Chapter 6
Connecting Windows Workstations

Because this is a book about using Linux on a Microsoft Windows-based network, this chapter shows you how to connect various Microsoft Windows workstations to a Linux-based PDC. You've already configured a Linux-based PDC and member server in Chapters 3 and 4. In this chapter I'll show you how to connect Windows 9x/NT-style workstations to those computers in a Microsoft-style Workgroup or Domain.

If you're setting up workstations on a Domain, the first step is to recheck the computer serving as the Linux PDC. On that computer, you want to make sure that computer accounts, logon scripts, and profiles are ready to connect. You'll also want to record some basic network settings from the PDC to help you configure your Windows workstations.

This book is directed toward administrators of Microsoft Windows networks. If this describes you, just about everything in this chapter should seem familiar. A number of the instructions in this chapter will in fact seem elementary. However, Linux administrators also will read this book, and will therefore need extra help with handling Microsoft Windows workstations.

In this chapter, I'll show you how to set the network properties for each Microsoft Windows workstation. When you reboot, you can then connect and log in to the Domain through the Linux PDC. You can also set up roaming profiles if desired. I'm assuming that each workstation has a standard network card that has already been detected by your Microsoft Windows workstation.

If you've administered a Microsoft network before, many of the techniques in this chapter should be familiar to you. While most of this book is geared toward the experienced Microsoft administrator, this chapter is focused more toward the Linux administrator who is less familiar with Microsoft workstations.

Whatever operating system you use, you can observe and troubleshoot network communication on this Domain though Linux log files in the /var/log/samba directory.

Preparing accounts

This section is based on the work you did in Chapters 3 and 4 to configure a PDC on a Linux computer. Windows NT/2000/XP workstations (like Linux workstations) can't connect to a PDC unless they have a computer account on the PDC. Every user who is connecting to a Domain also needs a user account on that PDC.

As described in Chapter 4, computer accounts are stored in the /etc/passwd file, and are made available to Microsoft Windows networks through /etc/samba/smbpasswd. Once your Windows computer has connected to the Domain, you'll see accounts for that computer in these two files.

If you haven't yet connected your Windows workstation computer to the Domain, the **add user script** command in the smb.conf configuration file should help. By design, it adds the computer account to these files when you connect your workstation to the Domain. Alternatively, you can add the computer account manually. For more information, see Chapter 4, "Setting Up Your File Server's Users."

Logon scripts

Logon scripts are commonly used to automatically connect users to shared directories and printers. They should be located in the directory defined in the smb.conf [netlogon] share. They'll work for all Windows workstations as long as they're saved as MS-DOS text files with a BAT extension. You can do this by saving files in text format with Microsoft Windows WordPad, not Notepad.

Microsoft Windows WordPad adds a return character to files saved in text format that Notepad and Linux text editors, such as gedit, do not.

Thus, to configure Windows logon scripts, you'll need both a PDC and a Microsoft Windows computer on a network. Installing a PDC on a Linux computer requires Samba. In this case, I'll be setting up a script for a user named **pilot**. Once you have these computers available, take the following steps:

1. On the Linux computer, log in as the root user. Open the Samba configuration file, /etc/samba/smb.conf. Make sure to activate the [netlogon] share. I'm assuming that you're setting up the share as described in Chapter 4:

```
[netlogon]
  comment = Network Logon Service
  path = /home/netlogon
  guest ok = no
  writable = yes
```

2. In the Samba configuration file, activate the appropriate **login script** command. I'm assuming that you're activating scripts by user name, which corresponds to the following command:

```
login script = %U.bat
```

3. Reload the Samba configuration with the **/sbin/service smb reload** command. This makes Samba read the new smb.conf configuration file.
4. On a Microsoft Windows computer, log in as the administrative or root user on the Domain.
5. Open Microsoft Windows WordPad by clicking Start | Run and then typing **wordpad** in the text box that appears. Then press Enter.

6. In Microsoft Windows WordPad, enter the commands that you want for the user named **pilot**. For example, I add the following command to mount the [tmp] share from the **nopaws** computer on the L: drive:

```
net use L: \\nopaws\tmp
```

7. Once you've created your desired netlogon file, save the file in text format. For user **pilot**, you would save it as pilot.bat.
8. If you've logged in as the administrative or root user on the Domain, you should be able to save pilot.bat directly to the [netlogon] share on the PDC.
9. Return to the PDC. Restart Samba with the following command:

```
/sbin/service smb restart
```

While all users will have to log in to the Domain again, that's required before users can access a netlogon share.

Profiles

As described in Chapter 3, you can configure roaming profiles for Microsoft Windows workstations. Profiles for Windows 9x/ME computers are different from Windows NT/2000/XP computers and are stored in different locations. While Windows 9x/ME profiles are stored in users' home directories, Windows NT/2000/XP profiles are stored as defined by the [Profiles] share and **logon path** variable defined in your smb.conf file.

If you're converting from a Microsoft-based PDC to a Linux-based PDC as defined in Chapter 4, you can copy roaming profile files to the corresponding home directories. You can copy Windows 9x/ME profiles directly to users' home directories. For user **mj**, that's the /home/mj directory on the Linux PDC.

One of the drawbacks of roaming profiles for Windows 9x/ME computers on a Linux PDC is that users can accidentally delete their own profiles on their home directories.

For Windows NT/2000/XP profiles, there are two variables. In Chapter 4, we defined the **path** variable in the [Profiles] share as /home/profiles. We defined the **logon path** variable as **\\%L\Profiles\%U**, which means the profile is stored in the /home/profiles/mj directory.

Configuring the Microsoft workstation

There are two basic types of Microsoft workstations. The first type is based on the 16-bit Microsoft operating systems: Windows 95, 98, and ME. The connections you make from these operating systems don't require a computer account on the PDC, because Windows 9x/ME computers are actually not full members of a Domain. This does not change whether the PDC is on a Windows or a Linux computer.

The other type of Microsoft workstation is based on the 32-bit Microsoft operating systems: Windows NT, 2000, and XP. These workstations are more flexible in terms of user names and passwords.

Other Microsoft operating systems are available. Because Linux with Samba is intended as a substitute for the Microsoft Windows server operating systems, I don't configure the Microsoft Server operating systems in this book. Older Microsoft workstations, such as those based on MS-DOS and Windows for Workgroups, are rarely in use and are therefore not covered in this book.

In the following sections, I'll examine how to connect each major Microsoft workstation to a Domain. While the techniques are basically the same for all three 16-bit Microsoft operating systems, I highlight some variations in each system. While I focus on connecting to a Domain, I address connections to a peer-to-peer Workgroup later in this chapter.

Connecting a Windows 95/98/ME workstation to a Domain

The methods you use to connect Windows 95/98/ME computers to a Domain are basically the same for all three operating systems. In the following sections, I'll illustrate how you can connect to a network, set up roaming profiles, connect to shared directories, and share with other computers in the network.

Windows 95 and encryption

If you're concerned about network security, you should use encryption at least on the most critical items such as passwords. With the following command, the standard Samba configuration assumes that you're using an operating system that encrypts passwords:

```
encrypt passwords = yes
```

Passwords sent from the latest version of the Windows 95 operating system, known as OSR2, are encrypted. If you have a Windows 95 workstation, it's easy to find its version. On the Windows 95 desktop, right-click the My Computer icon. This opens the System Properties dialog. On the General tab shown in **Figure 1**, you should see 4.00.950 B, which corresponds to OSR2.

***Figure 1**. Checking the version of Windows 95.*

Disabling encryption

If you want to use older Microsoft Windows 95 operating systems on your network, the only option is to disable encryption on the PDC and all other workstations on the network. Samba includes Microsoft Windows registry files that you can use on various Microsoft workstations. On Red Hat Linux 9, the files described in **Table 1** are located in the /usr/share/doc/samba-2.2.7a/docs/Registry directory.

If you've upgraded to a later version of Samba 2.2, the directory name will change accordingly. If you still can't find these files, the following command should help:

```
rpm -ql samba | grep .reg
```

***Table 1**. Registry files for disabling encryption.*

File	Operating System
Win95_PlainPassword.reg	Windows 95 OSR2 (4.00.950 B)
Win98_PlainPassword.reg	Windows 98 (all versions)
WinME_PlainPassword.reg	Windows ME
NT_PlainPassword.reg	Windows NT Workstation
Win2000_PlainPassword.reg	Windows 2000 Professional/Windows XP Professional
WinXP_SignOrSeal.reg	Windows XP Professional for joining a Domain with a Linux PDC

To apply the registry file to a particular operating system, copy it to a temporary directory on the target workstation. When you double-click it through a medium such as Windows Explorer, the contents of the REG file are automatically merged with the Windows operating system registry.

Be careful when applying registry files to Microsoft Windows. Any REG file that you run takes effect immediately. I recommend that you back up the Windows registry file before applying the Samba revision.

Remember, if you want to use clear text passwords on a network with Windows 95 OSR2 computers (or later), you'll need to disable encrypted passwords on each workstation with the files as noted in Table 1.

There are two basic steps associated with setting up a Windows 9x/ME computer on a network. First you need to configure networking on your computer. Then you can configure a network connection to a Workgroup or Domain.

Configuring a connection to a network

To configure Windows 95/98/ME on a network, have your Windows 95/98/ME CD ready, and then take the following steps:

1. Right-click the Network Neighborhood icon on the desktop to open the Network dialog.

2. On the Configuration tab shown in **Figure 2**, make sure that you have installed at least the noted components, similar to what's shown in the associated text box.

