.255.255.192 | A: 4,194,302<br>B: 16,382<br>C: 62 | 2 |
| 11100000 | A: 255.224.0.0<br>B: 255.255.224.0<br>C: 255.255.255.224 | A: 2,097,150<br>B: 8,190<br>C: 30 | 6 |
| 11110000 | A: 255.240.0.0<br>B: 255.255.240.0<br>C: 255.255.255.240 | A: 1,048,574<br>B: 4,094<br>C: 14 | 14 |
| 11111000 | A: 255.248.0.0<br>B: 255.255.248.0<br>C: 255.255.255.248 | A: 524,286<br>B: 2,046<br>C: 6 | 30 |
| 11111100 | A: 255.252.0.0<br>B: 255.255.252.0<br>C: 255.255.255.252 | A: 262,142<br>B: 1022<br>C: 2 | 62 |
| 11111110 | A: 255.254.0.0<br>B: 255.255.254.0<br>C: 255.255.255.254 | A: 131,070<br>B: 510<br>C: Not valid | 126 |


Because routers are required to communicate across IP subnets, some router's IP address on each subnet must be known to every client on that subnet. This address is called the *default gateway*, because it is where all out-of-subnet transmissions are directed by default (it's the gateway to the world outside each local subnet, in other words). If no default gateway is defined, clients can't communicate outside their subnet.

## Going public: Obtaining Internet-ready IP addresses

Deploying your own network or using a stand-alone system with NAT to connect to the Internet requires that you obtain one or more valid IP

addresses. For some uses, you may simply contract with an ISP to use a dial-up connection. Each time you connect you'll be assigned an IP address automatically from a pool of available addresses. Once you disconnect from the ISP, that IP address will return to the pool for re-use. This works equally well for stand-alone machines and for the servers that might dial into an ISP to provide an on-demand connection for users who have private IP addresses but can attach to the Internet using NAT software.

One way to attach an entire network to the Internet, is to lease a block or subnet of IP addresses from an ISP. Leasing IP addresses can be expensive and can limit your growth. Also, many ISPs can no longer lease large blocks of IP addresses so you may have to limit Internet access to specific machines or subnets.

For more information about taking this approach, you need to contact your ISP to find out what it can offer by way of available addresses and contiguous subnets. For some uses, public IP addresses are required because security needs dictate a true "end-to-end" connection between clients and servers across the Internet. In plain English, a true end-to-end connection means that the IP address that a client advertises to the Internet is the same one it uses in reality. In the section "The magic of proxy servers and address translation," you discover an alternate approach where the IP address advertised to the Internet is different from the private IP address that a client uses on its home subnet.


For some applications, particularly where secure IP-based protocols like IPSec (IP Secure) or particular Secure Sockets Layer (SSL) implementations are required, network address translation techniques may not work! Make sure you understand your application requirements in detail before you decide whether to lease public IP addresses or use private IP addresses with network address translation.

## The magic of proxy servers and address translation

If you don't want to pay to lease a range of IP addresses, and your application requirements permit you to use private IP addresses, you can employ the IP addresses reserved for private use in RFC 1918 on your networks. When used in combination with network address translation software to connect to an ISP, a single public IP address (or one for each Internet connection) is all you need to service an entire network.

RFC 1918 (`http://www.faqs.org/rfcs/rfc1918.html`) defines special IP addresses for use on private intranets. These addresses, which appear in Table 12-4, will not be routed on the Internet by design. This approach actually provides improved security for your network as a fringe benefit,

because it means that any impostor who wants to break into your network cannot easily masquerade as a local workstation. (Doing so would require routing a private IP address packet across the Internet.) Because all of these addresses are up for grabs, you can use whatever address class makes sense for your organization (and for Class B and Class C addresses, you can use as many as you need within the legal range of such addresses).

