PART III

Beyond the Basic LAN

CHAPTER 13

Sharing Resources

The Network+ Certification exam expects you to know how to

- 3.1 Identify the basic capabilities (for example: client support, interoperability, authentication, file and print services, application support, and security) of the following server operating systems to access network resources: UNIX/Linux/Mac OS X Server, NetWare, Windows
- 3.2 Identify the basic capabilities needed for client workstations to connect to and use network resources (for example: media, network protocols, and peer and server services)

To achieve these goals, you must be able to

- Understand the naming of shared resources using Universal Naming Convention (UNC) and Universal Resource Locater (URL)
- Learn about permissions for Windows 9*x*, Windows NT, Windows 2000/2003, Windows XP, NetWare 3.*x*, NetWare 4.*x*/5.*x*/6.*x*
- Understand sharing resources as it applies to the preceding operating systems
- Understand accessing shared resources

In every functional network, server systems share resources and client systems access those shared resources. This chapter looks at the different ways the common network operating systems enable servers to share resources, concentrating on folder and printer sharing in the different versions of Microsoft Windows, Novell NetWare, and Linux/UNIX. Then you'll see how to configure Windows 9*x*, 2000, and XP clients to access those shared resources. After all, an installed network of servers and client computers is useless without resources for the serving systems to share and the client systems to access!

The basic steps of making any resource sharable are pretty much the same whether you're sharing a folder on your C: drive to a small network or a huge web site to the entire Internet. To share a resource, you need to make it sharable and give it some name. How shared resources are named varies, depending on the operating system and the type of resource shared.

Historical/Conceptual

Resource Naming

Resource naming falls into one of two types: naming conventions invented by Novell and Microsoft, and the resource naming conventions we use for TCP/IP-based stuff. Windows and NetWare were sharing folders and files on their own internal LANs long before they moved out into the Internet world, and as a result have a different way to look at folder and printer sharing than what we might see on the Internet. Let's begin by understanding how Windows and NetWare name-shared resources.

Scott decides to share his C:\Half-Life folder on his Windows XP Professional system; how do his pals on the network know this resource is available for use? Let's assume that his system is running the correct protocol, is properly connected to the network, and his computer has the name Scottxp. That's half the battle, but a server name alone does not work. Each shared resource must also have a name.

The combination of server name and shared resource name gives people wanting to use the resource something to point at to select a specific resource. Windows clients use the Network Neighborhood/My Network Places tool to browse a network for available resources. Figure 13-1 shows the My Network Places folder on my PC. I can see all of the systems currently sharing resources, including Scott's PC (Scottxp). Double-clicking the Scottxp icon displays all the shared resources on his system, as shown in Figure 13-2.

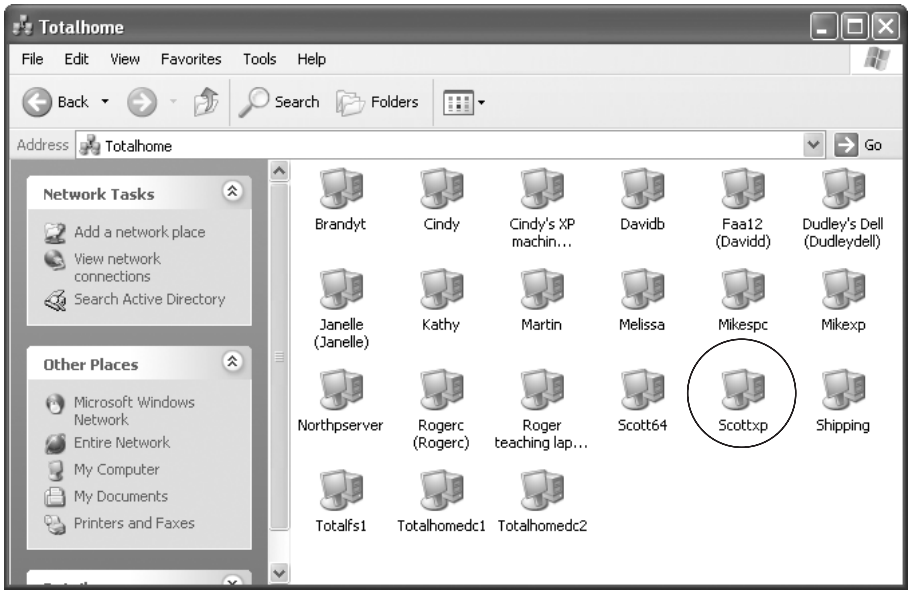


Figure 13-1 My Network Places showing Scott's PC

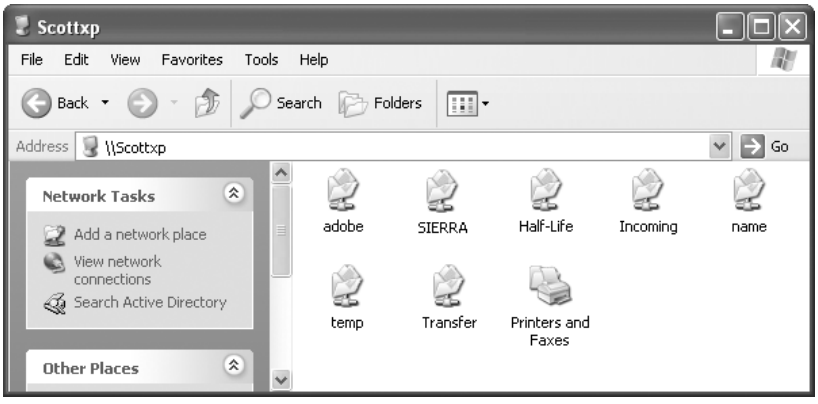


Figure 13-2 Shared folders on Scott's PC

NOTE Windows 9*x* and NT call this folder Network Neighborhood; Windows Me, 2000, 2003, and XP call it My Network Places. For the remainder of this chapter, I'm going to keep things simple and call it My Network Places, regardless of the operating system.

You can easily see the Half-Life folder in Figure 13-2, because Scott chose to call his shared folder "Half-Life." That's not always the case. The name by which a resource is shared on a network does not have to be the same name as the actual resource. We call the name of the shared resource on the network the *network share name*. A shared resource's network name is not, and often cannot be, the same as its real name. Scott's shared C:\Half-Life folder has the network name Half-Life, but he could just as easily have called it "TIMMY" or "ScottGame" or just about anything else within the limits of the NetBIOS or DNS naming conventions. To access this folder, just double-click it; then, assuming no security restrictions prevent it, you can access the files and subfolders on this share.

NOTE Using the term "NetBIOS naming conventions" often makes students' eyes roll back in their heads, especially when it means simply "normal" names, made with almost any combination of alphanumeric characters. You cannot use spaces or the following characters in names: \ / : * ? " ; |

Test Specific

UNC

Windows' My Network Places makes browsing through a network easy to do. Simply by clicking a group, serving system, or shared resource, you can access whatever you want—or at least whatever you're allowed to access. But networking hasn't always been about Windows client systems. Long ago, before Windows even existed, Microsoft championed the concept of the *Universal Naming Convention (UNC)*, which describes any shared resource in a network using this convention:

```
\\<server name>\<name of shared resource>
```

DOS programs (pre-Windows, remember?) accessed shared resources using commands typed at a command prompt. DOS systems needed UNC names to access shared resources. Let's say someone wanted to access Scott's C:\Half-Life folder. The UNC name you'd type to access his system would be \\Scottxp, and the UNC name for the shared folder would be \\Scottxp\Half-Life. If you were using the old DOS-based NOS called LAN Manager, you'd have to type strange commands at the C: prompt, like

```
NET use y:  \\Scottxp\\half-life
```

This command, in a process known as *mapping*, creates a Y: drive on the client system that's really the \\Scottxp\Half-Life share. All versions of Windows still support drive mapping. Fortunately, you no longer need to type strange commands at command prompts—although you still can if you want to! Just alternate-click (right-click) the folder in My Network Places and select Map Network Drive to get a wizard (in some versions of Windows); or open My Network Places in Windows XP and select Map Network Drive from the Tools menu to start the wizard (see Figure 13-3). Select a drive letter for the drive—and in Windows XP, browse to the shared folder—and the mapped share will appear like magic in your My Computer folder! Figure 13-4 shows the shared folder \\Scottxp\Half-Life mapped as the Y: drive on my system under My Computer. Windows XP is even nice enough to change the icon slightly and to list it separately from my local drives, so you know the folder is a *mapped drive*—can you see the difference?

Although Windows systems still support mapping, Windows applications can access shares directly by their UNC names, so you no longer need to map a shared folder to a drive letter. Even though mapping is not nearly as common as it once was, you'll still see it used in some networks, usually for security reasons or to support some older application that needs to access a drive letter and not a UNC name. Mapping, as well as a number of other share functions, simply would not work without UNC.

Make sure you can recognize a valid UNC name. They always begin with a double backslash (\\ ) followed by the name of the serving system, and then a single backslash (\ ) followed by the name of the shared resource.

Figure 13-3 The Map Network Drive wizard in Windows XP

TIP Make sure you can tell a valid UNC name from an invalid one!

UNCs are not limited to shared folders and drives. You can also use a printer's UNC to connect to a shared printer. This process, similar to mapping, is called *capturing* a printer. A captured printer uses a local LPT port that connects to the networked printer. Like mapping, this is usually only done to support older programs that are not smart enough to know how to print directly to a UNC-named printer; it's quite rare today. Back in the

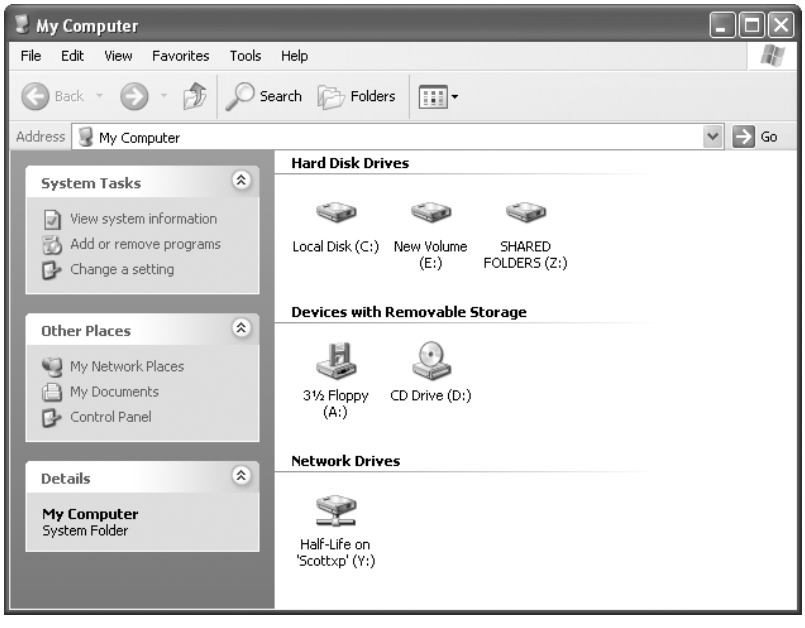


Figure 13-4 The \\Scottxp\Half-Life share mapped as Mike's Y: drive

old days, we could capture a printer just like a shared folder, using the **NET** command in Windows:

```
NET use LPT1 \\Tim\Printer
```

NOTE The CAPTURE command was the NetWare equivalent to the NET command in Windows.

Even though we rarely use these ancient commands to map folders and capture printers, UNCs are still very much part of the networking world, especially with Windows systems. Windows support for UNCs goes deep; almost any application in your system that has to do with locating a file or folder will read UNCs. Try opening either Internet Explorer or Windows Explorer and typing a known valid UNC name in the address area—the corresponding network folder will open. Figure 13-5 shows what happens when I type a UNC into the address bar of Internet Explorer.

URL

Although Microsoft developed UNC names to work with any shared resource, these days, they're mostly just used with folders and printers. Other shared resources like e-mail and web browsers use the more common, and more Internet-aware, *Universal Resource Locator (URL)* nomenclature. You'll learn more about URLs in Chapter 15, "TCP/IP and the Internet."