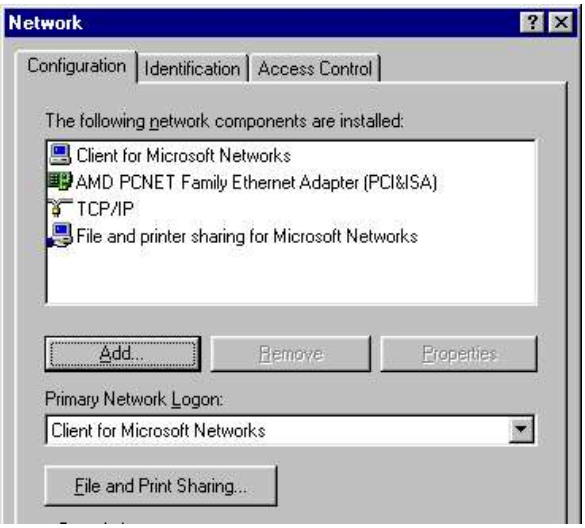

***Figure 2**. Windows Network Configuration.*

- **Client**: Client for Microsoft Networks supports login connections to a Domain.
- **Adapter**: A network adapter is required for a connection to a network; Windows normally detects network adapters upon installation. Hardware installation details on a Microsoft computer are beyond the scope of this book.
- **Protocol**: TCP/IP networking accommodates the default Linux network. If you have more than one TCP/IP protocol entry, make a note of the one associated with your network adapter.

This book assumes that you're using TCP/IP on your network, because that is the default for Linux and most Microsoft operating systems. If it isn't installed, click Add in the Network dialog shown in Figure 2.

- **Service**: "File and printer sharing for Microsoft Networks" allows you to share directories and printers from this computer.

3. Make sure the Primary Network Logon is set to Client for Microsoft Networks.
4. If any of these components are missing, click Add. In the Select Network Component Type dialog, select a missing component (Client, Adapter, Protocol, or Service), click Add, and then follow the prompts. Repeat as needed.

Configuring a connection to a Domain

Now I'll show you how to connect your configured Windows 9x/ME computer to a Domain. The following steps are essentially a continuation from the previous section, as I assume you still have the Windows Network configuration dialog open on your computer.

1. Click the File and Print Sharing button if you want to share directories or printers from this computer.
2. Highlight Client for Microsoft Networks and click Properties, which opens the window shown in **Figure 3**. If you're connecting this computer to a Domain, select the "Log on to Windows NT domain" check box, and then enter the name of the Domain governed by the Linux PDC. The network logon options allow you to reconnect automatically to any directories that you've shared before. Click OK when you've made your choices.

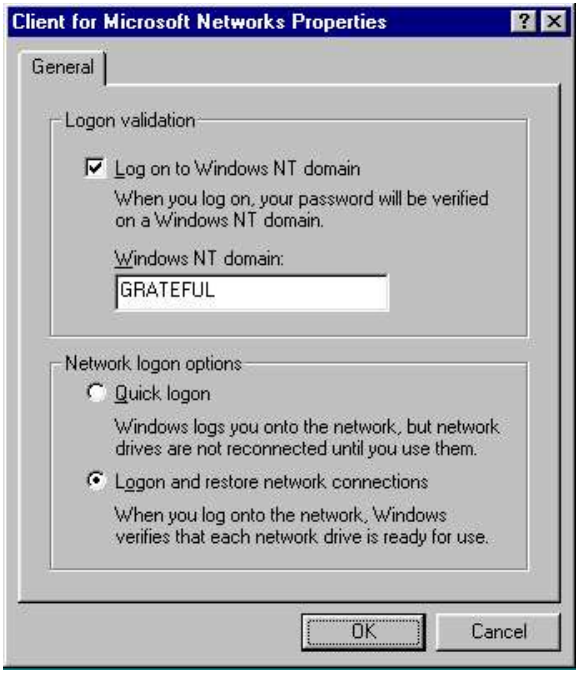

***Figure 3**. Setting up a connection to a Domain.*

3. You'll also need to set up your IP address information. Back on the Configuration tab, highlight TCP/IP. If you have more than one network adapter, be sure to select the TCP/IP setting associated with your network card, and then click Properties (see **Figure 4**).

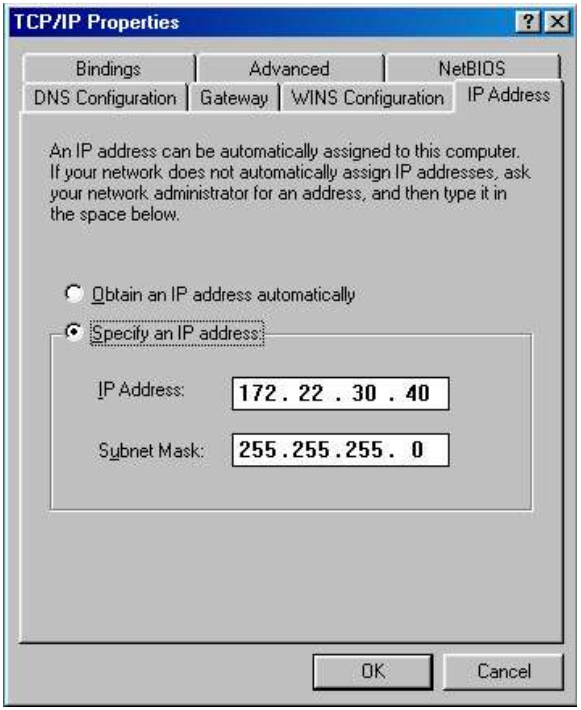

***Figure 4**. Setting a static IP address.*

4. If you aren't using a DHCP server, you'll need to add static IP address information. To do so, click the IP Address tab and then enter an IP address and subnet mask.
5. If you've configured a WINS server on the Linux PDC, click the WINS Configuration tab, and include the IP address of the PDC.

*You can enable WINS support on the Linux PDC by activating the **wins support** command in the smb.conf file as described in Chapter 3, "Setting Up Your Server File System."*

6. If you've set up a static IP address and need a connection to an external network such as the Internet, you'll need to add Gateway and DNS IP addresses on their respective tabs. Click OK when you're done.
7. Back in the Network dialog (Figure 2), click the Identification tab, where you can define three more things about this workstation:

   - **Computer name**: The NetBIOS name of the workstation; limited to 15 characters.
   - **Workgroup**: The name of the Workgroup or Domain that you're joining.

- **Computer Description**: A comment associated with this workstation.

8. Click the Access Control tab. If you're connecting to a regular peer-to-peer Workgroup, select "Share-level access control." If you're connecting to a Domain, select "User-level access control," and then enter the name of the Domain. Click OK when you're done.

If you're connecting to a Domain, you may get a message such as "Windows could not verify the specified security provider." If you do, click OK and wait a minute. It might take a short while before the Windows 9x/ME workstation finds the PDC.

9. If required, follow the prompts to insert your Windows 95/98/ME CD. Restart the computer when prompted.

10. After Windows 95/98/ME reboots, you'll get to log in to the Domain for the first time; **Figure 5** illustrates a login to the GRATEFUL Domain. (If you're connecting to a peer-to-peer Workgroup, the Domain option is not shown in Figure 5.)

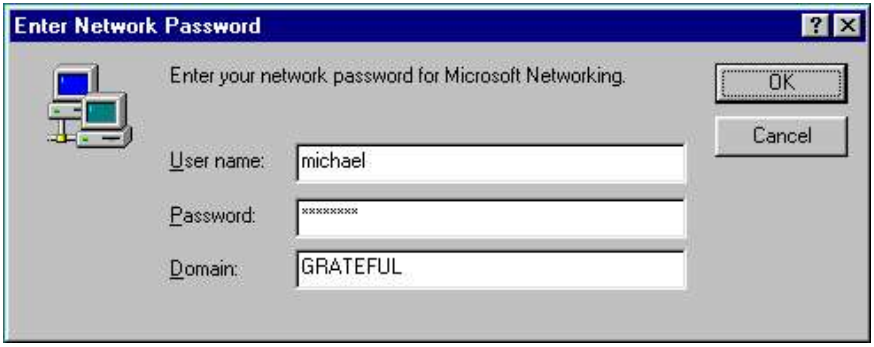

***Figure 5**. Logging in to the GRATEFUL Domain.*

Setting up roaming profiles

Roaming profiles allow users to get the same look and feel on their desktops regardless of which Windows 9x/ME workstation on a Domain they log in to. Roaming profiles are stored on the PDC, and when enabled, are sent over the network to the workstation. Unfortunately, Windows 9x/ME roaming profiles are not interchangeable with those available for Windows NT/2000/XP.

Not all administrators will want to create roaming profiles, because they can get quite large. For example, the roaming profile on my Windows XP Professional workstation is nearly 300MB. It takes quite a while to transmit that one profile from my PDC to my Windows XP Pro workstation on my Ethernet network. Network performance suffers during this process.

If your users have large roaming profiles and all log in at the start of a workday, that can easily tax the capacity of even faster networks. You may decide that it isn't efficient to set up roaming profiles for your users. In that case, you'll want to read the following with a view toward making sure that all user profiles are local.

To configure a roaming profile on a Windows 9x/ME computer, follow these steps:

1. Click Start | Settings | Control Panel.
2. Double-click Passwords in the Control Panel window.
3. Click the User Profiles tab. Make the selections shown in **Figure 6**.