**Table 12-4 Private IP Address Ranges from RFC 1918**

| *Class* | *Address Range* | *# Networks* |
|---|---|---|
| A | 10.0.0.0 - 10.255.255.255 | 1 |
| B | 172.16.0.0 - 172.31.255.255 | 16 |
| C | 192.168.0.0 - 192.168.255.255 | 254 |

Thus, using address translation software to offer Internet access reduces your costs and allows nearly unlimited growth. If you think private IP addresses combined with NAT software makes sense for your situation, consult with your ISP for specific details and recommendations on how to use this technology on your network.

You've probably heard the terms *firewall* and *proxy* thrown about often when reading or talking about Internet access. Firewalls and proxy servers are networking tools that are little more than special-purpose routers. A firewall may be used to filter traffic, both inbound or outbound.

Firewall filters can be based on source or destination address, a specific protocol, or port address, or even on patterns that appear in the content or a data packet. A proxy server is an enhanced firewall, and its primary purpose is to manage communications between an in-house network and external networks such as the Internet. Proxies hide the identity of internal clients and can keep local copies of resources that are accessed frequently (this is called *caching*, and improves response time for users).

You can check out several great online resources for firewalls, but online information on proxies is limited to product documentation. In addition to consulting the Windows NT Server Resource Kit and TechNet, here are several online resources you might want to check to discover more about these technologies:

- Zeuros Firewall Resource: `www.zeuros.co.uk/`
- Firewall Overview: `www.access.digex.net/~bdboyle/firewall.vendor.html`
- Great Circle Associates: `www.greatcircle.com/`
- 4 Firewalls: `www.4firewalls.com/`
- Microsoft's Proxy Server 2.0: `www.microsoft.com/proxy/`

- Aventail VPN: `www.aventail.com/`
- Netscape's Proxy Server: `www.netscape.com/`
- Ositis Software's WinProxy: `www.ositis.com/`
- Deerfield Communication's WinGate Pro: `www.deerfield.com/`

For example, your authors use Ositis Software's WinProxy product, which acts as a proxy and provides NAT services, to link their networks to an ISP across an ISDN connection. We allow the ISP to assign us an IP address each time we log onto their host for an Internet connection. This doesn't matter because the NAT services translate between whatever address they assign us and the internal addresses each machine uses on the other side of the WinProxy software. We only pay for the temporary use of a single IP address, but we can handle up to eight connections to the Internet at a time!

# Go Figure: Configuring IP Addresses for Windows NT Server

Configuring TCP/IP on Windows NT Server can range from simple to complex. We review the simple process and discuss a few advanced items, but for complex configurations, you should consult a reference such as the Windows NT Server Resource Kit or TechNet.

Three basic items are always required for configuring TCP/IP:

- IP address
- Subnet mask
- Default gateway

With just these three items, you can connect a client or server to a network. The protocol is configured on the Protocol tab of the Network applet. If the protocol isn't installed already, click the Add button to display a list of installable protocols. If it's already installed, select TCP/IP in the list and click Properties.

The TCP/IP Properties dialog box has five tabs. The first tab, Microsoft TCP/IP Properties dialog box (see Figure 12-2), is where the three IP configuration basics are defined. Notice there's a selection to obtain an IP address from a DHCP server. Because most servers don't work well using dynamic IP addresses, you should define a static IP address for your Windows NT Server instead of using DHCP. You will either obtain a public IP address from your ISP, or use a private IP address from one of the reserved address ranges defined in RFC 1918.

Likewise, you must calculate a subnet mask for your network. Here again, you may obtain this from your ISP if you're using public IP addresses, or calculate your own if you're using private IP addresses. In most cases where private IP addresses are used, the default subnet mask for the address class should work without alteration or additional calculations.

Finally, you must also provide a default gateway address for your server. The default gateway should be the address of the router on the local subnet to which the server is attached that can forward outbound traffic to other network segments. On networks using public IP addresses, this will probably be a router, firewall, or proxy server that connects the local subnet to other subnets or the Internet. On networks using private IP addresses, this will usually be the machine where the proxy and NAT software resides, that mediates between the local subnet and an Internet connection.