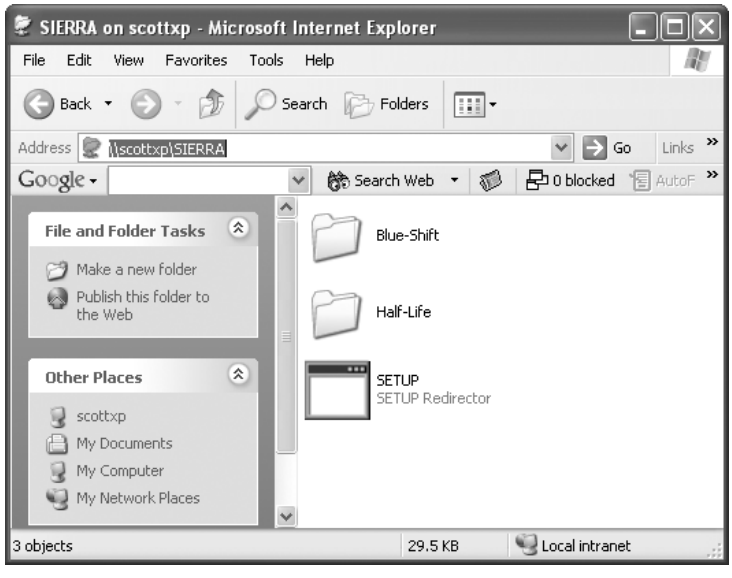


Figure 13-5 A UNC typed into the address bar of Internet Explorer

NOTE You will find that the *U* in URL can stand for either *uniform* or *universal*. While *universal* was the early choice, *uniform* is probably ahead in usage now. Neither one is "wrong" per se.

The rest of this chapter details how different network operating systems share resources, and then shows how to access those shared resources. We'll be messing with URLs like crazy, but before we do that, you need to understand the major concept of permissions.

Permissions

Once you've set up a resource for sharing and given it a network name, how do you control who gets to access it and what they can do to it? You know the answer to this one from Chapter 12, "Network Operating Systems," right? Permissions, of course! I touched on permissions—or *rights*, as they're called in Novell NetWare—in Chapter 12, but now it's time to go into them in more detail. I've included a bit of background here that'll sound familiar, but it could be a useful refresher if you've slept since reading the last chapter.

As you know, *permissions* are sets of attributes network administrators assign to resources to define what users and groups can do with them (the resources, not the admin!). A fairly typical permission used in all network operating systems is a permission assigned to folders called *execute*. The execute permission, as its name implies, enables the user or group that has it to execute, or run, any programs in that folder.

NOTE The execute permission in a Linux/UNIX environment enables you to view the contents of a directory, as well as run programs in that directory.

Types of permissions vary, depending on the resource being shared. If I share a printer, I certainly don't need a permission called execute, although I do admit wishing I could have executed a few troublesome dot-matrix printers in the past! Instead, printers usually have permissions like manage printer that let certain users or groups reset the printer or start and stop print jobs.

There are many, many more permission types than the two I described here. I just wanted you to get an idea of what a couple looked like. One of the most fascinating aspects of permissions is the different ways network operating systems utilize them. Let's look at the more common network operating systems and appreciate how they use permissions.

Dueling Security Models

The first thing to understand here is that Windows 9*x* is a freak of nature when it comes to networking and permissions. It does permissions one way, and all the other operating systems we'll be discussing do them another (better) way. It all boils down to the differ-

ence between the resource-based security model, and the server- and organization-based models. The server- and organization-based models have two layers of security in between a user and the resource he wants to access, while the resource-based model has only one.

As you'll recall from Chapter 12, "Network Operating Systems," in the server and organization models, all users must log on before they can access shared resources. In the server-based model, users log into each server for access to the resources it controls; in the organization-based model, users log in once for access to the entire network. Either way, after a user has logged on successfully, that user account receives an electronic key it can show to serving systems on the network when it wants access to specific resources.

Hillary works for a super-secret spy agency. When she walks in the front door, she shows the badge to a guard to prove that it's okay to let her in. The guard then gives her a special electronic card with a secret code on it. The code specifies where she can go and what she can do during that particular visit. Every time Hillary wants to access a particular area, a security device checks her card to see what permissions she has. The type and extent of her access to the resources in that area will be determined by the specific permissions encoded on her card. What I've just described is a two-layer security model: First, you must get in the door of the building (log onto the network); second, you must have the necessary permissions to access the various resources inside (see Figure 13-6).

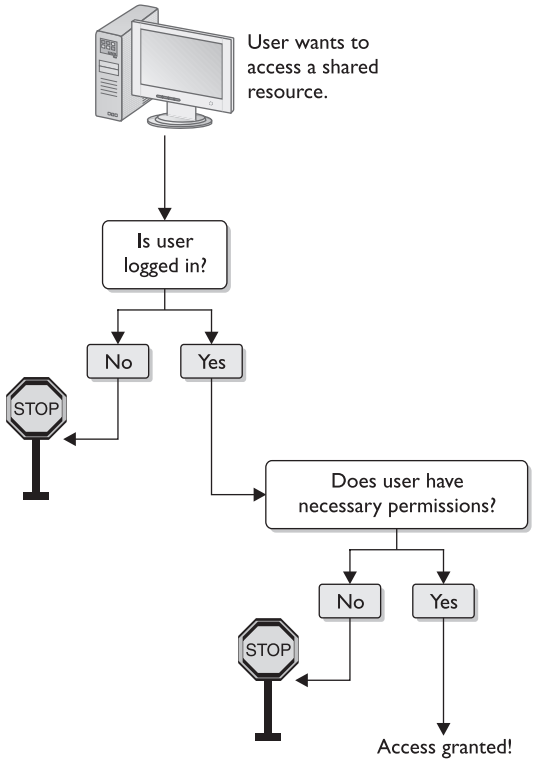


Figure 13-6 Two layer security: log onto the network and have permissions.

In resource-based security models, there's only one layer of security. Returning to my hypothetical situation, it would be as if there were no guard at the door—absolutely anyone could walk in to the building without anyone knowing who they were or when they came and went. The only security barrier between them and any particular area in the building would be the door to that area, which would either be locked, unlocked, or protected by a password code.

Windows 9*x* Permissions

All Windows 9*x* systems use the resource-based security model. The only security options are Full (the door is open—do what you want), Read-Only (you can look, but not touch), and Depends On Password (Full access requires a password). Interestingly, Windows 9*x* doesn't have a No Access option. Think about it—there's no point in having a resource nobody can access ever, but because 9*x* doesn't have real user accounts, it does not know who's knocking at the door. Your only choices with Windows 9*x* are (a) let everyone in to do whatever they want; (b) let everyone in, but only let them look; or (c) let everyone in to look, but only those who know the password can do whatever they want.

Additionally, these permissions, called *share permissions*, only control the access of other users on the network with whom you share your resource; they have no impact on you (or anyone else) sitting at the computer whose resource is being shared. These Windows 9*x* permissions are prehistoric in networking terms; they date from the days of LANMan 1.0 (the first NOS for PCs), before anybody could imagine a need for more than this basic amount of security.

Clearly, they were wrong about that. Gone are the days when computer security meant locking the computer room door! Today's networks have to be secured against all enemies, foreign and domestic. Network administrators need to be able to keep track of who can use their networks, and in what specific ways, regardless of whether the person is sitting at the serving system itself or dialing in from a country half-way around the world. Truly useful security also requires a more powerful and flexible set of permissions. Modern network operating systems like Windows NT/2000/2003/XP and Novell NetWare implement robustly featured user accounts, as you'll recall from Chapter 12, "Network Operating Systems." User accounts enable a network admin not only to control initial access to the network, but also to fine-tune any user's access to every resource being shared.

TIP Windows 9*x* does have user accounts, but those accounts exist only to enable a user on a Windows 9*x* client on a network to log into a Windows NT/2000/2003 server. The user account login provides absolutely no protection for the local machine itself. I'll let you in on a secret: you can just press the ESC key at the logon screen and Windows 9*x* will shrug its shoulders and let you have full access to its local resources.

Windows NT Permissions

If Windows 9*x* is a freak of nature, Windows NT has multiple personalities. Windows NT can handle security in two completely different ways.

Windows NT file and folder permissions are based on the powerful NT file system (NTFS) file format. When you format a partition in Windows NT, you can choose from two file formats: the old FAT partition used by Windows 9*x* systems, or NTFS. Figure 13-7 shows the Windows NT Disk Administrator tool—note the two NTFS partitions and one FAT partition. You don't have to use NTFS to format an NT volume, but if you choose not to format a partition on a Windows NT system with NTFS, you will lose all of the security that NTFS provides. You will be reduced to the Windows 9*x* share permissions just described. Of course, pretty much everyone uses NTFS on their Windows NT/2000/2003/XP systems nowadays!

NOTE Windows 2000, 2003, and XP can use three different file systems: FAT, FAT32, and NTFS. Just as with FAT, FAT32 offers no security benefits to the system and thus should be avoided in most circumstances. Everything discussed in this section about NT's use of NTFS also holds true for Windows 2000, 2003, and XP.

NTFS embeds the powerful NTFS permissions into each shared resource. This does *not* mean that the resource itself handles security. That job goes either to the individual NT serving system or to the NT domain, depending on how the NT network is configured. This is a critical point and one that is often lost on folks new to more advanced network operating systems. If the network isn't running any copies of Windows NT Server, Windows 2000 Server, or Windows Server 2003, each system on the network must act as its own server; this means users must have an account on each system they want to access (see Figure 13-8).

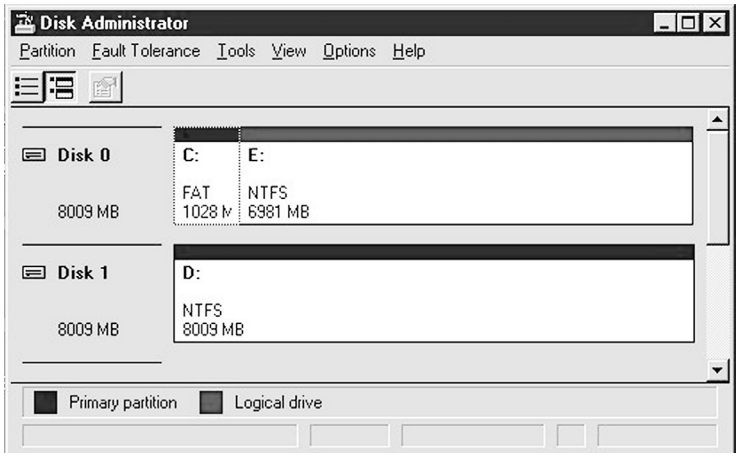


Figure 13-7 The Windows NT Disk Administrator tool

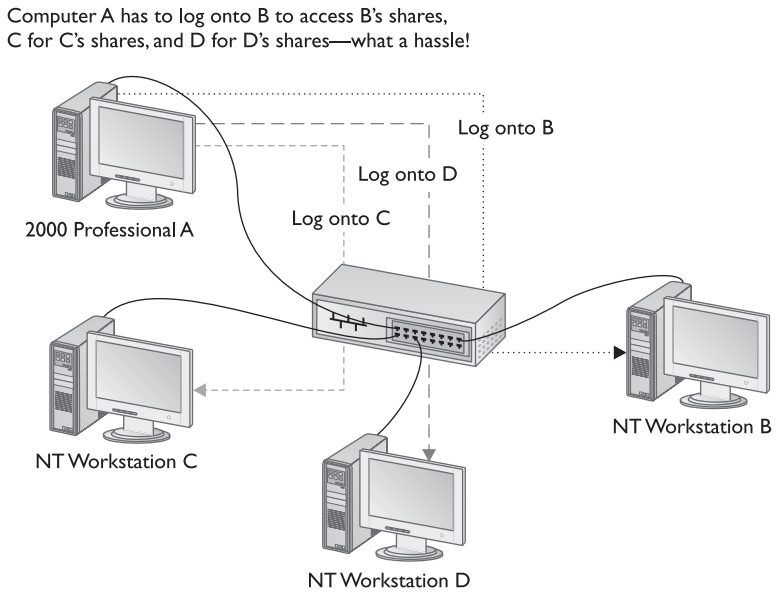


Figure 13-8 Logging onto each sharing system separately

Once you install a copy of Windows NT Server, Windows 2000 Server, or Windows Server 2003 and implement a domain, each user gets a domain user account that gives them access to the network in one quick logon (see Figure 13-9). In a domain-based network, no one has a local user account—all users get domain user accounts that must be set up by a special program on the Windows NT, 2000, or 2003 server system. Local accounts still exist in a Windows NT, 2000, or 2003 domain-based network (except for domain controllers, which have no local user accounts), but are rarely used except for perhaps an occasional maintenance function. To log on locally to a system that uses a domain, you must perform a special local logon. In fact, the Windows NT/2000/XP logon gives you the ability to log into the domain or just to the local system (see Figure 13-10). We log into a local system only to perform maintenance.