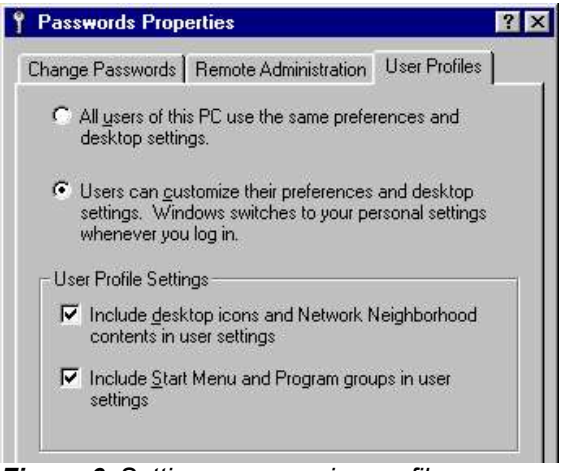

***Figure 6**. Setting up a roaming profile.*

If you want to disable roaming profiles, select the first option on the User Profiles tab in Figure 6: "All users of this PC use the same preferences and desktop settings."

4. Log off your system. Windows 9x/ME then transfers your user profile to the PDC. In the /etc/samba/smbusers file, Windows user **michael** is associated with Linux user **mj**; therefore, this profile is saved in the /home/mj directory.

If you're connecting to a peer-to-peer Workgroup, user profiles are stored locally. Because there is no central logon server, you can't set up a roaming profile.

Connecting to a share

There are three basic ways to connect to a shared directory or printer from a Microsoft Windows computer. You can connect graphically through Network Neighborhood, map a network drive to a known share, or connect from the MS-DOS prompt.

The most efficient way to view a Network Neighborhood is through Windows Explorer. To open it, click Start | Programs | Accessories | Windows Explorer. When you navigate to

Network Neighborhood, you can see the computers in your Domain. **Figure 7** shows the computers in my Grateful Domain.

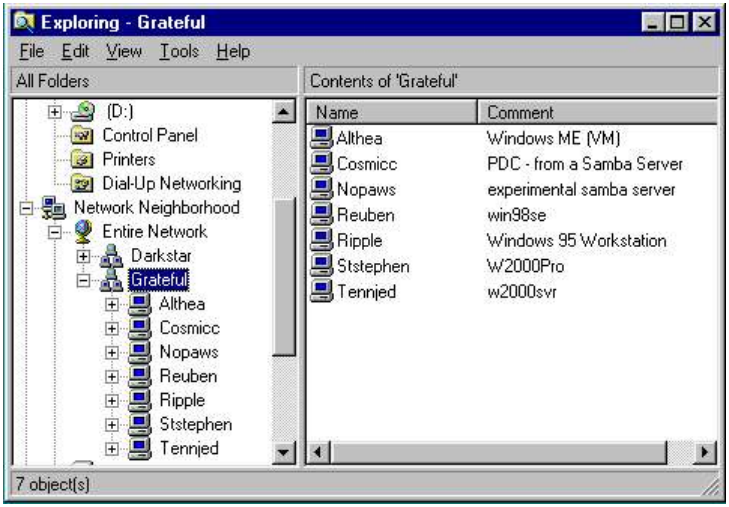

***Figure 7**. A Network Neighborhood view of a Domain.*

Once you've connected to a Domain via Network Neighborhood, highlight the computers of your choice. See what happens in the right-hand pane of Windows Explorer. You'll see the printers and directories that are shared and browseable from that computer.

As you can see in **Figure 8**, I've connected to my home directory on the PDC, where you can see the files associated with my Windows 9x/ME profile. As long as you have appropriate permissions through the smb.conf file and on the directory, you can use it just like any other directory on your workstation.

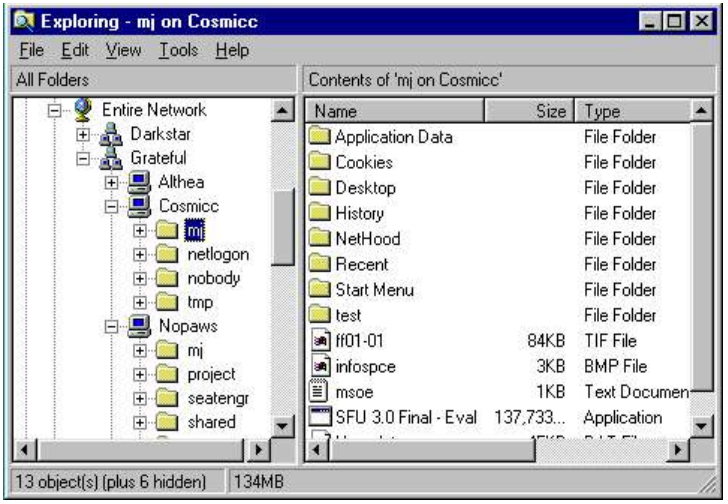

***Figure 8**. Using a shared directory from a Linux Domain member server.*

However, it can be cumbersome to drill down through Network Neighborhood to connect to a shared directory. A simpler option is to map that shared directory to a drive letter such as E:, F:, or G:. Once mapped, the drive letter will appear in Windows Explorer along with other local drives, which makes access to the shared directory much more convenient.

It's easy to specify a drive letter mount point. On the desktop, right-click My Computer and select Map Network Drive from the pop-up menu. In the Map Network Drive dialog, select from the available drive letters and set the path to the desired shared directory. For example, **Figure 9** illustrates how you would map drive E: to the same home directory (/home/mj) as shown in Figure 8.

Figure 10 illustrates the result in Windows Explorer.

***Figure 9**. Mapping to a mount point.*

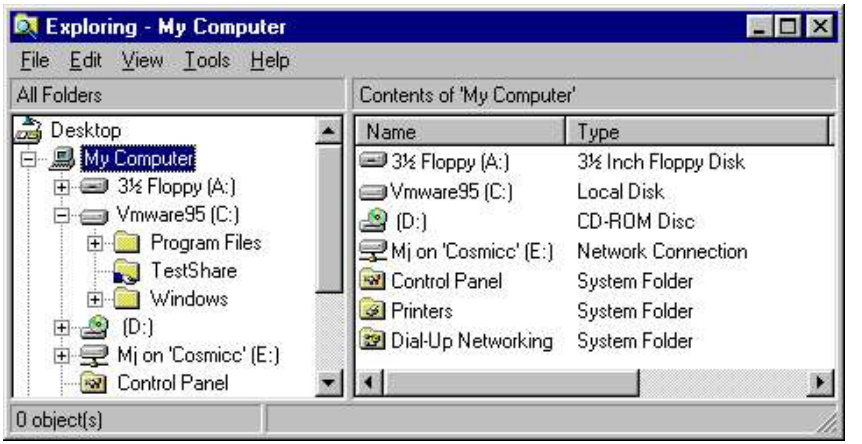

***Figure 10**. Regular and mapped drives.*

The shared directory is mounted to the noted drive letter, and is accessible as any other drive on your computer. Finally, you can mount a network drive in a similar fashion by using a text command at the MS-DOS prompt. Because the text commands apply to all Microsoft operating systems, I describe them in detail later in this chapter.

If you're prompted for a password, that's a problem on a Domain. If the user name and password you used to log in to Windows 9x/ME doesn't work for a share, the GUI prompt doesn't allow you to enter a different user name. You'll need to log out and log back in to the Windows 9x/ME computer with an appropriate user name.

Creating a Domain share

Now that you've logged in to a network, you can create a share from your workstation that's usable by other users on the Domain. To set up a share, point to a folder through Windows Explorer or My Computer. Right-click the folder that you want to share; if you've set up networking properly, you can select the Sharing command from the pop-up menu. This brings up a Properties dialog associated with the name of the directory that you want to share. As you can see in **Figure 11**, I've configured a directory on the local workstation named TestShare for access by several different users, as taken from the database of user names on the PDC.

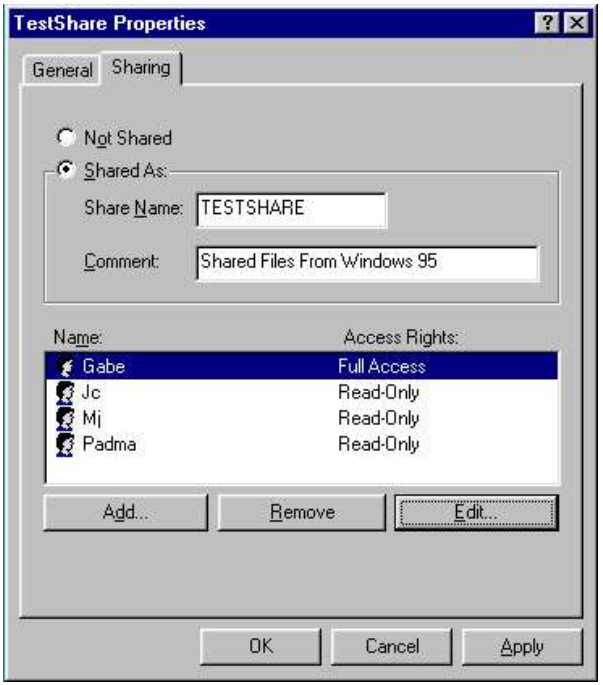

***Figure 11**. Sharing over a Domain from a Windows 95/98/ME directory.*

I described the nature of Linux permissions in Chapter 4, "Setting Up Your File Server's Users." As you can see, even the access rights that you can assign on a Windows 9x/ME computer are somewhat more fine-grained than what you can assign to a Linux file. However, the rights you set on a Windows 9x/ME workstation can be applied to users who log in from other Microsoft Windows computers on the network, even though the PDC is on a Linux computer.

Once you've granted access rights to certain users or groups, you can customize their rights. Highlight a user and click Edit. As you can see in **Figure 12**, you can customize the rights for individual users or groups. A detailed discussion of the rights that you can grant on a Windows computer is beyond the scope of this book.