Once you define an IP address, a subnet mask, and a default gateway, click OK, then close the Network applet, and reboot. That's all there is to basic TCP/IP configuration on Windows NT!

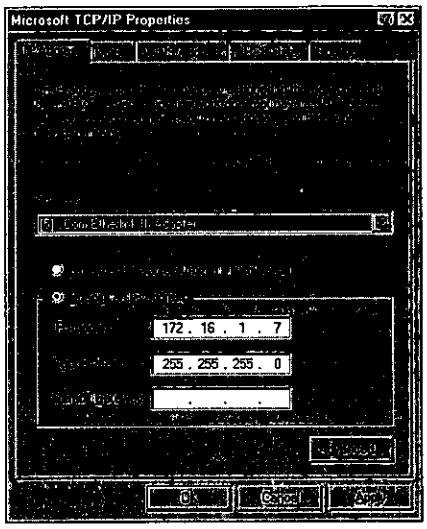



**Figure 12-2:** Microsoft TCP/IP Properties dialog box.

More complex configurations become necessary when your network is larger, and therefore, more complicated. The DNS tab is where you can define IP addresses for one or more Domain Name System (DNS) servers. DNS servers resolve domain names into IP addresses. For more information about DNS, please consult the section on DNS later this chapter.

The WINS Address tab is where IP addresses for Windows Internet Name Services (WINS) servers are defined. WINS servers resolve NetBIOS names into IP addresses. WINS is convenient for Windows NT networks with multiple servers and network segments. For more information about WINS, please consult the section on WINS later this chapter.

The DHCP Relay tab is where Dynamic Host Configuration Protocol (DHCP) server addresses may be defined. Using a Windows NT Server computer to relay DHCP messages permits clients on multiple cable segments to access a single DHCP server; otherwise, a DHCP server must be present on each network segment for DHCP to work. A DHCP server simplifies managing IP-based networks by assigning IP addresses and configurations to DHCP client systems on demand. For more information about DHCP, please consult the section on DHCP later this chapter.

The Routing tab is used only on Windows NT computers that include two or more NICs, where each NIC is connected to a separate subnet. If you check the "Enable IP Forwarding" box on this tab, it allows traffic from one subnet to travel through the server to another subnet. This capability is what allows Windows NT to act like a router. For more information on routing, please consult the Windows NT Server Resource Kit and TechNet.

# Automating IP Addressing with DHCP

DHCP, the Dynamic Host Control Protocol, is used to dynamically assign IP addresses and other configuration settings to systems as they boot, which allows clients to be automatically configured each time they boot, thus reducing installation administration. DHCP also allows a large group of clients to share a smaller pool of IP addresses, if only a fraction of those clients need to be connected to the Internet at any given time.

## What's DHCP?

DHCP is a service that Windows NT Server can deliver. In other words, a Windows NT Server can run DHCP server software to manage IP addresses and configuration information for just about any kind of TCP/IP client.

DHCP manages IP address distribution using leases. When a new system configured to use DHCP comes online and requests configuration data, an IP address is leased to that system (by default, each lease lasts three days). When the duration of the lease is half expired, that client can request a lease renewal for another three days. If that request is denied or goes unanswered, the renewal request is repeated when 87.5% and 100% of the lease duration has expired. If a lease expires and is not renewed, that client can't access the network until it obtains a new IP address lease. You can initiate manual lease renewals or releases by executing `ipconfig /renew` or `ipconfig /release` at the Windows NT Command Prompt.


You can view the current state of IP configuration using the `ipconfig` command. Issuing the command `ipconfig /all |more` at the Command Prompt displays all a machine's IP configuration information one screen at a time.


## Configuring DHCP (Not for the timid!)

To install DHCP, use the Services tab in the Network Control Panel applet. (Click the Add button, select Microsoft DHCP Server from the Select Network Service window, then supply the Windows NT Server 4.0 CD when prompted.) When DHCP is installed, a new entry named DHCP Manager appears in the Administrative Tools (Common) menu.