Table 13-1 lists Microsoft's standard *NTFS permissions* for files and folders under Windows NT. These standard permissions are groupings of what Microsoft calls special permissions that have names like Execute, Read, and Write. These special permissions are rarely accessed directly in most NT environments, but you can find these permissions in a resource's Properties. Figure 13-11 shows the special permissions for a folder.

The beauty of NTFS is that it doesn't matter to the serving system if you log in locally, log in over the network, or log into a domain. NTFS permissions work the same way whether the NT/2000/2003/XP system is on a network or running as a stand-alone system. If only one Windows NT, 2000, 2003, or XP system existed in the universe, you would still need a user account and NTFS permissions to access anything on the system.

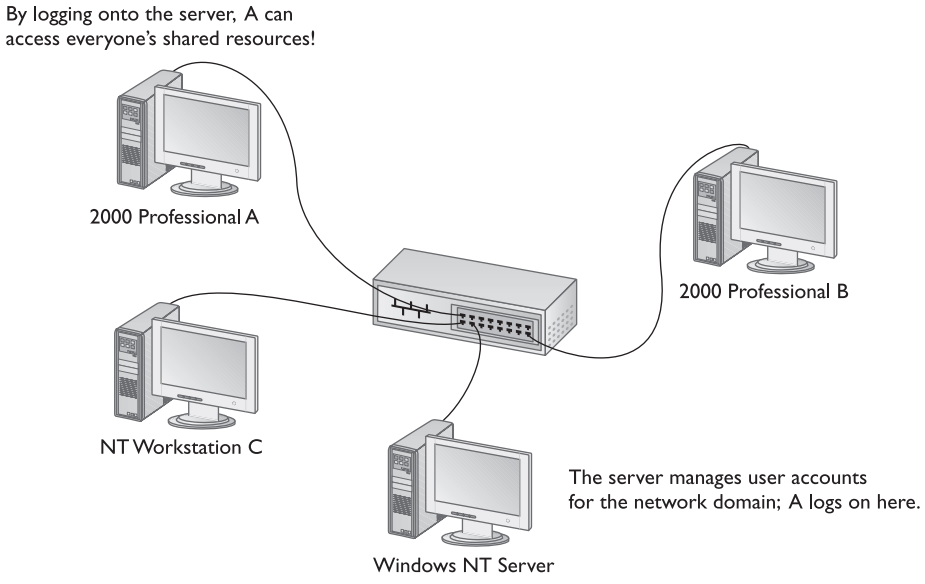


Figure 13-9 Logging onto the server does it all!

To differentiate these permissions from the share permissions, we call them NTFS permissions or local permissions.

So how can NTFS be used in a network? Simple! NTFS resources can store information on any user account. The account can be local just to that system, or if the system is part of a Windows NT/2000/2003 domain, it can be a domain user account. The only difference is whether the local system handles the user account logons, or the domain does—either way, it doesn't matter to NTFS!

Figure 13-10 The Windows 2000 logon screen showing both local and domain logon options

Permission	Folders?	Files?	What Does It Allow?
No Access	Yes	Yes	Denies all access to the file or folder. Users can see the file or folder, but cannot access it in any way.
List	Yes	No	Users can only see the contents of the folder and go to subfolders.
Read	Yes	Yes	Users can read files and folders, open files and subfolders, but cannot change any files.
Add	Yes	No	Users can add files or subfolders to the folder, but cannot open or change any files.
Add & Read	Yes	No	Users can read and add files or subfolders to the folder. Users can also open files in the folder.
Change	Yes	Yes	Users can do anything but delete the file or folder. They cannot change permissions on any files or subfolders.
Full Control	Yes	Yes	Users can do anything they want.

Table 13-1 NTFS Permissions for Files and Folders Under Windows NT

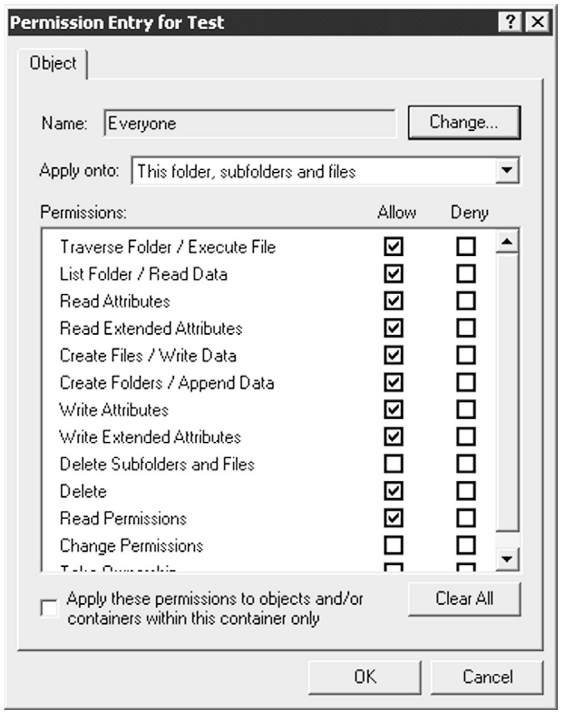


Figure 13-11 NTFS special permissions for a folder in a Windows NT system

Whoa! Wait a minute, Mike! Are you telling me that Windows NT/2000/2003/XP systems have two different types of user accounts? Yup, that's right! You get two types: local users and domain users. (Actually, there are more than two, but we don't need to cover that here.) To use domains, however, you have to buy a special server version of Windows, such as Windows NT Server, Windows 2000 Server, or Windows Server 2003.

All of this security is invisible to the network user as long as he has a good user account and the necessary NTFS permissions. This process is not unique to Windows networks—NetWare and Linux networks use the same two-step security method.

Share vs. NTFS Permissions

When you have a folder stored on NTFS-formatted hard drives in a networked PC, that folder has one or two levels of permissions that apply whenever a user tries to access it: NTFS permissions and (if the folder is shared and the user accesses the folder over the network) share permissions. The question always comes up at this point: what wins if the two sets of permissions are in conflict?

In the case of a conflict between share and NTFS permissions, the most restrictive permission always applies. Suppose John has a folder called Incoming on his NTFS drive, shared as INCOMING on the network. He sets the NTFS permission for the folder to Read-Only for Everyone, and changes the share permission to *Full Control*. When Mary accesses that folder over the network and tries to add a file, imagine her surprise when she gets only Read-Only access to that folder!

The reverse scenario would be true as well. If John changed the NTFS permission to Full Control for Mary's account, but changed the network share permissions to Read, Mary would get only Read access to that shared folder. The most restrictive permission always applies.

Windows 2000/2003 Permissions

From the permissions standpoint, Windows 2000 and 2003 work pretty much exactly the same way as Windows NT; however, if you look at Table 13-2, you will find a few subtle distinctions between the standard permission types. Take a good look at the Windows 2000/2003 permissions. What about the Write permission—why would anyone want that? You can add or edit a file, but you can't open it? That sounds crazy! Actually, it makes a lot of sense to folks who administer more complex networks. Imagine a network full of users who need to add files to a folder, but by the same token, we don't want them to see files others are adding. Trust me, it happens. NTFS permissions give administrators incredible control over exactly what a user can or cannot do to a file or folder, even though it may not be obvious how these permissions work. If you want to get into NTFS permissions, go for your MCSA or MCSE certification!

Windows XP Permissions

Windows XP offers several variations on permissions according to the version you use, Home or Professional, and whether you log into workgroup or a domain. Windows XP Home offers only *simple file sharing*, which gives users the capability to share a folder,

Permission	Folders?	Files?	What Does It Allow?
Deny Access	Yes	Yes	Denies all access to the file or folder. Users can see the file or folder but cannot access it in any way.
List Folder Contents	Yes	No	Users can only see the contents of the folder and go to subfolders.
Read	Yes	Yes	Users can read files and folders, and open files and subfolders but cannot change any files.
Write	Yes	Yes	Users can add files or subfolders to the folder but cannot open or change any files or subfolders.
Read & Execute	Yes	Yes	Users can read and add files or subfolders to the folder. Users can also open files in the folder.
Modify	Yes	Yes	Users can do anything but delete the file or folder. They cannot change permissions on any files or subfolders.
Full Control	Yes	Yes	Users can do anything they want.

Table 13-2 Windows 2000/2003 Standard Permission Types

and then decide if anyone accessing that folder on the network can change the files. Note the fairly self-explanatory pair of check boxes under the Network Sharing And Security section of Figure 13-12.

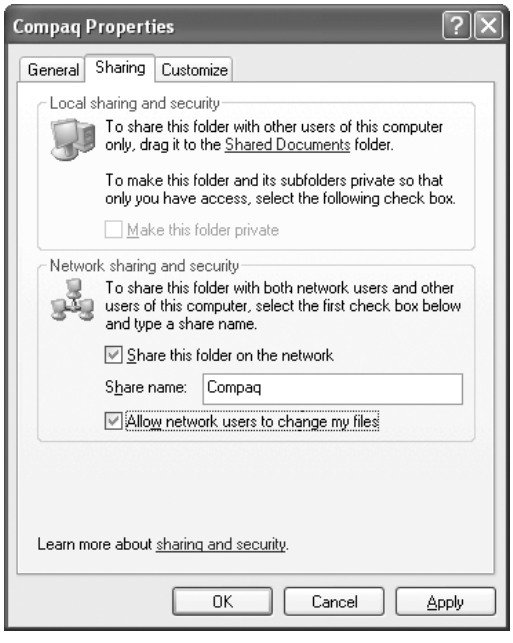


Figure 13-12 Simple file sharing in Windows XP Home

NOTE You can log into a domain only with Windows XP Professional, not with XP Home.

Even with NTFS, Windows XP Home offers no file or folder-level security once you share a folder over the network. It functions essentially like a less-secure version of Windows 9*x*, even lacking the capability to password-protect a shared folder!

Windows XP Professional in a workgroup environment by default uses simple file sharing, just like Windows XP Home, but you can disable this feature of dubious merit if you choose. Open Folder Options (either through the Control Panel applet of that name or select Tools | Folder Options | View tab in My Computer) and scroll all the way to the last option. Deselect the check box next to Use Simple File Sharing (Recommended) to disable (Figure 13-13).

Once you've disabled simple file sharing, Windows XP Professional offers the full range of sharing and security options available to Windows 2000 and Windows Server 2003. There's no difference. One thing to note, though, is that you need to be logged in with an Administrator account to change the file sharing in Folder Options *successfully.* Limited user accounts in Windows XP Professional can *appear* to change the sharing options, but Windows ignores the action. The box will remain unchecked, but Limited Users *cannot make network shares* in Windows XP!

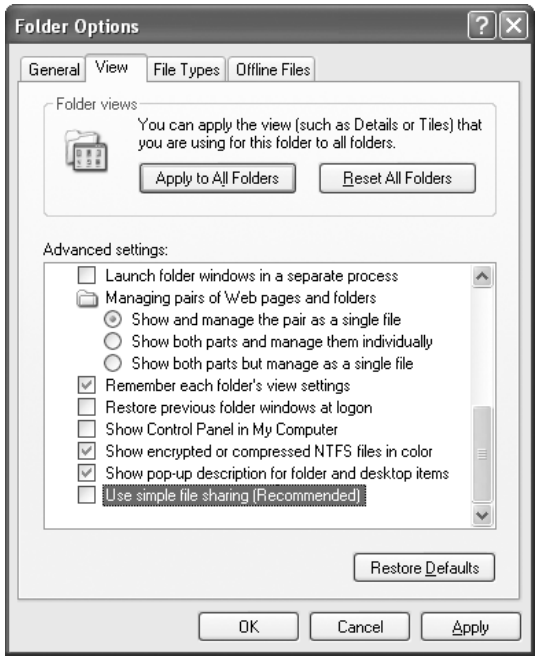


Figure 13-13 Disabling simple file sharing in Windows XP Professional

Windows XP Professional machine connected to a domain has only the option of full file sharing, à la Windows 2000 or Windows Server 2003. You can change the option in Folder Options, but Windows will ignore the selection completely. You have no simple file sharing when connected to a domain, even if you log into the PC locally.