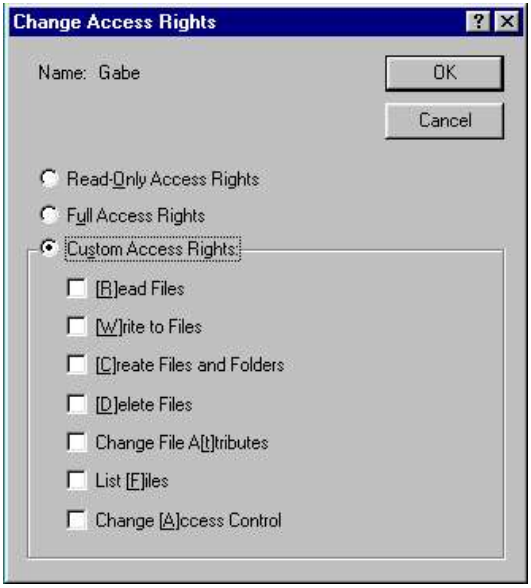

***Figure 12**. Customizing user access rights to a shared directory.*

Special Windows 98 issues

There are no basic problems with connecting Windows 98 computers to a peer-to-peer Workgroup or a Domain with a centralized database of user names and passwords. Windows 98 computers by default send passwords to the PDC in encrypted format. Naturally, Windows 98 includes drivers for more network adapters than are available on the Windows 95 CD.

Otherwise, there are few significant differences between Windows 98 and the other 16-bit Microsoft operating systems, at least with respect to network connections. Here are a couple of items that you might find useful:

- When you view Windows 98 network properties similar to Figure 2, Windows 98 always includes a default "Dial-Up Adapter" even if you don't have a telephone modem. Just be careful to configure the network card and not the modem.
- If you want to keep older (pre-OSR2) Windows 95 computers on your network, you'll need to set up clear text (non-encrypted) passwords on your network. While I don't recommend this procedure, it may be acceptable if your network is protected by a firewall. You can set up clear text passwords on a Windows 98 computer by applying the Win98_PlainPassword.reg file described in Table 1 to your Windows 98 registry.

Special Windows ME issues

There are no basic problems with connecting Windows ME computers to a Workgroup or Domain. Windows ME computers normally transmit passwords in encrypted format.

Functionally, Windows ME is very close to Windows 95/98. However, there are some significant differences in the look and feel of this operating system. For example:

- Windows ME includes My Network Places instead of Network Neighborhood. You can right-click the My Network Places icon to configure a connection to a Domain or a Workgroup. Once you see the Network Properties dialog, the actions you take are identical to Windows 95/98.
- If you double-click and open My Network Places, you can run the Add Network Places wizard to browse and connect to available shared directories.
- In My Network Places, the Home Networking wizard can help you configure your computer on a Domain or a peer-to-peer Workgroup. It can also help set up a connection through a network gateway to a remote network such as the Internet.
- If you're setting up clear text (non-encrypted) passwords on your network to accommodate older, pre-OSR2 Windows 95 computers, apply the WinME_PlainPassword.reg file described in Table 1 to your Windows ME registry.

Creating a Windows 95/98/ME Workgroup share

If you're setting up a peer-to-peer Workgroup, the steps you'll take to share a directory from a Microsoft Windows 95/98/ME workstation are slightly different. The main difference is that peer-to-peer Windows Workgroups use only passwords. User names are not required to connect to a shared directory from these 16-bit operating systems.

To include a Linux computer as a Samba server on a Windows Workgroup, you'll need to change at least the following variable in the /etc/samba/smb.conf configuration file:

```
security = share
```

Naturally, you'll need to change other variables, such as making sure that the **workgroup** variable is set to the name of the peer-to-peer Workgroup instead of a Domain. I describe the differences in more detail in Chapter 5, "Connecting Linux Workstations."

Creating a peer-to-peer Workgroup share on a Windows 9x/ME directory is a fairly straightforward process. Open Windows Explorer by clicking Start | Programs | Accessories | Windows Explorer, and then right-click the directory that you want to share. In the pop-up menu that appears, click Sharing. This should open a Properties window named for the directory that you're sharing, similar to **Figure 13**.

If you don't see Sharing in a pop-up menu when you right-click a folder, click Properties. If you can share a directory, click the Sharing tab. If you don't see a Sharing tab, you need to configure networking as described earlier in this chapter.

You can set up different passwords to share files from the folder. One password would support read-only sharing; a second password would support full-control shares, where connecting users have full control over the files and directories. If you set up both types of shares, you must use different passwords.

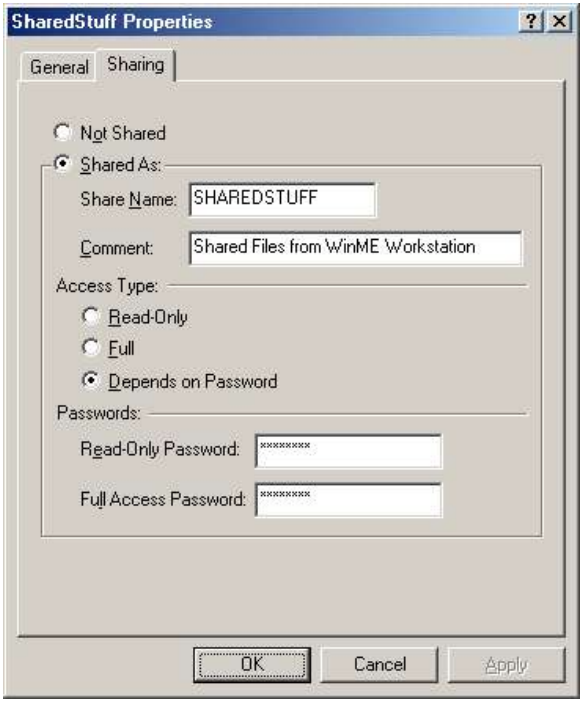

***Figure 13**. Setting up a peer-to-peer share in a Workgroup.*

Windows NT 4 Workstation

In this section, I'll show you how to connect a Windows NT 4 Workstation to a Domain governed by a Linux-based PDC. To this end, I'll illustrate how you can connect this 32-bit workstation to a network, set up roaming profiles, connect to shared directories on the server, and share its own resources with other computers in the network.

If you're setting up clear text (non-encrypted) passwords on your network to accommodate older (pre-OSR2) Windows 95 computers, apply the NT4_PlainPassword.reg file described in Table 1 to your Windows NT 4 Workstation registry.

There are two basic steps associated with setting up a Windows NT 4 Workstation computer on a network. First you need to configure networking on your computer. Then you can configure a network connection to a Workgroup or Domain.

Configuring a connection to a network

To configure a Microsoft Windows NT 4 Workstation on a network, have your Windows NT 4 Workstation installation CD ready, and then take the following steps:

1. Log in to NT 4 Workstation, using an account with administrative privileges.
2. Right-click the Network Neighborhood icon, and then click Properties. This opens the Network dialog shown in **Figure 14**.

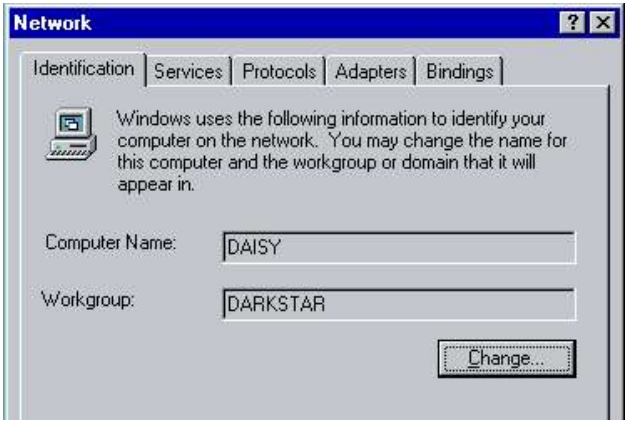

***Figure 14**. Configuring Windows NT 4 Workstation network services.*

If you don't see a Network Neighborhood icon in Microsoft Windows NT 4 Workstation, click Start | Settings | Control Panel. In the Control Panel window, click Network. (These steps also happen to work in Windows 95/98.)

3. Before you connect this computer to the Domain, you may want to check its network settings on the other tabs shown, which are described in **Table 2**.

***Table 2**. Network properties configuration tabs.*

Tab	Function
Identification	Lists the current computer name and Workgroup or Domain.
Services	Specifies services to be installed with the network protocol suite. You should have at least the Computer Browser, NetBIOS Interface, RPC Configuration, Server, and Workstation services installed; you can add them by using this tab, as long as you have the Windows NT 4 installation CD.
Protocols	Allows you to configure TCP/IP (or some other protocol stack) on this computer. More on this shortly.
Adapters	Lets you install and configure any network adapters on this computer.
Bindings	The NetBIOS, Server, and Workstation services are all normally bound as a WINS client to your network adapter.

4. Unless you have a DHCP server, you'll want to configure your TCP/IP settings manually on the Protocols tab. If you don't see those settings on that tab, you'll need to click Add and follow the prompts to add them. When you see TCP/IP Protocol, highlight it and then click Properties. As you can see from **Figure 15**, I've configured the manual IP address that I'm using on my Windows NT 4 Workstation computer.