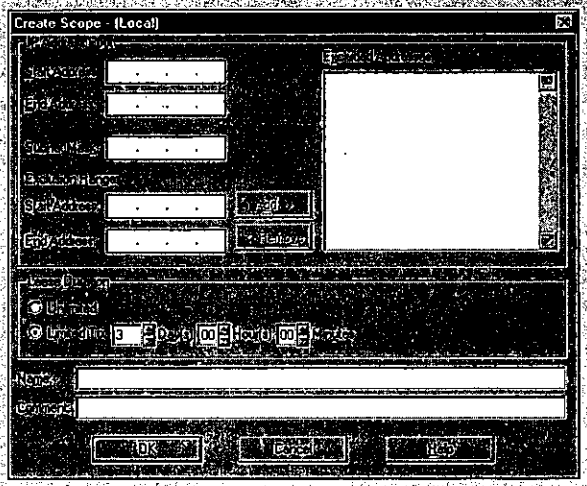
To configure a DHCP server, use the DHCP Manager's Create Scope (Local) window, which appears in the figure. Here, you manage a single collection of IP addresses and lease properties, known as a DHCP scope.

The Start Address and End Address that appear in the upper-left corner of the Create Scope window define the range of addresses that the DHCP server will manage. This list is inclusive, so the values supplied for Start and End will also be used as part of the pool.

To the right, a list of Excluded Addresses may appear. This window shows any IP addresses you instruct the DHCP server to ignore within the range defined by the Start and End addresses. Normally, you need to exclude addresses that are already allocated to servers, routers, and other devices that require fixed IP addresses.

The Subnet Mask field simply defines the subnet mask for the pool of addresses in the current DHCP scope.

The Exclusion Range fields let you define a range of addresses to exclude from the IP address pool. To exclude a single address from the pool, use the same IP address for the Start Address and End Address fields. To exclude a range of contiguous addresses in the pool from use, enter the first address to be excluded in the Start Address field, and then enter the final address to be excluded in the




(continued)

*(continued)*

Stop Address field. You can use the Add button as many times as you must, so that you can exclude as many individual addresses, or ranges of addresses from the address pool as you must.

The Lease Duration can be set to unlimited, which means that the lease works just like a static allocation except it's managed by a DHCP server. Or, you can set a more limited duration for IP address leases. Notice that the default lease duration is set to three days.

You can also give a DHCP scope a name, and associate a comment with that name. This kind of information will come in handy if you work on a large, complex network with multiple DHCP scopes, and need to be able to tell them apart easily.

## Is DHCP in your future?

We can think of two profound reasons why DHCP is a real godsend to Windows NT administrators like you:

1. DHCP allows you to manage an entire collection of IP addresses in one place, on a single server, with little effort beyond the initial configuration of the address pool. In the bad old days before DHCP, managing IP addresses usually required walking from machine to machine on a far too frequent basis.
2. DHCP automates delivery of IP addresses and configuration information (including subnet mask and the default gateway addresses) to end-user machines. This makes it astonishingly easy to set up IP clients, and to handle configuration changes when they must occur.

   To configure IP on a new client, all an end user (or you) must do on Windows NT, Windows 95, or Windows 98 is click the single radio button on the IP Properties window that says "Obtain an IP address from a DHCP server." DHCP does the rest automatically.

   When configuration changes occur, these changes will be automatically introduced when IP leases are renewed. You can even cancel all existing leases and force clients to renew their leases whenever major renumbering or configuration changes require immediate updates to their IP configurations.

The ultimate reason for using DHCP is because it makes your job much easier. DHCP is recommended for all networks that use TCP/IP with ten or more clients.

# Some You WINS, Some You Don't

Within a Microsoft Windows network, TCP/IP hosts can called by NetBIOS names instead of IP addresses or domain names. Because NetBIOS names

are more or less unique to Microsoft Networks, there is no current standard for associating NetBIOS names with IP addresses. On a Microsoft Network that uses TCP/IP as its only networking protocol, it is essential to be able to resolve NetBIOS names to IP addresses.