NetWare 3.*x* Rights

Novell NetWare 3.*x* was the first NOS to adopt more advanced permissions. Novell calls its permissions rights. Unlike Windows' NTFS, each Novell NetWare 3.*x* server stores this information in its own *Bindery* (see Chapter 12, "Network Operating Systems").

Table 13-3 shows the NetWare 3.12 rights. Compare these to the Windows NT and 2000/2003 permissions. At first, they may seem quite different, but if you take your time and compare them, you'll see they're almost exactly the same.

NetWare 4.*x*/5.*x*/6.*x*

NetWare 4.*x*/5.*x*/6.*x* dispense with the Bindery, replacing it with *NetWare Directory Services (NDS)*. NDS controls access to network resources using network-wide permissions, rather than the file server–specific permissions used by NetWare 3.*x*'s Bindery. NetWare 5.*x* introduced a new file format called *Novell Storage Services (NSS)*. From the standpoint of sharing files and folders, NetWare has never changed from its original permissions. Isn't it nice when you can count on something to stay the same? Microsoft, are you listening? Hello? Oh well, let's move on.

Right	Folders?	Files?	What Does It Allow?
Read	Yes	Yes	Users can read files and folders, and open files and subfolders, but cannot change any files.
Write	Yes	Yes	Users can open and write to files.
Create	Yes	Yes	Users can add files or subfolders and can open or change any files.
Erase	Yes	Yes	Users can delete any file or subfolder.
Modify	Yes	Yes	Users can change the attributes of or rename files or subfolders.
File Scan	Yes	Yes	Users can see the file or the contents of the folder.
Access Control	Yes	Yes	Users can modify other users' and groups' rights to this file or folder.
Supervisory	Yes	Yes	Users can do anything they want.

Table 13-3 NetWare 3.12 Rights

UNIX/Linux

The concept of permissions can get confusing when switching between Linux and Windows. *UNIX/Linux* systems do have local file and folder permissions like NetWare and NT/2000, but they look quite different than the ones we've just seen. They do share one common feature with Windows, however: permissions are the same for both networked and local users. File-serving programs like FTP use the local permissions to handle network access permissions.

Unlike NetWare and Windows, UNIX/Linux provides only three permissions (see Table 13-4). Because they lack the more detailed permissions available in NetWare and Windows, most network administrators don't like to use UNIX/Linux systems for pure file sharing. I realize in saying this I'm risking an avalanche of indignant e-mail from the million or so UNIX/Linux users out there, but what can I do? When I'm right, I'm right! (I just won't have any friends.)

Sharing Is Sharing

For all the differences in names and functions among the different types of permissions, the bottom line is they all perform roughly the same functions: enabling those who administer networks to control the level of access to shared files and folders. Keep in mind that permissions are not at all limited to just files and folders—pretty much any shared resource on any network will have some type of permissions to assign to users and groups. Still, files and folders are the things we love to share the most, and once you appreciate the variations in the ways different network operating systems share files and folders, sharing other resources like printers will seem pretty anticlimactic!

Now that you've got a grip on permissions, let's put this knowledge to work and start sharing some files and folders. Oh, and by the way, let's go ahead and start sharing some printers too, while we are at it!

Sharing Resources

Sharing a resource involves three distinct steps. First, make sure your system is capable of sharing. Second, you need to share the resource and name it. Third, you need to set permissions on that shared resource.

Let's get one thing settled right now: No network NOS enables you to share individual files. Sure, you can share entire volumes or you can share folders in those volumes, but you cannot share a file! Don't confuse the capability to place permissions on a file with sharing a file. Just because you can't share a file doesn't mean you can't place per-

Permission	Folders?	Files?	What Does It Allow?
Read	Yes	Yes	Users can read files and folders, and open files and subfolders, but cannot change any files.
Write	Yes	Yes	Users can open and write to files.
Execute	Yes	Yes	Users can execute the file

Table 13-4 UNIX/Linux Permissions

missions on it. Sharing is a network function; permissions are unique to a resource. When you share a folder, you apply permissions to the shared folder, which are then attributed to the files and subfolders in that shared folder. This subtle difference can cause confusion in the unwary.

Sharing Folders

Because sharing folders is the area of biggest interest to most techs, let's start with them. We'll look at the network operating systems just discussed and see what you need to do to set up a resource for sharing. This is going to look a bit redundant—but who cares? The process never changes. Once you know what you have to do, all that remains are the specific details of how to create a share on a particular NOS.

Windows 9*x*

Remember when I said the first step in sharing a resource is to ensure that your system is capable of sharing? Well, every NOS discussed in this book is preset to share resources automatically, except Windows 9*x*. All Windows 9*x* systems require you to install and activate a special service called File and Print Sharing. To install this service on a Windows 9*x* system, access the Network Neighborhood Properties dialog box (see Figure 13-14). You can also access these settings by running the Network applet in the Control Panel. Make sure you know how to get to the Network settings in a Windows 9*x* client—we're going to be doing this a lot over the rest of the book! Do you see the File And Print Sharing button? Click it to see the sharing options (see Figure 13-15).

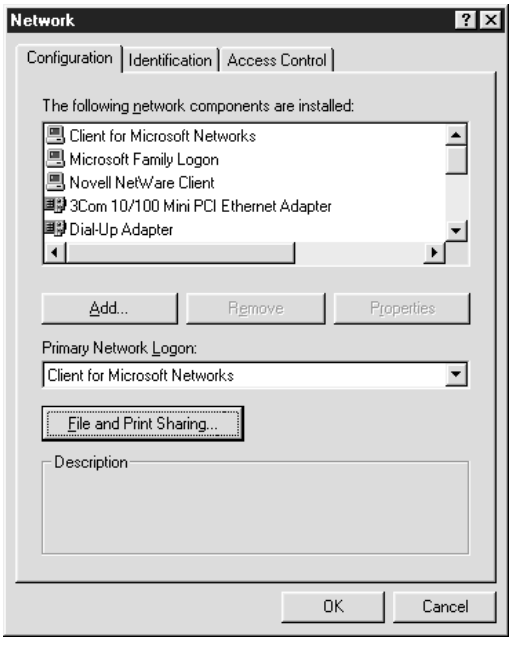


Figure 13-14 The Network Neighborhood Properties dialog box showing the system's network settings

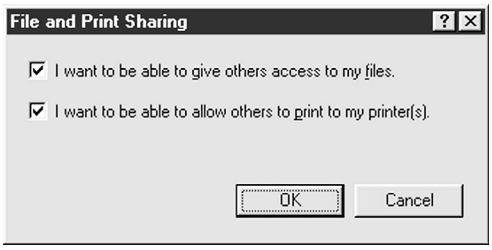


Figure 13-15 The File and Print Sharing options

This is pretty simple stuff here. If you want to share your files and folders, click the *I want to be able to give others access to my files* check box. Let's see if you get the idea. What should you check to allow others to access your printers? Hey! You are a genius! Once you've checked the boxes you want, click OK, and be ready with your installation CD—Windows will want it. And you can pretty well count on a reboot, too (hey, it's Windows). But that's it! You've completed the first step in sharing a resource: making sure the system is configured to share by installing the *File and Printer Sharing service*. A *service* is any program that runs on Windows, which you don't normally see. Your Windows 9*x* system may look the same as before, but trust me, a new set of programs is now running, even if you can't see them. We're all done setting up systems to share. You won't see this step again because all the other network operating systems do this automatically. Hooray!

Let's assume you've installed the File and Printer Sharing service on your system (see Figure 13-16). You can go back to Network properties to see if it's there. Once this is installed, you're ready to start sharing some files and folders! Wheeee!

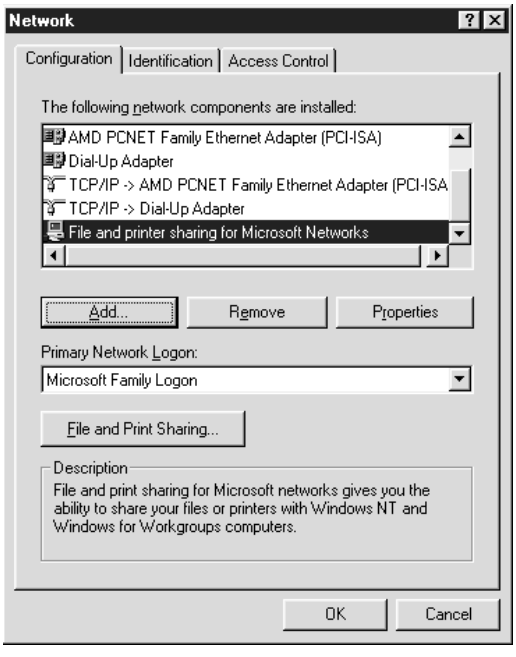


Figure 13-16 File and Printer Sharing installed

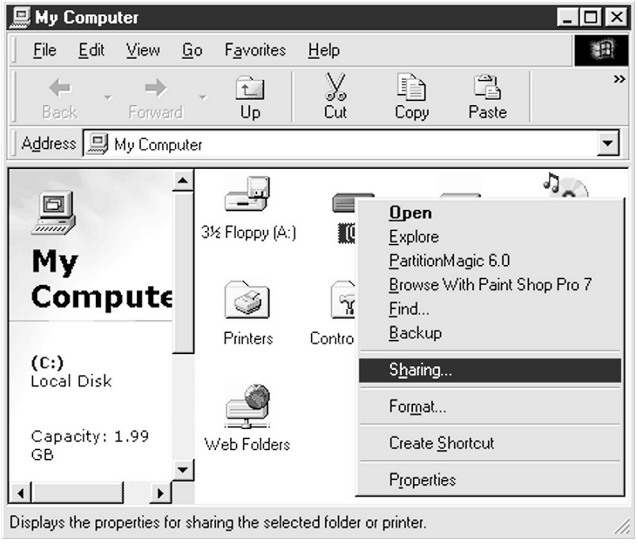


Figure 13-17
Selecting Sharing in My Computer

Windows 9*x* lets you share folders and entire hard drives—whatever you're sharing, you configure it in the exact same way. Just use My Computer or Windows Explorer to select the resource you want to share, alternate-click the resource and select Sharing (see Figure 13-17).

If you don't see the Sharing menu option, you've either forgotten to add the File and Print Sharing service, or you didn't select the *I want to be able to give others access to my files* check box. Go ahead and select Sharing to see the dialog box in Figure 13-18.

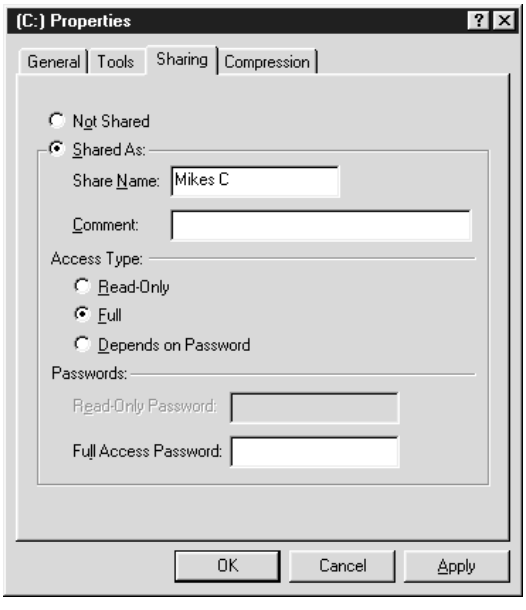


Figure 13-18
The Sharing tab

Remember seeing this dialog box earlier when we looked at how networks share? Well, this time, click the box labeled Shared As and enter a name. Because this is Windows 9*x* and you are using NetBIOS names, you would think the name could be up to 15 characters long, as you learned in Chapter 10, "Network Protocols." Yeah, well, you'd think that, but try it—you only get 12 characters, because Windows 9*x* systems are limited to the old DOS 8.3 filename size.