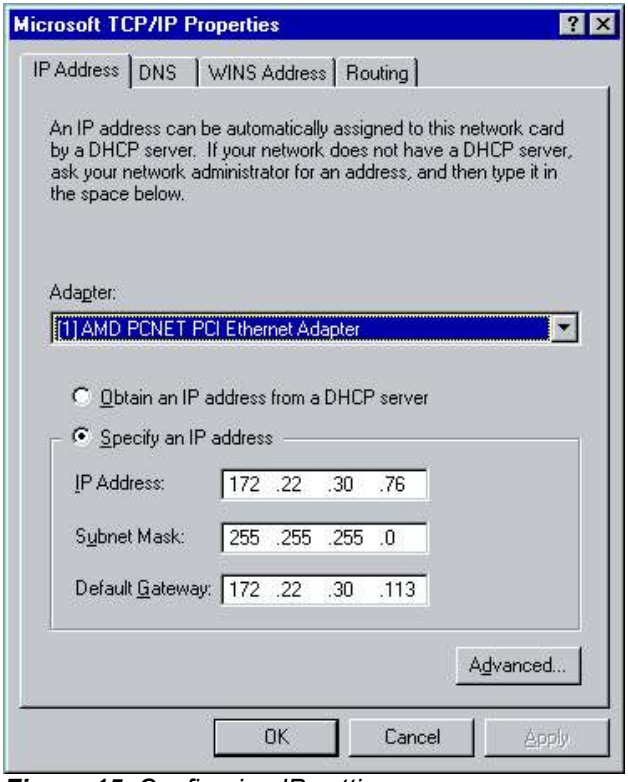

***Figure 15**. Configuring IP settings.*

You can configure other settings:

- Adapter: If you have more than one network adapter on this computer, you'll want different settings for each adapter. Different network adapters are frequently connected to different networks.
- If you have a DHCP server on your network, it's usually best to let it assign your IP address.
- If you click the Advanced button, you can configure different gateways as well as elementary firewalls.
- On the DNS tab, you can configure the DNS servers for your network.
- On the Routing tab, you can set up this computer as a router, a computer that allows two computers to communicate with each other.

If you have a DHCP server, it should at least assign an IP address, subnet mask, and a gateway address for each computer on your network. You can override your DHCP server by configuring your TCP/IP settings manually.

5. When you're finished configuring your IP address, click OK, and then return to the Identification tab of the Network dialog shown in Figure 14.

Configuring a connection to a Domain

Now that you've set up your NT 4 Workstation connection to a network, you can set it up for a connection to a Domain. To do so, let's go back to the Identification tab of the Network dialog.

1. Click Change to open the Identification Changes dialog shown in **Figure 16**.

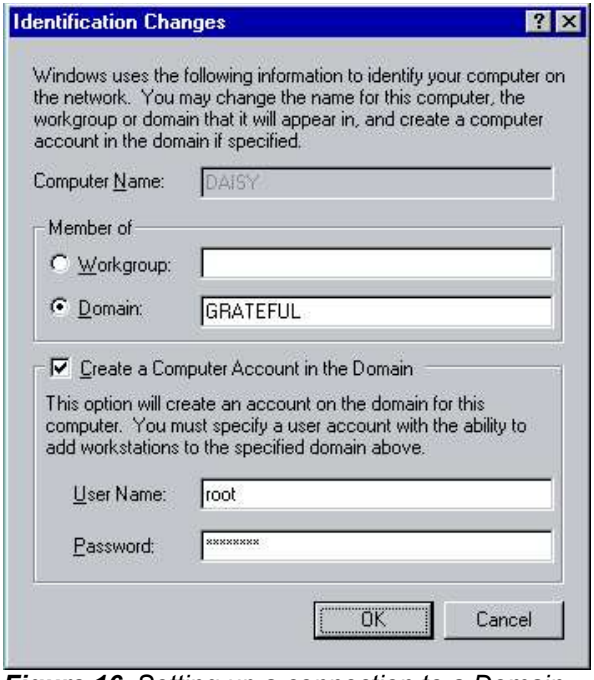

***Figure 16**. Setting up a connection to a Domain.*

2. To connect to a Domain, enter its name in the appropriate text box.
3. All Linux and Windows NT/2000/XP computers that connect to a Domain require a computer account on the PDC. You can create that computer account when you connect to the Domain for the first time. Note that you'll need the name and password of the administrative account on the PDC. (This assumes you've configured the **add user script** command in the PDC's smb.conf file as discussed in Chapter 4.)
4. If successful, you'll see a message welcoming you to the Domain that you selected. Click OK as needed to close the Network dialog.
5. Reboot the computer when prompted. Test the connection by logging in as a user on the Domain. Next, you'll see how you can implement roaming profiles on this Windows NT 4 Workstation computer.

6. Click Start | Shut Down. In the Shut Down Windows dialog, select "Close all programs and log on as a different user," and then click Yes.

Setting up a roaming profile

Roaming profiles allow users to get the same look and feel on their desktops, regardless of which Windows computer they use to log in to the Domain. Unfortunately, Windows 9x/ME roaming profiles are not interchangeable with those available for Windows NT/2000/XP.

As described earlier, roaming profiles can get quite large. For example, the roaming profile on my Windows XP Professional workstation is nearly 300MB. If your users have large roaming profiles, that can easily tax the capacity of many networks. In that case, you'll want to read the following with a view toward making sure that all user profiles are local.

If you want to configure a roaming profile on a Windows NT 4 Workstation computer, follow these steps:

1. Log in with an administrative account for the local computer (not the Domain).
2. Right-click the My Computer icon, and then click Properties to view the System Properties dialog.
3. Click the User Profiles tab. You can adjust the profiles of all users who've logged in to this particular workstation, as shown in **Figure 17**. As you can see, I've selected my Domain user **waymon**.

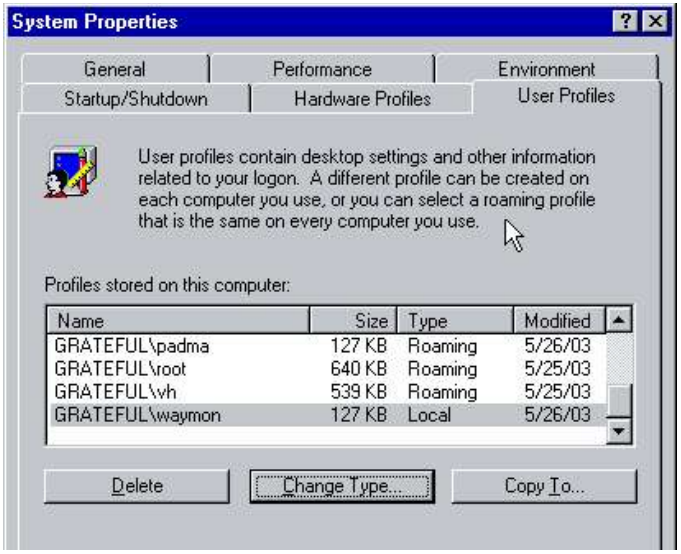

***Figure 17**. Local and roaming user profiles.*

4. Select a Domain user and click Change Type. As shown in **Figure 18**, you can store the profile locally. If you set a roaming profile, the profile is stored on the PDC, as defined by the [Profiles] share in the PDC's smb.conf file.

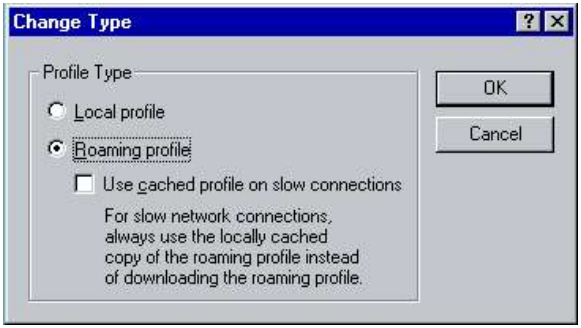

***Figure 18**. Switching between local and roaming profiles.*

Users who connect to a Domain over a slow connection such as a telephone modem should store profiles locally. If a roaming profile is desired, select the option "Use cached profile on slow connections."

5. Click OK to complete your changes. This saves the profile on the local computer. The next time **waymon** logs in to and out from this workstation, the profile is sent to and saved on the PDC computer.

Connecting to a Domain share

The easiest way to connect to a shared directory is through Windows Explorer. You can also mount the shared directory to a Microsoft Windows drive letter. Later in this chapter, I'll show you how to mount to a shared directory from the command-line interface.

To access the Windows Explorer file browser, click Start | Programs | Windows NT Explorer. (Depending on your profile, you may need to click Start | Programs | Accessories | Windows Explorer.)

You can navigate to the shared directories on the Grateful Domain through the Network Neighborhood, as shown in **Figure 19**. As you can see, I've connected to the share named "shared," which permits user access as defined in Chapter 4. The **nopaws** server is a Linux computer, configured with Samba as a Domain member server.

The first time you connect to a shared directory though a Windows workstation, you may have trouble connecting to a share on which you think you have permissions. While your workstation may have found the PDC, the PDC may not have found your computer yet on the browse list. Before doing anything else, try logging off and logging back in to that workstation.

It's easy to specify a drive letter mount point. Right-click My Computer and select Map Network Drive from the pop-up menu. In the Map Network Drive dialog, select from the available drive letters. Windows NT is a bit different from Windows 9x/ME in that you can browse the available shares, as shown in **Figure 20**.