## Understanding WINS

Because resolving NetBIOS names to IP addresses is the key to providing access to many of Windows NT's built-in services and facilities, Microsoft provides two methods to handle this process:

1. Using a file named LMHOSTS (here, LM stands for LAN Manager, and points to the network operating system that preceded Windows NT in the Microsoft product world), you can create a static table that associates specific NetBIOS names with specific IP addresses. Such a file must be present on every machine to provide the necessary name to address resolution capabilities.

   For small, simple networks, using LMHOSTS files is an acceptable method. On large, complex networks, the busy work involved in maintaining a large number of such files can quickly get out of hand.

2. Larger, more complex networks are where WINS comes into play. WINS stands for Windows Internet Name Service. WINS runs on Windows NT Server machines as a service that automatically discovers NetBIOS names and manages a dynamic database that associates NetBIOS names with TCP/IP addresses as networks grow in size, multiple WINS servers sometimes become necessary to help speed up the time it takes to handle name resolution requests.

   A single WINS server can handle an entire network. But on networks that include multiple sites, or with many thousands of users, multiple WINS servers can distribute the load involved in providing name resolution, and speed users' access to NetBIOS-based resources.

WINS has several advantages over LMHOSTS files. For one thing, it's built on a dynamic database, which means that as networks change and names and addresses come and go, the database changes as the WINS server detects new name and address relationships, or finds old names with new addresses. WINS can be especially important on networks where DHCP is in use, if clients also share files or printers on their machines. Also, WINS is something like a multilingual Spanish-English dictionary that's constantly updated as new words are added. You give it a Spanish word, and out pops an English word or a translation that means the same thing!

## WINS servers

A WINS server maintains a database that maps computer names to their respective IP addresses and vice-versa. Rather than sending out broadcasts

for address information, which eats excess network bandwidth, a workstation that needs a NetBIOS name resolved makes a request directly to a designated WINS server (that's the real purpose of the WINS tab in the TCP/IP Properties window, in fact).

This approach lets workstations take advantage of a well-defined service and obtain address information quickly and efficiently. Also, when workstations with NetBIOS names log onto the network, they provide information about themselves and their resources to the WINS server, that that any changes will automatically appear in the WINS server's database.

## WINS clients

When configuring workstations or servers (at least, those servers that don't play host to the WINS server software) on your network, you'll provide an IP address for one or more WINS servers on your network. When those machines boot, they provide the WINS server with their computer names, share names, and IP addresses. The WINS server handles everything else. If a workstation needs an IP address that corresponds to a NetBIOS name, it asks the WINS server to supply that information.

# DNS

One way to simplify TCP/IP host identification is to use Fully Qualified Domain Names (FQDNs) instead of IP addresses. An FQDN is the type of name that's used to identify resources on the Internet to make access easier for humans (such as `http://www.microsoft.com`). Resolving domain names and FQDNs to IP addresses is a crucial service on TCP/IP networks in general — especially on the Internet, where hundreds of millions of names and addresses may be found. This is where the Domain Name System, sometimes called the Domain Name Service, always abbreviated as DNS, comes into play.

As with NetBIOS names and IP addresses, the association between FQDNs and IP addresses can also be maintained in two ways:

1. A HOSTS file can be created on each system. The HOSTS file maintains a local table that associates specific FQDNs with specific IP addresses. Whenever such associations change, the HOSTS file must be updated manually and copied to all machines on a network.

   HOSTS files are not well-suited for interaction with large IP-based networks, especially the Internet. This explains why HOSTS files are mostly a historical relic of an earlier, simpler era of IP networking. Except as a fallback should access to DNS fail, nobody uses HOSTS files anymore.