After you've given the share a name, you need to set the share permissions for this share. In most cases, you wouldn't be that interested in security (or you'd be using something besides Windows 9*x* for networking!), so just leave the permissions set to Full. Click OK and you will see the little hand icon appear that indicates a resource is shared (see Figure 13-19).

Windows NT and Windows 2000/2003

Remember what you just learned about sharing a folder or drive in Windows 9*x*? Well, it works pretty much the same way in Windows NT and Windows 2000/2003. You don't need to configure any version of NT or 2000/2003 to share—they are preconfigured to share by default—so you just need to worry about setting up the share. Just as you did with Windows 9*x*, select the drive or folder you want to share, alternate-click and select Sharing in NT or Sharing and Security in 2000/2003 to see the sharing dialog box. Figure 13-20 shows this box in Windows NT, while Figure 13-21 shows the same box in Windows 2000. They look basically the same.

If you don't see the Sharing menu option, it means you are not a member of the Administrators (NT/2000/2003) or the Power Users (2000/2003) group. A user account that is a member of the *Power Users* group has the capability to do many of the basic administrator functions; this is a handy way to give other users the ability to do things like share folders, without making them members of the all-powerful Administrators group.

Just like with Windows 9*x*, you must name the shared folder. Windows NT and 2000/2003 allow share names of up to 80 characters. Be aware, however, that any shares with names longer than 12 characters will not be visible to Windows 9*x* systems.

Even in Windows NT and 2000/2003, you still need to set share permissions before you can share a resource. Click the Sharing tab on the Properties dialog box of the shared resource. Because we're going to use NTFS permissions to do the actual security work, we don't need to do anything here at all. Just leave this at the default full-control settings; that's the normal process on NTFS systems.

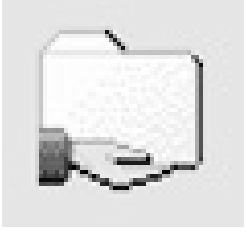


Figure 13-19
A shared folder showing the hand icon

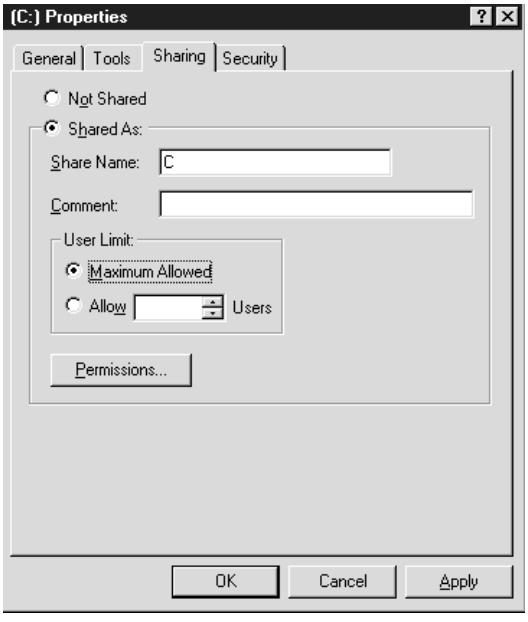


Figure 13-20
The Windows NT Sharing tab

Now let's have some fun and start playing with NTFS permissions! In Windows NT, click the Security tab, and then click the Permissions button (see Figure 13-22). In Windows 2000/2003, just click the Security tab to see the NTFS settings (see Figure 13-23).

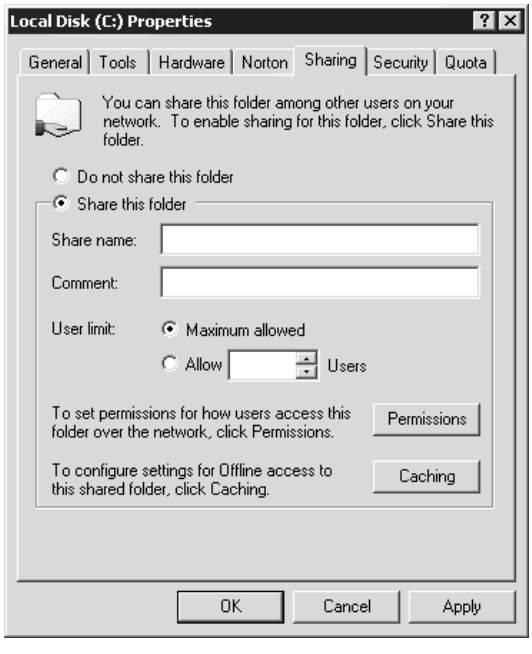


Figure 13-21
The Windows 2000 Sharing tab

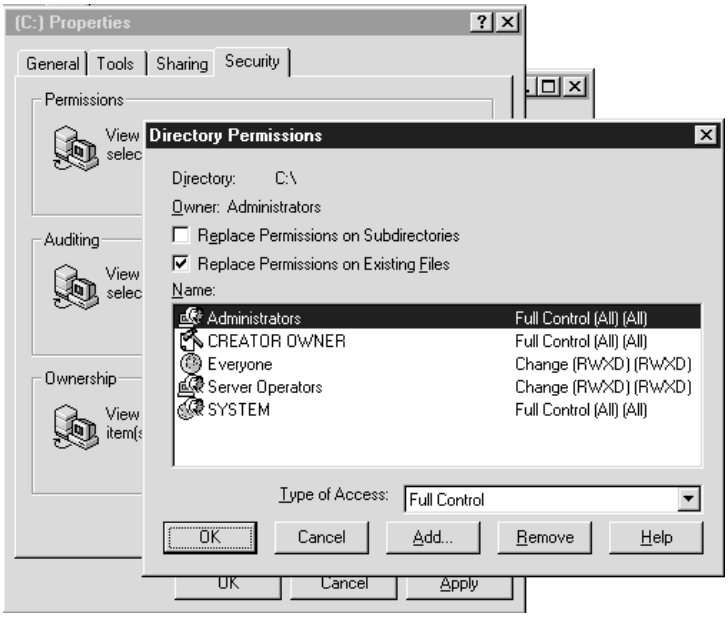
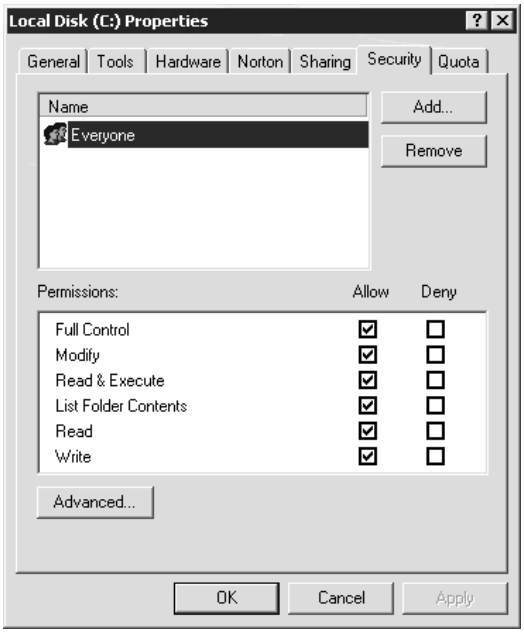


Figure 13-22 Windows NT permissions

Figure 13-23 Windows 2000 NTFS settings



By default, everyone has complete access to a new share in Windows NT/2000/2003. All network operating systems start a new share with some default permissions applied to everyone, so your first job is to start limiting who gets access. Let's concentrate on Windows 2000/2003 for a moment because Windows NT does all this in roughly the same fashion. Let's say we only want the Accounting group to be able to read documents, and we want Mike Meyers to have full control. Microsoft does a nice job of making this easy to set up. Start by clicking Add to see a list of users and groups. Find the Mike Meyers user account in the Accounting group in the list (see Figure 13-24). Pretty much all network operating systems let you add multiple users and/or groups in one shot.

Click OK to return to the Security tab. You'll see Mike Meyers is now listed. Take a look at the default permissions. Like most other network operating systems, Windows 2000/2003 provides only Read & Execute, List Folder Contents, and Read permissions by default. We'll need to click Full Control to let Mike Meyers do whatever he wants. The Accounting group's default settings are just fine for what we want, so we'll leave them alone.

Oops! You can't leave yet! Remember the Everyone group? It has full control! As long as that's the case, everyone has full access—better change access to the defaults (Read & Execute, List Folder Contents, and Read) for the Everyone group before you exit (see Figure 13-25).

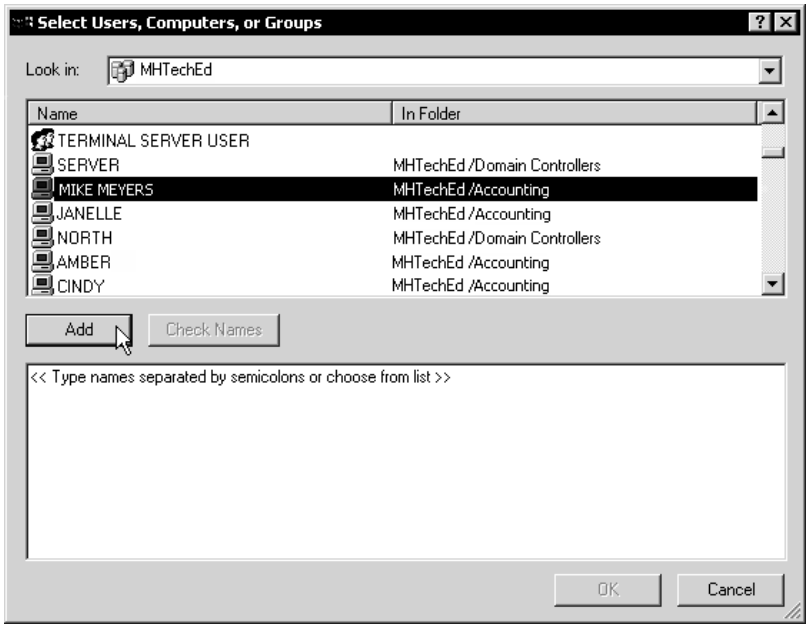


Figure 13-24 Finding Mike Meyers and Accounting in the list

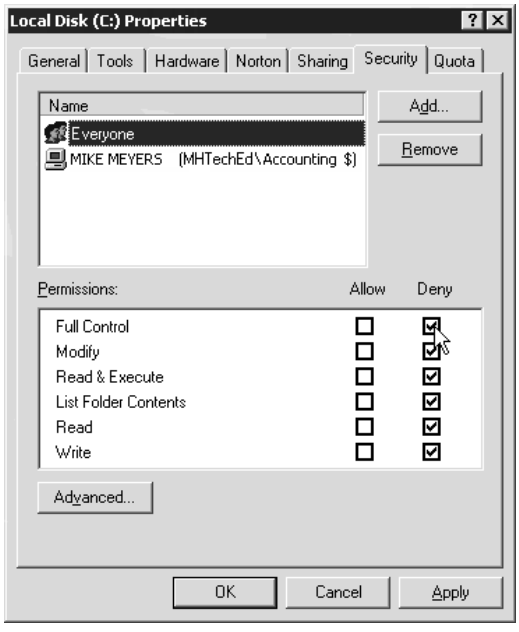


Figure 13-25 The Everyone group is denied access.

Hey, this brings up an interesting issue: What if a user account is a member of two different groups, and these two groups have different permissions for the same folder? Or what if a user account has certain permissions for a folder—what permissions do they have for any subfolders and any files in those subfolders? All network operating systems have different ways of handling these more complex permissions issues—you could easily make a career out of being little more than a permissions expert. Luckily, the Network+ exam isn't too interested in more than a basic understanding of the existence of permissions, so you can blissfully ignore these fascinating questions until you decide to go for your more advanced certifications.

Windows XP

As you might guess from the Windows XP permissions discussion earlier in the chapter, Windows XP does sharing in a couple of ways, depending on whether you use Home or Professional, and whether you choose to disable simple file sharing in the latter. To share a folder, simply alternate-click and select Sharing and Security. If you have Windows XP Home or Professional running in a workgroup with simple file sharing enabled, you get the Properties window for that folder, open at the Sharing tab, as in Figure 13-26. Select the Share this folder on the network option, and then assign a share name. Like other later versions of Windows, this name can be up to 80 characters and include spaces. If you want users to be able to change files within that folder, select the Allow Network Users To Change My Files option.