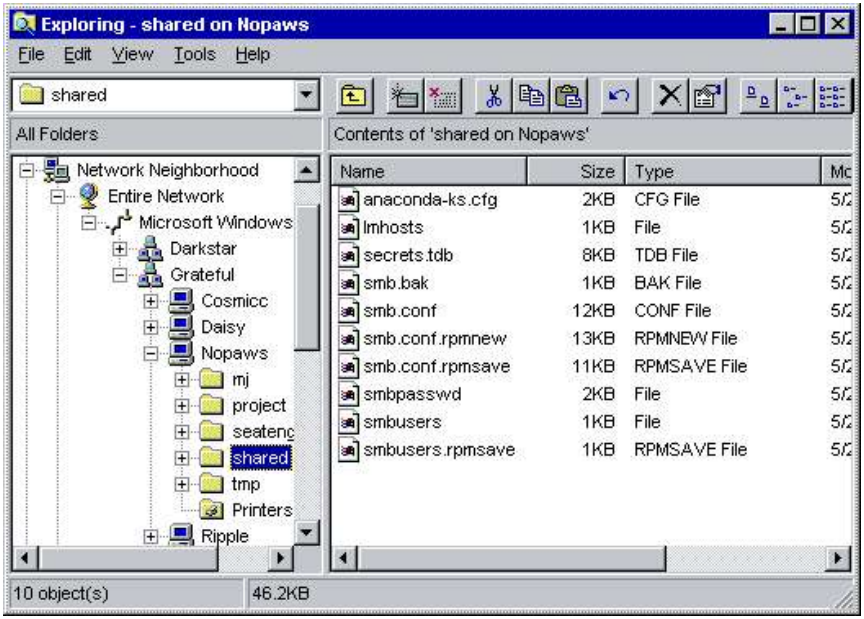

***Figure 19**. Navigating to a shared directory.*

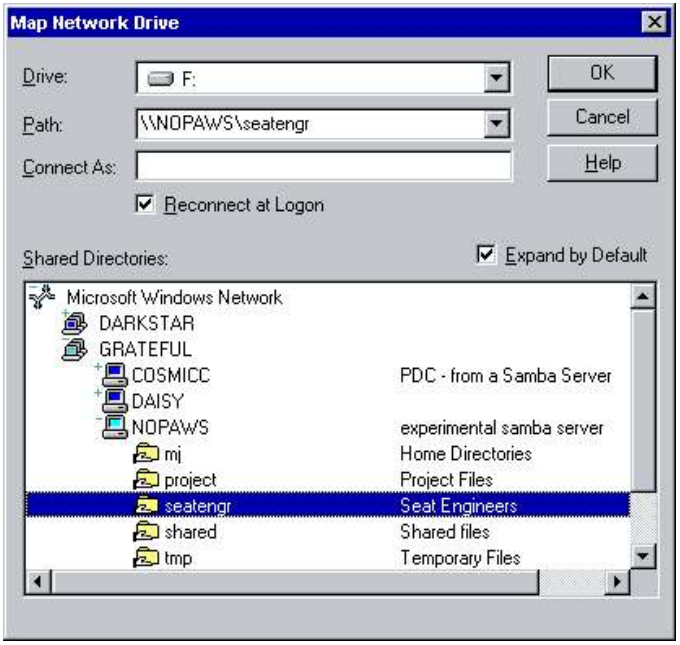

***Figure 20**. Mapping from Windows NT 4 Workstation to a mount point.*

When you highlight a specific share from a computer on the Domain, the path is automatically shown in the Path text box. Figure 20 illustrates a connection to the seatengr share on the **nopaws** Domain member server.

If the connection does not work, you can enter a different user name in the Connect As text box. If the user name corresponds to one on your Domain, you'll be prompted for a password.

The shared directory is mounted to the noted drive letter, and is accessible as any other drive on your computer. Whether you can read from or copy to the share depends on the permissions set on the Linux Domain member server in the governing smb.conf file.

Finally, you can mount a network drive in a similar fashion from the MS-DOS prompt. Because the text commands apply to all Microsoft operating systems, I describe them in detail later in this chapter.

Connecting to a Workgroup share

If you've configured a Windows NT 4 Workstation computer on a peer-to-peer Workgroup, the process for connecting to a share depends on whether it's from a Windows 9x/ME or a Windows NT/2000/XP computer. In either case, open Windows Explorer (or Windows NT Explorer) as described earlier. When you select a shared directory, Windows prompts for a user name and password.

For connections to a Windows 9x/ME share, you don't need a user name to connect to a shared directory. When you see the Enter Network Password dialog shown in **Figure 21**, you don't need to enter a user name. Just enter the read-only or full-access password associated with the shared directory and click OK.

***Figure 21**. Connecting to a shared directory on a peer-to-peer Workgroup.*

Don't worry about the error message shown in Figure 21. It just means that the password you used to log in to the workstation is different from the read-only or full-access password for the share. If the password does match, you won't see the Enter Network Password dialog, and will automatically get the permissions associated with the password.

On a Windows NT/2000/XP share, connections depend on the allowed users on the target computer. If share access is limited to a specific user name, you need to log in with the same user name on the local computer.

For example, I have a share named Downloads on my Windows XP Professional computer named **allaccess**. I limit access to the Domain user named **michael**, as configured on **allaccess**. If I log in to my Windows NT 4 Workstation computer as a Local or Domain user named **elizabeth**, I won't have access to the \\allaccess\downloads shared directory.

Creating a Domain share

Now that you've logged in to a network, you can create a share that's usable by other users on the Domain. To set up a share, point to a folder through Windows Explorer or My Computer. Right-click the folder that you want to share.

If you've set up networking properly, you'll be able to select the Sharing command in the pop-up menu. This brings up a Properties dialog associated with the name of the directory that you want to share. Make any changes and then click Permissions. As you can see in **Figure 22**, I've configured a directory named ShareTest for access from several different users, as taken from the database of user names on the PDC.

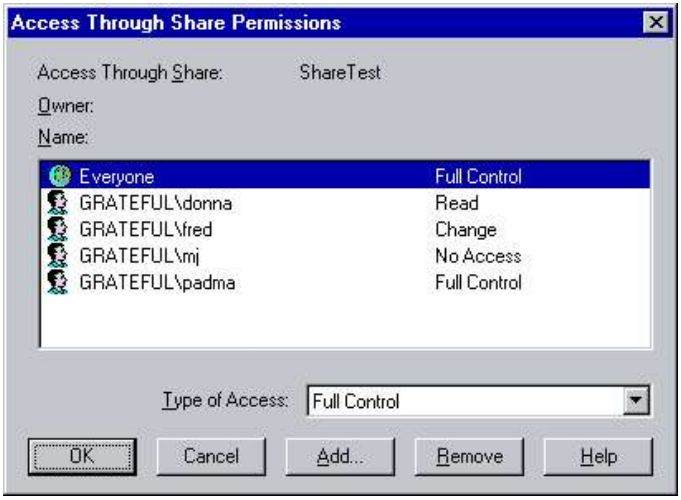

***Figure 22**. Setting up share access for specific users.*

If you want to add more users or groups to the share list, click Add. In the Add Users and Groups dialog, you can add more users or groups from your Domain.

On the Security tab, you can also set permissions, audit rights, and ownership for the subject directory. Details of this process are extensive and are beyond the scope of this book.

Creating a Workgroup share

You can set up shared directories from Windows NT 4 Workstation on a peer-to-peer Workgroup. To set up a share, log in to the workstation with an administrator account. The remaining steps are virtually identical to creating a share for a Domain.

The only difference is that the users that you can add to the permissions list are limited to those configured on the local NT 4 Workstation computer. By definition, there is no centralized list of users on a PDC. For example, assume I've given permissions to a shared directory named ShareTest to the user named **michael** on an NT 4 Workstation named **daisy**. Also assume I've deleted the "Everyone" entry from the Access Share Through Permissions dialog shown in Figure 22.

Other computers in the Workgroup can connect to this Workgroup share. From a Linux workstation, I'd mount this share on the /mnt/source directory with the following command:

```
# smbmount //daisy/sharetest /mnt/source -o username=michael
```

*Before a regular user can use the **smbmount** command, you need to set appropriate permissions as described in Chapter 5, "Connecting Linux Workstations."*

From a Windows workstation, I can connect if I can log in with a *local* user account named **michael**. I'd create that account if it didn't already exist.

Then after I log in, and I navigate to the share through Network Neighborhood, the local Windows workstation automatically passes **michael**'s user name and password to the share on the NT 4 Workstation named **daisy**.

Windows 2000 Professional

In this section, I'll show you how to connect a Windows 2000 Professional workstation to a Domain governed by a Linux-based PDC. To this end, I'll illustrate how you can connect this 32-bit workstation to a network, set up roving profiles, connect to shared directories, and share with other computers on the network.

If you need to accommodate older Windows 95 (pre-OSR2) computers on your network, you can set up clear text (non-encrypted) passwords by applying the Win2000_PlainPassword.reg file described in Table 1 on your Windows 2000 Professional workstation registry.

There are two basic steps associated with setting up a Windows 2000 Professional computer on a network. First, you need to configure networking on your computer. Then you can configure a network connection to a Workgroup or Domain.

Configuring a connection to a network

To configure a Microsoft Windows 2000 Professional workstation on a network, have your Windows 2000 Professional installation CD ready, and then take the following steps:

1. Log in to Windows 2000 Professional, using an account with administrative privileges on the *local* computer.
2. Right-click the My Network Places icon, and then click Properties. This opens the Network and Dial-up Connections window.
3. If you see a Local Area Connection icon, double-click it. This should open the Local Area Connection Status dialog. Click the Properties button to open the Local Area Connection Properties dialog shown in **Figure 23**.

If you don't see a Local Area Connection icon in the My Network Places window, you may need to reinstall the network card on your computer.

4. Select Internet Protocol (TCP/IP), and then click Properties. This opens the Internet Protocol (TCP/IP) Properties dialog.
5. If you don't have a DHCP server on your network, you'll need to enter your IP address and DNS server information in the appropriate text boxes.
6. Click the Advanced button to open the Advanced TCP/IP Settings dialog. Click the WINS tab, where you can enter information for the WINS server(s) on your network.

7. If needed, use the other tabs to add more information that you might need. The function of each tab is described in **Table 3**. When you're satisfied with the settings, click OK.