2. Access to a DNS server permits network machines to request name resolution services from that server instead of maintaining name to address associations themselves. Although DNS servers must be configured manually, a DNS server can handle the name resolution needs of an entire network with ease. DNS servers can also communicate with one another, so that a name resolution request that the local server cannot handle can be passed up the FQDN name hierarchy until it reaches a server that can resolve the name into an address, or indicate that the name is invalid.

The Internet includes tens of thousands of DNS servers. ISPs manage many of these DNS servers; others fall under the control of special top-level domain authorities. To stake out an Internet presence, you must obtain a unique FQDN through the InterNIC (or let your ISP do it for you). After you obtain this name, it will be associated with a special root IP address in some DNS server (probably at your ISP, unless you decide to set up a DNS server of your own).

## DNS: No or yes?

Unless you're managing a large, complex network, chances are better than average that you will be working with somebody else's DNS server — probably, your ISP's — rather than managing your own. But if you have a large network with over 1,000 computers, or your network spans multiple sites using private wide-area links, a DNS server may be just the thing to help you stake out the right kind of Internet presence.

If you think you may be interested in setting up a DNS server, you need to consult a technical resource, such as the Windows NT Server Resource Kit or TechNet. We also highly recommend *DNS on Windows NT,* a book by Paul Albitz, Matt Larson, and Cricket Liu (O'Reilly & Associates, Sebastopol, CA, 1998, ISBN 1-56592-511-4) as the ultimate resource for using Windows NT as a DNS Server. Albitz and Liu also wrote a general book called *DNS and BIND,* now in its third edition (O'Reilly & Associates, Sebastopol, CA, 1998, ISBN 1-56592-512-2) that is widely regarded as the best general reference on DNS.

# Dealing with Difficulties

Problems occurring on TCP/IP networks are almost always associated with incorrect configurations. The wrong IP address, subnet mask, default gateway, DNS, WINS, or DHCP server can bring a system, if not a whole network, to its knees. Therefore, you need to take extra caution to double check your settings and changes before putting them into effect.

If you are connecting to an ISP, you should contact the ISP's technical support personnel early on to eliminate as much "wheel-spinning" as possible. You may discover the problem is not on your end, but theirs. If so, your only recourse is to wait it out, then complain. If problems occur too often for your comfort at an ISP, take your business elsewhere.

Windows NT includes a few TCP/IP tools you can employ to help track down problems. We already mentioned ipconfig, so here are the others:

- **PING:** This tool tests the communication path between your system and another remote system. If a ping returns, you know the link is traversal and the remote system online. If the ping times out, then either the link is down or the remote system offline.
- **TRACERT:** This tool reveals the hops (systems encountered) between your system and a remote system. The results inform you if your trace route packets are getting through and at what system is a failure occurring.
- **ROUTE:** This tool is used to view and modify the routing table of a multi-homed system.
- **NETSTAT:** This tool displays information about the status of the current TCP/IP connections.
- **TELNET:** This tool is used to establish a text-based terminal emulation with a remote system. Telnet gives you access to remote systems as if you were sitting at its keyboard. Windows NT Server does not include an inbound Telnet server.

Complete details on these tools are included in the Windows help files, the Windows NT Server Resource Kit, and TechNet.

# Enough TCP/IP to Choke a Hippo

If this chapter has whetted your whistle for TCP/IP, there are lots of great resources where you can obtain more details and information:


- Bisaillon, Teresa and Brad Werner. *TCP/IP with Windows NT Illustrated.* McGraw-Hill, 1998. ISBN: 0079136486.
- Stevens, W. Richard. *TCP/IP Illustrated Volumes 1,2 & 3.* Addison-Wesley, 1994. ISBN: 0201633469, 020163354X, 0201634953.
- Comer, Douglas E. *Internetworking with TCP/IP, Vol. I-III.* Prentice Hall ESM, 1995, 1996, 1997. ISBN: 0132169878, 0139738436, 0138487146.
- Wilensky, Marshall and Candace Leiden. *TCP/IP For Dummies,* 3rd Edition. IDG Books Worldwide, Inc., 1999. ISBN: 0764504738.