Figure 13-26
Simple file sharing in action

With simple file sharing disabled or if you run Windows XP Professional in a domain, you can share a folder or other resource precisely as you would in Windows 2000 and Windows Server 2003.

NetWare 3.*x*

The first thing to appreciate about Novell networks is that NetWare servers do not use the classic drive letters you see on Windows systems. When you look at a NetWare server, you see drive volumes with names like SYS: and VOL:. The SYS: volume is roughly equivalent to the Windows C: drive—by default, the SYS: volume stores all of the critical programs that make up the NetWare NOS itself.

You never sit down and work directly at a NetWare server, one of the features that distinguishes NetWare servers rather dramatically from Windows servers! You instead use client machines to access the server and do administrative work remotely. NetWare comes with a series of utilities that you run remotely on the server to perform almost every network task, including creating users/groups and setting rights to shared folders. These utilities are located by default in a special folder called \public on the SYS: drive on the server but most NetWare administrators will move this folder to a more secure area.

How can users access these programs if they can't see volumes with weird names like SYS:? The answer lies in special mapping that is automatically done for any system that needs to access a NetWare server. Every NetWare client has a special drive pre-mapped to a drive letter—in the case of Windows systems, this drive is usually called the F: drive, but that can easily be changed. When a Windows PC loaded with the correct NetWare

client software boots up, this F: drive is automatically mapped, whether or not the client is logged into the server. Figure 13-27 shows an example of this mapped drive on a Windows system. Using this mapped drive, the client system can run utilities without even logging into the network. The mapped drive also acts as a public folder where certain files and utilities are made available to all, regardless of the level of rights that client has to other areas of the server.

There is no specific step you must perform to start sharing folders in NetWare. All folders on all drives are ready for sharing—you only need to set up the trustee rights. The term *trustee rights* is NetWare lingo for user and group permissions to a shared folder. Any user or group with rights to a certain shared folder is said to have trustee rights to that folder. Don't let these terms throw you—it's the same as setting permissions on a Windows NT, 2000, 2003, or XP system!

We set up trustee rights in NetWare 3.*x* by running the ancient, but completely functional, *SYSCON* program. There are other methods, but SYSCON is the most famous and most common way to set up trustee rights in NetWare 3.*x*. SYSCON does far more than just make shares available—this same program handles a number of administrative tasks, such as creating users and groups. SYSCON is a text mode utility that runs at a command prompt—a testament to the DOS era of networking. While SYSCON is powerful, it is also an absolute pain to use, and more than a little practice is required to get it to work properly. Figure 13-28 shows SYSCON being used to set up trustee rights to a folder. Note the names of the two groups and the rights assigned to them. You can't tell which is the trustee, right? That's okay—it's RWCEMF, on the right side. Go back to the permissions section of this chapter and check the NetWare rights table to see which rights are assigned to the users in this example.

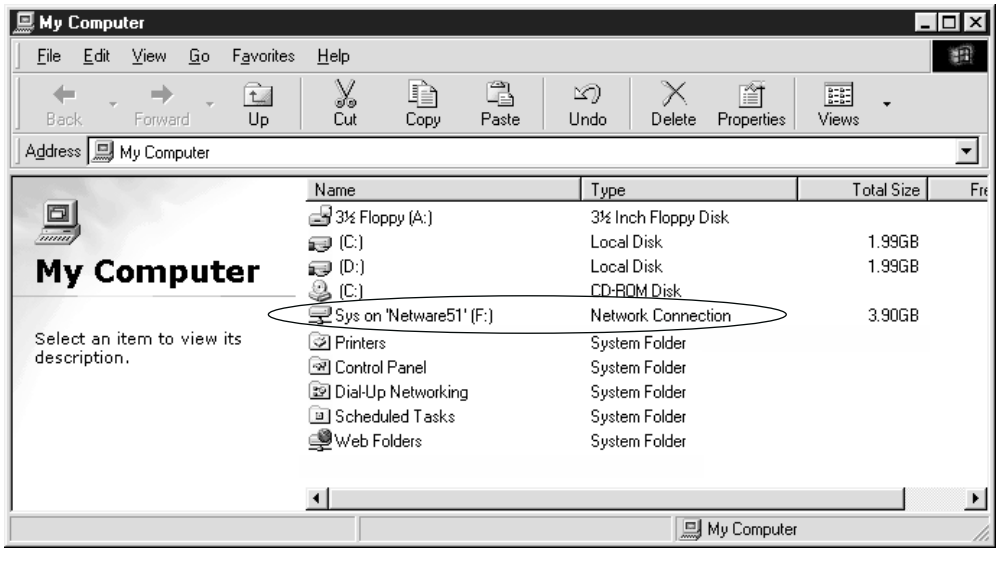


Figure 13-27 A mapped NetWare drive on a Windows system

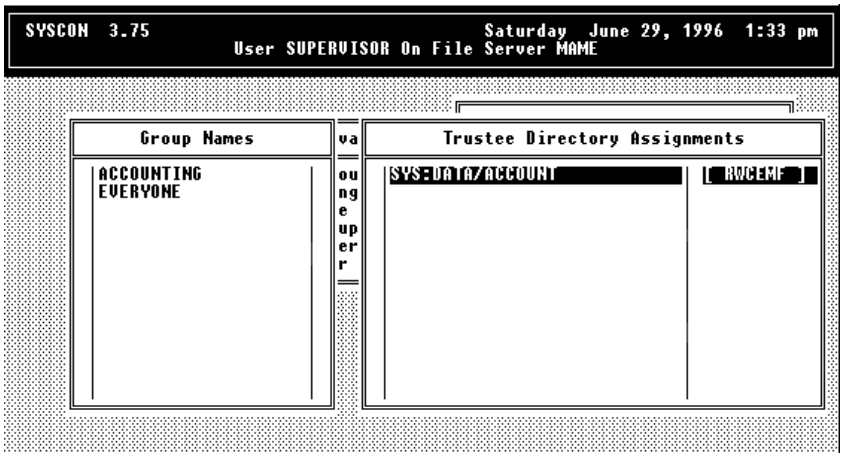


Figure 13-28 SYSCON in action

NetWare 4.*x*/5.*x*/6.*x*

NetWare 4.*x*/5.*x*/6.*x* work basically the same way as NetWare 3.*x*—but happily, the tools you have for assigning trustee rights have improved dramatically. The current tool we use to assign trustee rights is NWADMIN. Unlike the old SYSCON, *NWADMIN* is a Windows-based application that lets you click the folder you want to share and easily assign trustee rights. Figure 13-29 shows NWADMIN in action, configuring trustee rights for a shared folder.

Sharing Folders in UNIX/Linux

There are no standard graphical tools for sharing files in UNIX/Linux—each Linux distribution uses their own tools. UNIX/Linux systems do not have a sharing option that easily fits into the paradigm of Windows and NetWare systems. UNIX/Linux systems share files across a network in a variety of ways. These include File Transfer Protocol (FTP), Network File System (NFS), and Samba. FTP, as discussed in Chapter 11, "TCP/IP," enables two TCP/IP hosts to transfer files across a network. All implementations of TCP/IP support FTP, making it an excellent choice for moving files from a UNIX host to a machine running another operating system such as Windows 9*x*, Windows NT, a different flavor of UNIX, or even a Macintosh.

Network File System (NFS) enables a UNIX system to treat files and directories on another UNIX host as though they were local files. Let's say Fred needs to access the /mark/projects/current directory on Mark's UNIX system, named MARK1. Fred mounts the /mark/projects/current/ directory to his own file system as /markstuff/, adding it to his local directory structure. As far as any program on Fred's UNIX machine can tell, the files in the /markstuff/ directory are local files. NFS enables his UNIX machine to share files transparently by adding network directories to its local directory structure. Unfortunately, Windows-based machines don't get to play, because they don't come with an

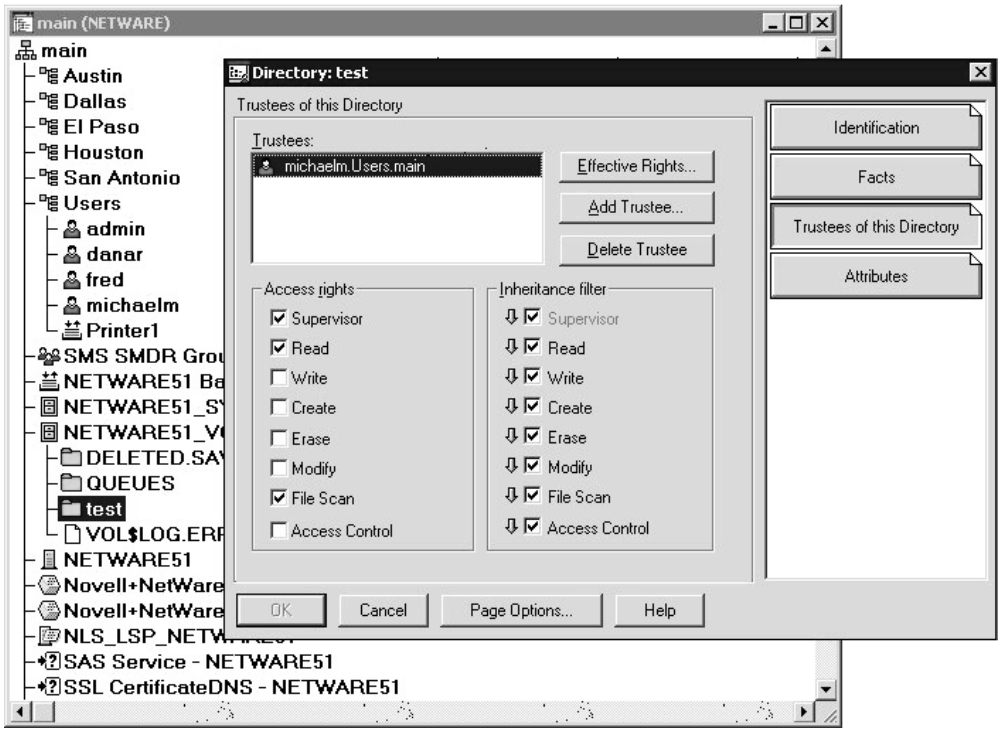


Figure 13-29 NWADMIN in action

NFS client. Although some fine third-party NFS tools are available for Windows, most of us just use FTP or Samba for Windows-to-UNIX/Linux file transfers.

UNIX systems, however, can also pretend to be Microsoft clients and servers using Samba, which enables UNIX systems to communicate using Server Message Blocks (SMBs). To a Windows-based system running Client for Microsoft Networks, a UNIX system running Samba looks just like a Microsoft server (see Figure 13-30). We'll see more of Samba, NFS, and FTP in later chapters.

Sharing Folders in Macintosh

Pre-OS X Macintosh systems have rudimentary networking functions, similar to Windows 9*x* networking. Unlike a Windows 9*x* system, a Macintosh is ready to share folders immediately. To share a folder on a Macintosh, you select the folder, click File/Get Info, and click the *Share this item and its contents* check box. Like Windows 9*x*, you only have three share permissions, which Apple calls Read & Write, Read Only, and interestingly enough, Write Only (see Figure 13-31). A shared folder manifests itself with a different icon, as shown in Figure 13-32. Unfortunately, these shares are only good for Mac-to-Mac communication—we'll see how to get Mac to talk to Windows clients in Chapter 18, "Interconnecting Network Operating Systems."

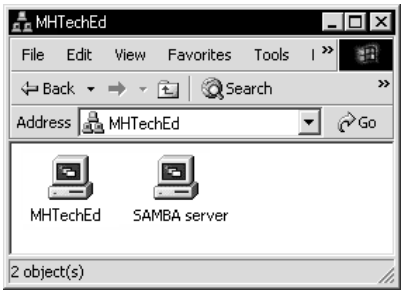


Figure 13-30 A UNIX system running Samba looks just like a Microsoft server.

TIP The introduction of OS X has fundamentally changed the way Macs share files and folders. With OS X, the sharing functions are now basically identical to UNIX/Linux.