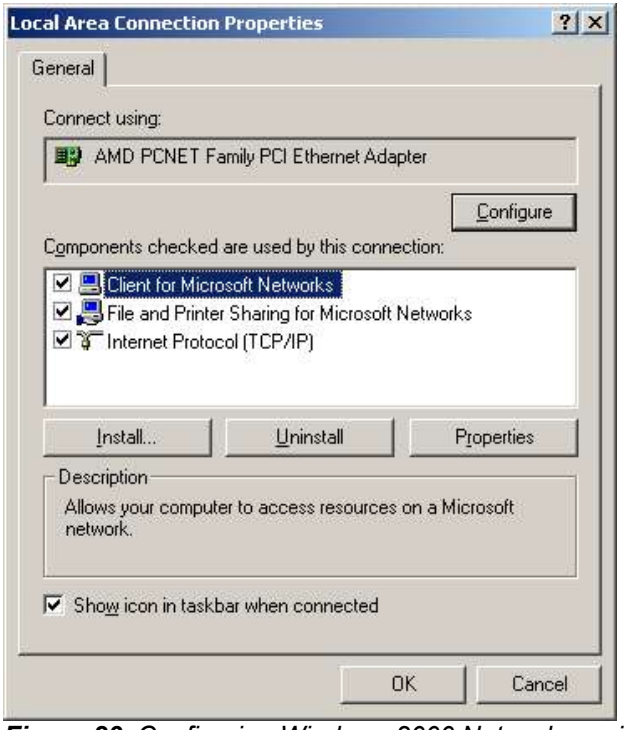

***Figure 23**. Configuring Windows 2000 Network services.*

***Table 3**. Tabs in the Advanced TCP/IP Settings dialog.*

Tab	Function
IP Settings	Lets you assign additional IP addresses and default gateways to this particular network card.
DNS	Allows you to specify any additional DNS servers that you might need, such as those specified by your network's ISP.
WINS	Supports configuration of a WINS server for this computer; to support connections to a Linux-based PDC, be sure to select the "Enable NetBIOS over TCP/IP" option.
Options	Permits configuration of a firewall on this computer.

Unlike Windows 9x/ME or NT, changes that you make to many network properties of a Windows 2000 Professional computer are incorporated immediately and generally do not require you to reboot your computer.

Configuring a connection to a Domain

Now that you've set up your Windows 2000 Professional computer connection to a network, you can set it up for a connection to a Domain, using the following steps:

1. Right-click the My Computer icon. In the pop-up menu that appears, click Properties to open the System Properties dialog. Click the Network Identification tab shown in **Figure 24**. As you can see, this computer is currently named **ststephen**, a member of the DARKSTAR Workgroup.

***Figure 24**. Configuring Windows NT 2000 network services.*

The options available to connect a Windows 2000 or a Windows XP Professional computer to a Domain are nearly identical. In this section, I'll show you the Network Identification Wizard. In the corresponding section on Windows XP Professional, I'll show you how to do the same thing with the Properties button.

2. Click the Network ID button to start the Network Identification Wizard.
3. When the Network Identification Wizard opens, click Next.
4. You're asked whether this computer is part of a business network or for home use. To accommodate the use of Linux computers on this network, select the business network option and then click Next.

If you want to set up this Windows 2000 computer on a peer-to-peer Workgroup or a Domain, select the business network option.

In the following step, I'm assuming that you want to make your computer a member of a Domain.

5. If you choose the other option (Properties), the Network Identification Wizard allows you to make this computer a member of a Workgroup. You'll just need to name the Workgroup and follow the prompts until you reboot this computer.
6. If you're making this Windows 2000 computer a member of a Domain, you'll need the information described in **Figure 25**. Click Next after you've collected this information.

***Figure 25**. A list of data you need to join a Domain.*

7. In the following screen, enter a user name and password on the PDC (not the workstation), and the name of your Domain. Click Next.
8. If you haven't connected this computer to the PDC before, you'll need to create a computer account on the Domain. Enter the name of your computer and the Domain. Click Next.
9. The Network Identification Wizard prompts you for an administrative account on the PDC. The standard administrative account on a Linux computer is root. You'll need to enter **root**, the root user password, and the name of the Domain. Assuming you set up the **Add User Script** command in the PDC's smb.conf file as described in Chapter 4, Linux should then create a computer account for you on the PDC.
10. Next, you can set up a Domain user for access to resources on this computer, as shown in **Figure 26**. After you've added a user from the Domain, click Next.
11. Now you can assign the user various levels of access. For example, I can set **donna** to be a standard power user, or a restricted user. Alternatively, if I select Other, I can select from one of the groups described in **Table 3**.

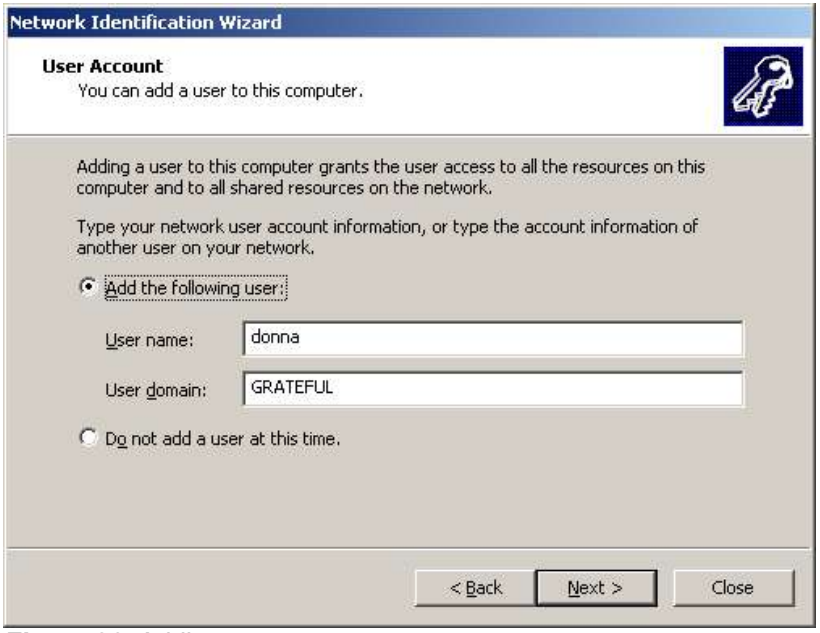

***Figure 26**. Adding a user.*

***Table 3**. Microsoft Windows user categories.*

Category	Function
Standard User	Allowed to log in, modify computer settings, and install applications on the local computer.
Restricted User	Allowed to log in, and read and write files.
Administrators	Given complete access to the Domain.
Backup Operators	Given full access solely for backing up or restoring files.
Guests	A restricted user, with additional limits.
Replicator	Domain users who are allowed to copy directories from computer to computer.

12. Click Next to complete the Network Identification Wizard. When you click Next once again, and then click OK to close the System Properties dialog, you'll be prompted to reboot this workstation. When you reboot, this computer will be connected to the Domain.
13. Test the connection by logging in as a user on the Domain. Next, you'll see how you can implement roaming profiles on a Windows 2000 Professional computer.
14. Click Start | Shut Down. In the Shut Down Windows dialog, select "Log off *username*," and then click OK.

Setting up a roaming profile

Roaming profiles allow users to get the same look and feel on their desktops when they log in from any Windows NT/2000/XP workstation on a Domain. Unfortunately, Windows 9x/ME roaming profiles are not interchangeable with those available for Windows NT/2000/XP.

Not all administrators will want to create roaming profiles, because they can get quite large. For example, the roaming profile on my Windows XP Professional workstation is nearly 300MB. If your users have large roaming profiles, that can easily tax the capacity of many networks. In that case, you'll want to read the following with a view toward making sure all user profiles are local.

To configure a roaming profile on a Windows 2000 Professional computer, follow these steps:

1. Log in with the user account of your choice from the Domain.
2. Right-click My Computer, and then click Properties. This opens the System Properties dialog.
3. Click the User Profiles tab. Roaming profiles are enabled by default, as shown in **Figure 27**. As you can see, I've logged in as the Domain user **mary**.

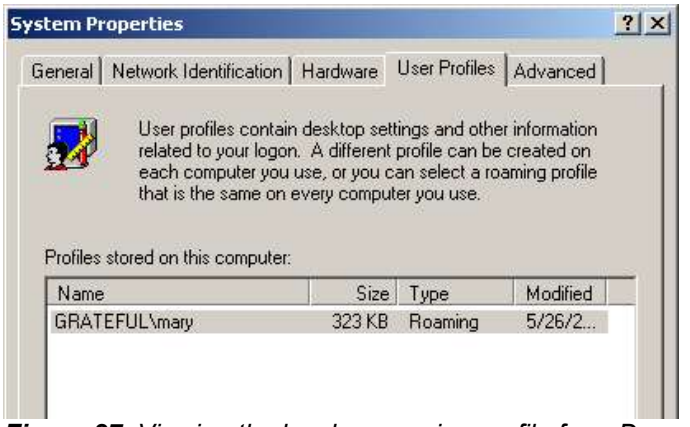

***Figure 27**. Viewing the local or roaming profile for a Domain user.*

4. Select the listed user and click Change Type to open the Change Profile Type dialog, which allows you to switch the configuration for this Domain user between a local and a roaming profile.

Connecting to a Domain share

The easiest way to connect to a shared directory in Windows 2000 is through My Network Places in Windows Explorer. You can also mount the shared directory to a Microsoft Windows drive letter. Later in this chapter, I'll show you how to mount to a shared directory from the command-line interface.

To access the Windows Explorer file browser, click Start | Programs | Accessories | Windows Explorer. You can navigate to the shared directories on the Grateful Domain through

My Network Places, as shown in **Figure 28**. In this case, you can see the different directories shared from the **nopaws** computer, configured with Samba as a Domain member server.