Follow the Steps

Regardless of the NOS, the steps you take to share a folder are basically the same. First, you make sure the system is capable of sharing—this is done for you in all but Windows 9*x* systems. Second, you decide what you want to share, and make that folder available for sharing. Finally, you set whatever share/permissions/rights you want the share to have. Remember these three steps and sharing a folder is always easy!

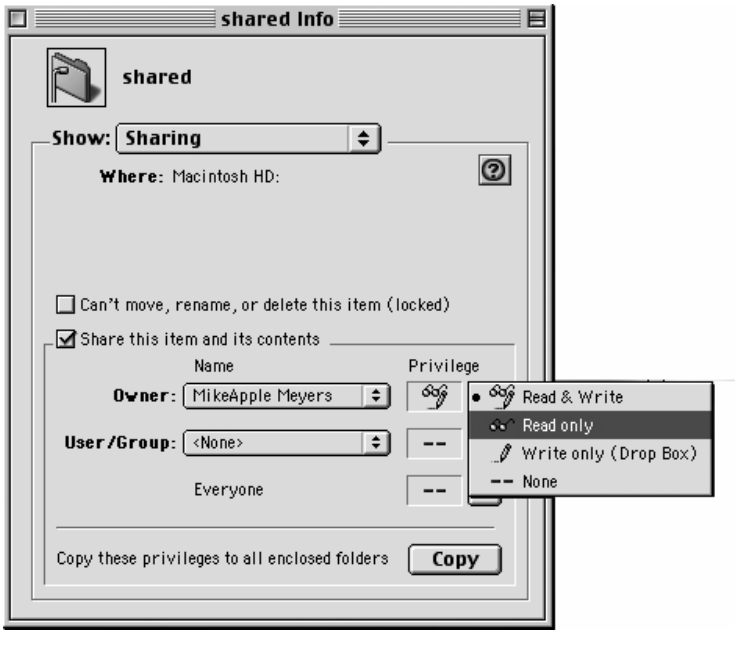


Figure 13-31 Sharing a folder on a Macintosh

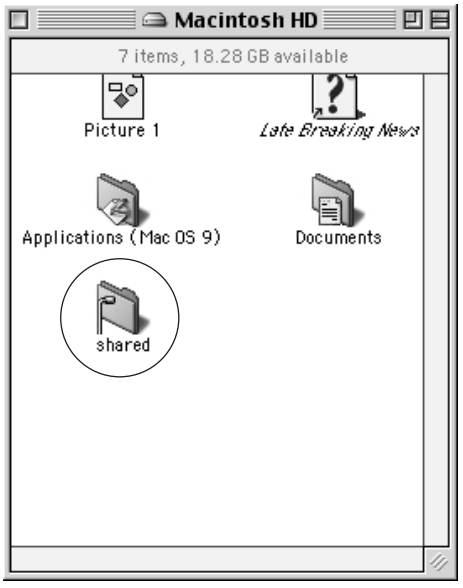


Figure 13-32 A shared folder on a Macintosh

Sharing Printers

The process of sharing printers is similar to the folder sharing process—you must make sure the system is capable of sharing a printer, give the printer a share name, and set permissions. The actual process by which network operating systems share printers varies dramatically, although the sharing process doesn't vary nearly as much between versions of Windows and NetWare. This means we won't have to go into quite the same level of detail we saw with folders. Let's see how they do it!

Sharing Printers in Windows 9*x*

Sharing a printer in Windows 9*x* requires almost exactly the same steps as sharing a folder in Windows 9*x*. First, make sure the sharing system has added the File and Print Sharing service, and that you have clicked the *I want to be able to allow others to print to my printer(s)* check box. Having done those steps, you share the printer by opening My Computer, finding the printer you want to share and—yup, that's right!—selecting Sharing and giving the printer a share name, as shown in Figure 13-33. Windows 9*x* has no form of permissions for shared printers, but it does at least allow you to set a password for the network share. Like all network shares, this password only affects network users.

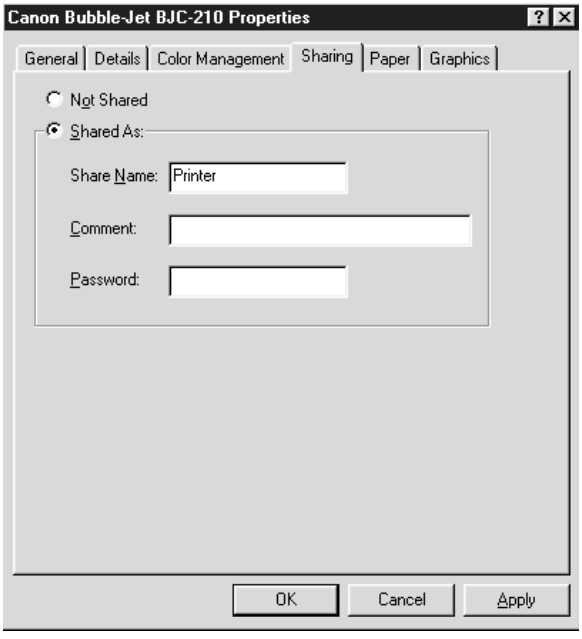


Figure 13-33
Sharing a printer in Windows 9*x*

Sharing Printers in Windows NT/2000/2003/XP

Windows NT/2000/2003/XP share a printer exactly like Windows 9*x*, but they do provide more substantial permissions. A Windows NT/20002003/XP system provides three levels of print permissions: Print, Manage Printer, and Manage Documents. The Print permission enables users and groups to print to the printer. Manage Printer lets users control the printer properties, and Manage Documents gives users the right to delete, pause, and restart print jobs. Like folder permissions, these settings are found on the Security tab of the printer's Properties dialog box. Figure 13-34 shows the Printer sharing Security tab in Windows 2000.

Sharing Printers in NetWare

Novell NetWare has a bit of a problem with printers. While Windows NT/2000 can allow any system to act as a printer server, Novell NetWare can only control printers installed on a NetWare Server. Unfortunately, printers don't tend to hang around servers—they're installed around the network at user systems or as stand-alone network printers. Novell realized this long ago and developed complex, but powerful, methods for print serving; these print methods break away from the only-servers-serve attitude, instead allowing any system on any NetWare Network to act as a print server. All versions of Novell NetWare share basically the same two methods. The first is to allow a NetWare server to act as a print server. The second is to configure a client system to act as a print server. Novell allows virtually any type of OS client to act as a NetWare print server. This includes all versions of Windows, UNIX/Linux, and Macintosh computers, although

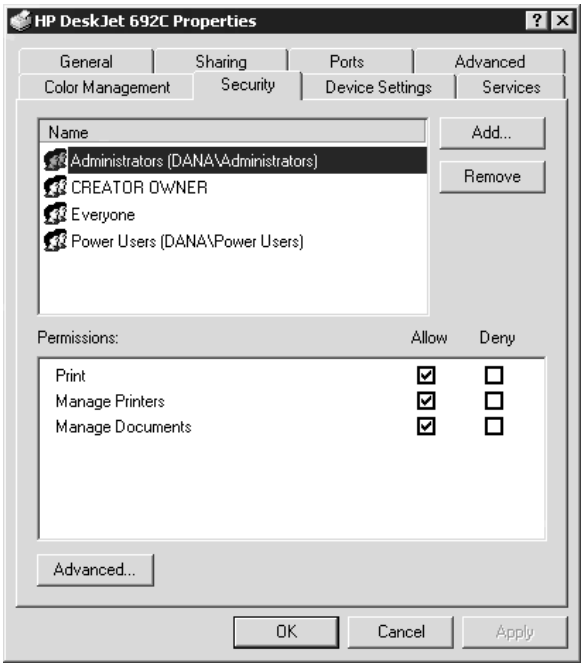


Figure 13-34 The Windows 2000 Printer Properties Security tab

you will need to install special NetWare printer server software on them to enable this to happen.

Sharing Printers in UNIX/Linux

Once again, UNIX/Linux does not have the same concept of actively sharing a printer that we saw in both Windows and NetWare. Instead, Linux systems typically use one of two methods: Samba or LPD/LPR. LPD/LPR consists of two TCP/IP functions: *Line Printer Daemon (LPD)* and *Line Printer Remote (LPR)*. The LPD program works as the server and runs on the system sharing the printer. Meanwhile, LPR runs on any system wanting to access a printer under the control of LPD. As a matter of fact, almost every operating system capable of supporting TCP/IP also includes the LPD and LPR programs, or at least something similar enough to support them. Go to a command prompt in Windows, type **lpr**, and press ENTER—it's almost certainly there!

Accessing Shared Resources

Once a folder or printer has been shared by a serving system, the next step is for the client systems to access that device and start to use it. The steps involved in accessing a shared resource usually include browsing to locate the shared resource, and then connecting to it to make it seem as though it were a local resource; neither of these steps is completely necessary in all situations, however.

In this section, we will concentrate exclusively on Windows client systems. That's about as much as the Network+ test wants to you to know. We'll save most of the UNIX/Linux connection issues for other chapters.

Accessing Files in Windows

You can access a shared resource in Windows in literally about six different ways, but the most common method is to browse through My Network Places to locate the shared resource you desire. Tim wants to store some files in a folder on the server. He talks to the person who shared the folder on the server, who tells him to use the timstuff share on the server. Tim uses My Network Places to locate the share, as shown in Figure 13-35. Once he has found the share, Tim has some choices. He can just leave the share open in My Network Places and use it like any other folder, but this has a downside: he'll have to do this every time he uses the share. Being a clever fellow, Tim instead decides to map the shared folder and give it a drive letter, checking the box that orders the share to reconnect at logon. We call this a *persistent connection* (see Figure 13-36). Any time you map a drive, only to have it disappear after a reboot, you can be pretty sure you did not make a persistent connection.

Keep in mind that Windows doesn't care what type of server is providing this share. As long as you have the right user account with the right permissions, you'll be able to treat a share the same way, whether it comes from a Windows NT Server system, a Linux box, or a NetWare server. All shared folders manifest the same way (see Figure 13-37).

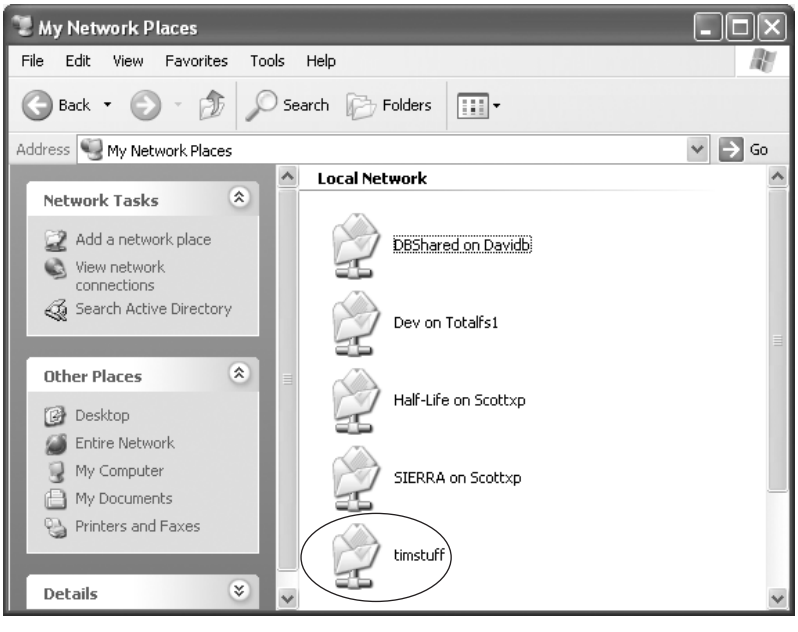


Figure 13-35 Finding the share in My Network Places

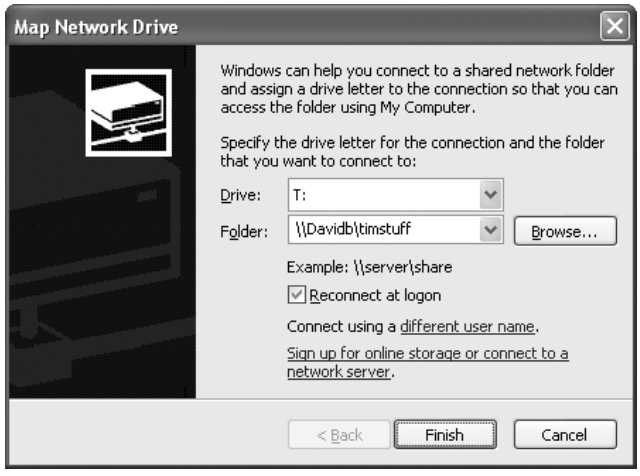


Figure 13-36 Setting a persistent connection

Beginning with Windows 95, you could create a desktop shortcut to a network share as an alternative to mapping the share to a drive letter. Just right-click and drag the shared folder to the desktop to create a shortcut. Windows 2000 added the Network Place concept. Basically just a shortcut, a Network Place points to a shared folder, but it is not limited just to shared folders—you can make a web site, an ftp site, almost anything you can share, a Network Place. The usual way to create these shortcuts is to open the My Network Places folder and select Add Network Place (see Figure 13-38).

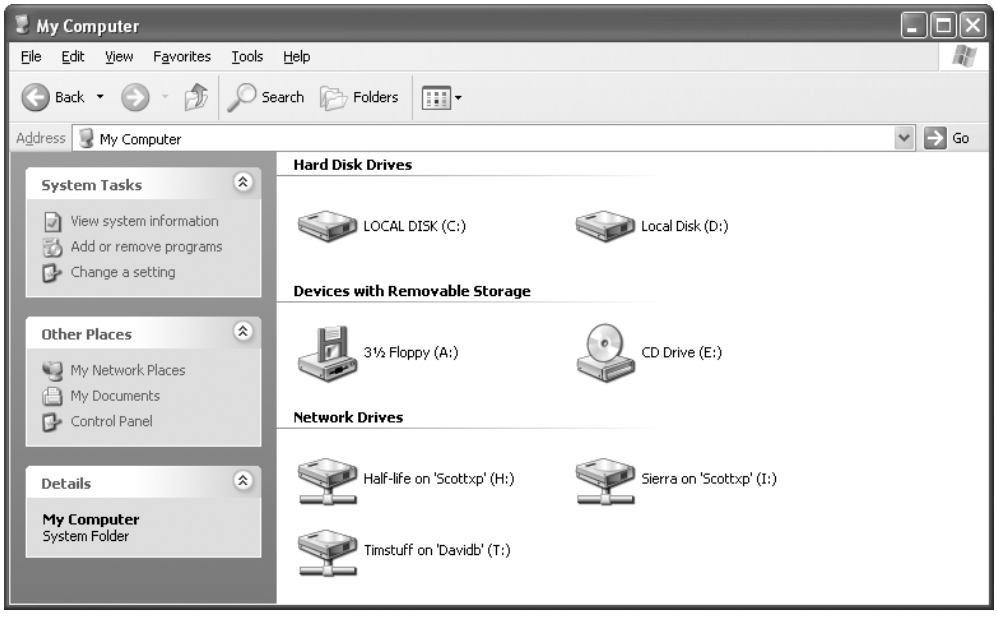


Figure 13-37 All shared folders look alike.

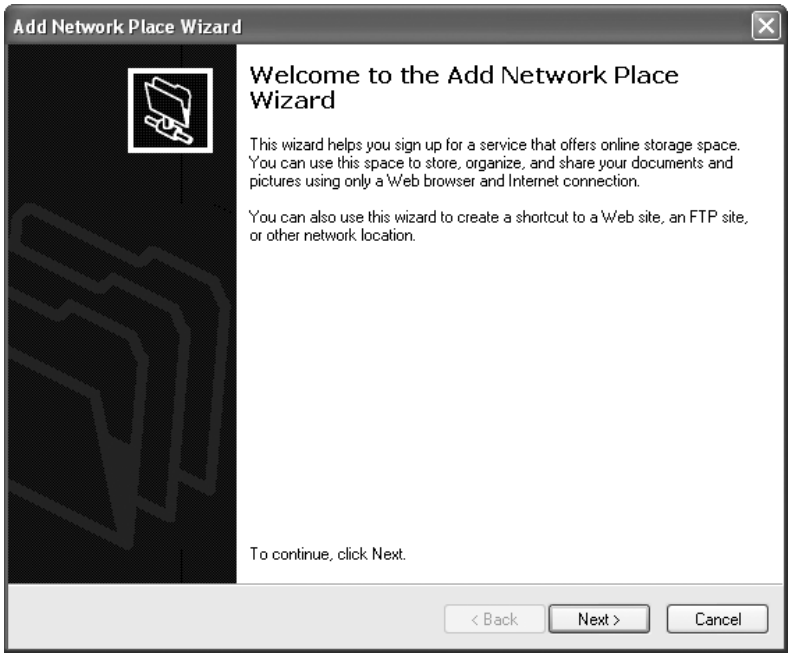


Figure 13-38 The Add Network Place Wizard in My Network Places

Accessing Shared Printers in Windows

One aspect of printers not shared by folders is that your system needs printer drivers to send print jobs. Back in the old days, we would capture the printer to an LPT port, and then install the printer drivers onto our local systems. We would then tell the printer to install to the captured port. Today, Windows makes all of this much easier—when we access a network printer, the printer drivers install automatically on the local system, a big benefit of Windows networking! NetWare has a similar feature.

Troubleshooting Shared Resources

Almost all problems with sharing or accessing shared resources stem from some mistake in the process of creating the share, as opposed to a problem with the shared resource itself. In fact, most of what appear to be sharing errors have nothing to do with the sharing process—they are lower-level errors like severed cables, incorrectly installed protocols, or attempts to access the wrong system. For the moment, let's assume none of these is the culprit, and look at some of the classic sharing errors that do take place on a network. We'll divide the most classic errors into two groups: sharing errors and access errors.

Sharing Errors

Sharing errors are problems that take place as you try to share a resource. Whenever I have a sharing error, I make a point to mentally review the steps required to create a share. Usually I realize that I skipped a step or failed to do a step properly.

The most frequent mistake people make is not sharing the right resource. This isn't exactly a sharing error, but it happens so often that I simply must mention it. One folder that folks like to share on a Windows system is the Desktop. That makes sense—I love to dump junk on my Desktop, and then tell others to "Get it off my Desktop!" The problem with Windows Desktops is that many people don't know where they are located. Do you remember? On a Windows 9*x* system, you'll find the Desktop in \Windows\Desktop. Windows NT and Windows 2000 make life more difficult, because they create a separate Desktop for each user. In Windows NT, each user's Desktop is hidden under the \WINNT\PROFILES folder. To find the Desktop of my Windows 2000 system, you have to dig down to \Documents and Settings\michaelm\Desktop. Desktops are not the only folders people have a problem finding to share. You can mess up sharing a folder in zillions of ways, so the wise user will double-check each folder before they start sharing it!

The other sharing issue that will bust you on more advanced network operating systems like Windows NT, 2000, and NetWare stems from permissions. The complexities of permissions make it way too easy to give someone insufficient permissions, preventing them from doing what they need to do on that share. Getting this right requires a bit of patience on your part as you experiment with different permission combinations to find the one that will let the user do what they need to do without unnecessarily sacrificing security.

Finally, watch out for share name incompatibilities. You can make a share in Windows XP called "This is the share Mike made on 12-1-04. Please use freely, but send me an e-mail when you do!" but many other systems, especially Windows 9*x* systems, won't be able to see it. Always think about the other systems on your network before you create share names!

Access Errors

As with sharing errors, almost all *access errors* are due to configuration problems, not some corrupt piece of software. The single biggest error flows directly from permissions: if you can't access a shared resource in the way you think you should be able to, ask the person who controls the share to give you the permissions you need. It's not uncommon to hear a conversation like this:

"Hey Alison, I can't make any changes to the Accounting database!"

"Yeah, well, you're not supposed to be able to! You don't have the right permissions!"

"Okay, well, either I make these changes or the boss is gonna yell at me—can you change my permissions?"

"Okay, gimme a sec."

"Thanks!"

I know some folks in more formal offices will laugh at this, because they have rigid procedures for changing permissions, but the basic process is still the same. Check to be sure you have the permissions you need. The fact that you had the right permissions yesterday is no guarantee that some network guru isn't going to change them today. Always assume permission problems first!

Chapter Review

Questions

1. Which of the following operating systems can use the file and folder permissions based on NTFS?
   - **A.** NetWare
   - **B.** Windows 2000
   - **C.** Linux
   - **D.** Windows 98
2. Your network consists of a NetWare server and a mixture of Windows 98 systems and Windows NT workstations. You add a Windows NT server to the network. All systems have the TCP/IP protocol suite installed. Samantha is unable to access a shared file on the NT server, but she is able to print from the shared printer. What could be the problem?
   - **A.** Samantha's system and the Windows NT server are not connected to the same hub.
   - **B.** Client for NetWare Networks has not been installed on Samantha's system.
   - **C.** Client for Microsoft Networks has not been installed on Samantha's system.
   - **D.** You have not given Samantha permission to access the shared file on the NT server.
3. Chris needs to work on a folder on the Desktop of your Windows 98 system. He is unable to access the folder, and you realize that you haven't shared the folder. You alternate-click the folder, but you don't get the Sharing option. What has happened?c
   - **A.** Chris hasn't been given the correct permissions to access the folder on your Desktop.
   - **B.** At the logon screen, you clicked Cancel instead of entering a password.
   - **C.** You did not install File and Printer Sharing services on your system.
   - **D.** The server is down.
4. The office's expensive laser printer is connected to Karen's Windows 98 system. Previously, other users have been able to print from that printer with no problem, but today they can't access it. What could be the problem?
   - **A.** At the logon screen, Karen clicked Cancel instead of entering a password to log onto the network.
   - **B.** File and Print Sharing services have not been installed on Karen's system.

C. The server is down.

D. Karen's system doesn't have the proper permissions set so other users can use the printer.

5. James is running a Windows 2000 system. He has shared his C: drive, but no one is able to access it. What could be causing this problem?

A. He needs to be the Administrator before he can share the drive.

B. He is set up as a Power User, and Power Users can't set permissions.

C. After he shares a device, he still needs to go into Security and set its permissions.

D. Everyone else's system has a problem. Only James' system is set up correctly.

6. Which of the following client systems can act as a printer server on a Novell NetWare network? (Select all that apply.)

A. UNIX/Linux

B. Macintosh

C. Windows 9*x* and 2000

D. NetWare Server

7. For client systems to act as a printer server on a Novell NetWare network, each of the client systems must have the NetWare printer server software installed.

A. True

B. False

8. To share printers on a UNIX/Linux network, which two printing services must be installed?

A. EPP

B. PPT

C. LPD

D. LPR

9. When we access a network printer, we don't need printer drivers.

A. True

B. False

10. Joe does a lot of work in a specific network folder, so he decided to map the folder as a network drive. The next morning when he booted up his system, the mapped network drive wasn't there. What happened?

A. You must map a network drive each day.

B. He didn't have the correct permissions to map a network drive.

C. The server was turned off, so the mapped network drive didn't appear.

D. He forgot to check the box to have the share reconnect at each logon.

Answers

1. **B.** Of the answers offered, only Windows 2000 uses NT File System (NTFS).
2. **D.** Samantha does not have permission to access the shared file on the NT server.
3. **C.** You did not install File and Printer Sharing services on your system.
4. **A.** Karen bypassed signing onto the network by clicking Cancel instead of entering a password at the logon screen. Windows 98 doesn't have permissions. Previously, the users have been able to use the printer, so the File and Print Sharing service is installed.
5. **C.** After a resource is shared in 2000, you still need to go to the Security tab and set permissions before others can access the resource. James must already be signed on as an Administrator or Power User because he was able to share the resource. A Power User can share resources.
6. **A, B, C, D.** All four answers are correct. With NetWare, each of these choices can act as a print server.
7. **A.** True. For client systems to act as a printer server on a Novell NetWare network, each of the client systems must have NetWare printer server software installed.
8. **C, D.** The LPD program works on the server and runs on the systems sharing the printer. LPR runs on any system wanting to access a printer under the control of LPD.
9. **B.** False. When you access a printer, you still need printer drivers, but Windows takes care of this automatically, so you don't have to worry about it.
10. **D.** Joe should have checked the box to have the mapped network drive reconnect at logon. If the server had been down, the mapped drive would have had a big red *X* across it.