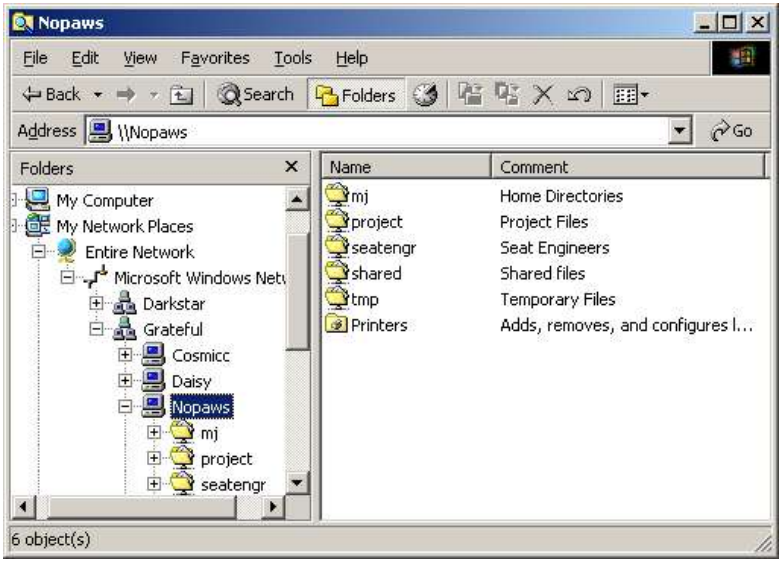

***Figure 28**. Navigating to a shared directory.*

The first time you connect to a shared directory though a Windows workstation, you may have trouble connecting to a share where you think you have permissions. You may be prompted to enter your Domain user name and password a second time. The Microsoft browse list may not be up to date.

It's easy to specify a drive letter mount point. Right-click My Computer and select Map Network Drive from the pop-up menu. In the Map Network Drive dialog, select from the available drive letters. Windows 2000 is a bit different from Windows NT or Windows 9x/ME in that you can configure a connection as a different user, as shown in **Figure 29**.

If you have more than one user account on a Domain, you can use this feature to connect to different shares on that Domain.

To select the desired share, enter its name in the Folder text box. Alternatively, you can view available shares. Click Browse, and then select the desired share in the Browse For Folder dialog, as shown in **Figure 30**.

Highlight a specific share from a computer on the Domain. When you click OK, the path is automatically shown in the Folder text box of the Map Network Drive dialog. If needed, you can connect with a different Domain user name and password. For example, if I logged in as user **mj**, I might want access to user **mary**'s directories. When you click the link associated with **mary**'s home directory, it opens the Connect As dialog shown in **Figure 31**.

***Figure 29**. Mapping a drive letter to a shared directory.*

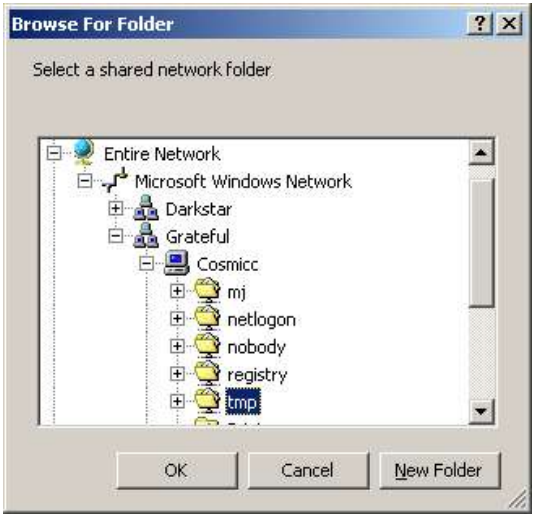

***Figure 30**. Browsing through shared directories.*

***Figure 31**. Connecting as a different Domain user.*

The shared directory is mounted to the noted drive letter, and is accessible as any other drive on your computer. Whether you can read from or copy to the share depends on the permissions set on the Linux Domain member server in the governing smb.conf file.

Finally, you can mount a network drive in a similar fashion from the MS-DOS prompt. Because the text commands apply to all Microsoft operating systems, I describe them in detail later in this chapter.

Connecting to a Workgroup share

If you've configured a Windows 2000 Professional computer on a peer-to-peer Workgroup, the process for connecting to a share depends on whether you're connecting from a Windows 9x/ME or a Windows NT/2000/XP computer. In either case, open Windows Explorer as described earlier.

For connections to a Windows 9x/ME share, you'll normally see the Enter Network Password dialog shown in **Figure 32**; you don't need to enter a user name. (Don't worry about the "Incorrect password ..." error message.) Just enter the read-only or full-access password associated with the shared directory and click OK.

***Figure 32**. Connecting to a shared directory on a peer-to-peer Workgroup.*

When you log in to a Windows 2000 Professional computer in a peer-to-peer Workgroup, the password you use matters. If it's the same password as is used to share a Windows 9x/ME directory, you won't see Figure 32. You're automatically given the Workgroup permissions (Read-only or Full) associated with your login password. Otherwise, the passwords don't match, and you'll see Figure 32 with the associated error message.

Peer-to-peer Workgroup connections to a Windows NT/2000/XP share work in the same way as from a Windows NT 4 Workstation computer. They depend on the allowed users on the target computer. If share access is limited to a specific user name, you need to log in with the same user name on the local computer.

For example, I have a share named Downloads on my Windows XP Professional computer named **allaccess**. On **allaccess**, I limit access to the user named **michael**. If I log in to my Windows 2000 Professional computer as a user named **elizabeth**, I won't have access to the \\allaccess\downloads shared directory. I need to log in as **michael** to get access to that particular share.

Windows XP Professional

In this section, I'll show you how to connect a Windows XP Professional workstation to a Domain governed by a Linux-based PDC. To this end, I'll illustrate how you can connect this 32-bit workstation to a network, set up roving profiles, connect to shared directories, and share with other computers in the network.

If you need to accommodate older Windows 95 (pre-OSR2) workstations, you can set up clear text (non-encrypted) passwords on your network. Apply the Win2000_PlainPassword.reg file described in Table 1 to your Windows XP Professional workstation registry. In this case, the same registry commands work for both Windows 2000 and Windows XP Professional.

The Windows XP Home operating system is designed only for home networks. While you can set it up on a peer-to-peer Workgroup, you can't use it to log in to a PDC. Therefore, I do not cover Windows XP Home in this book.

There are two basic steps associated with setting up a Windows 9x/ME computer on a network. First, you need to configure networking on your computer. Then you can configure a network connection to a Workgroup or Domain.

Configuring a connection to a network

To configure a Microsoft Windows XP Professional workstation on a network, have your Windows XP Professional installation CD ready, and then take the following steps:

1. Log in to Windows XP Professional, using an account with administrative privileges on the local computer.
2. Copy the following file from the /usr/share/doc/samba-2.2.7a/docs/Registry directory: WinXP_SignOrSeal.reg. It's required to allow a Windows XP Professional workstation to join a Domain configured on a Linux PDC.

*In a Microsoft Windows registry, the **RequiresSignOrSeal** value is associated with Microsoft digital signatures. The registry file disables these signatures for a connection to a Linux PDC. Also, if you're using a version of Samba other than 2.2.7a, the location of the registry file will change accordingly.*

3. Run the registry file from a viewer such as Windows Explorer; the appropriate command is automatically applied to the Windows XP Professional registry.
4. Click Start | Connect To | Show All Connections to open the Network Connections window shown in **Figure 33**. If you see a lot of connections, scroll to the bottom of the window.

***Figure 33**. Viewing Windows XP Professional network connections.*

5. Examine the icon associated with your Local Area Connection. If it's "Bridged" as shown in Figure 33, right-click the Network Bridge icon. Otherwise, right-click the Local Area Connection icon. In either case, click Properties in the pop-up menu that appears. This should open the properties associated with your connection. Click the Properties button. This should open the Local Area Connection Properties dialog shown in **Figure 34**.

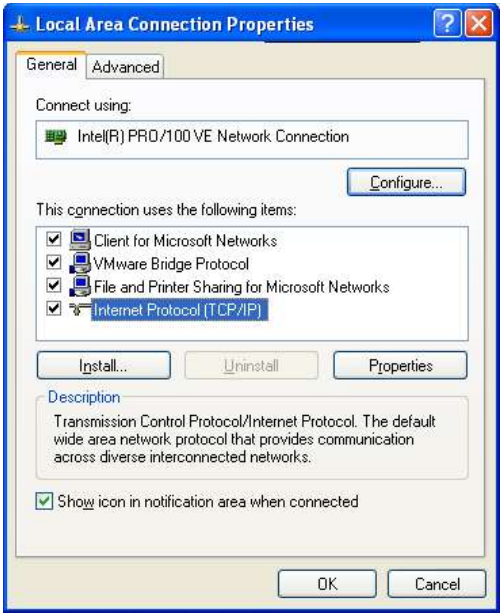

***Figure 34**. Windows XP network properties.*

6. If you don't have a DHCP server on your network, or otherwise need to manually configure your network settings, highlight Internet Protocol (TCP/IP) and then click Properties. This opens the Internet Protocol (TCP/IP) Properties window, where you can configure the IP address and DNS servers for your Windows XP Professional workstation.
7. In the Internet Protocol (TCP/IP) Properties window, click Advanced. This opens the Advanced TCP/IP Settings window, which is functionally identical to that used on a Windows 2000 Professional computer. This window is covered earlier in this chapter in Table 3.

Configuring a connection to a Domain

Now that you've set up your Windows XP Professional computer connection to a network, you can set it up for a connection to a Domain, using the following steps:

1. Click Start and point to My Computer. Right-click and select Properties from the pop-up menu that appears. This opens the System Properties dialog.
2. Select the Computer Name tab. I illustrate one view of this tab in **Figure 35**. In the "Computer description" text box, you can set the comment associated with your computer in a Domain or Workgroup browse list.

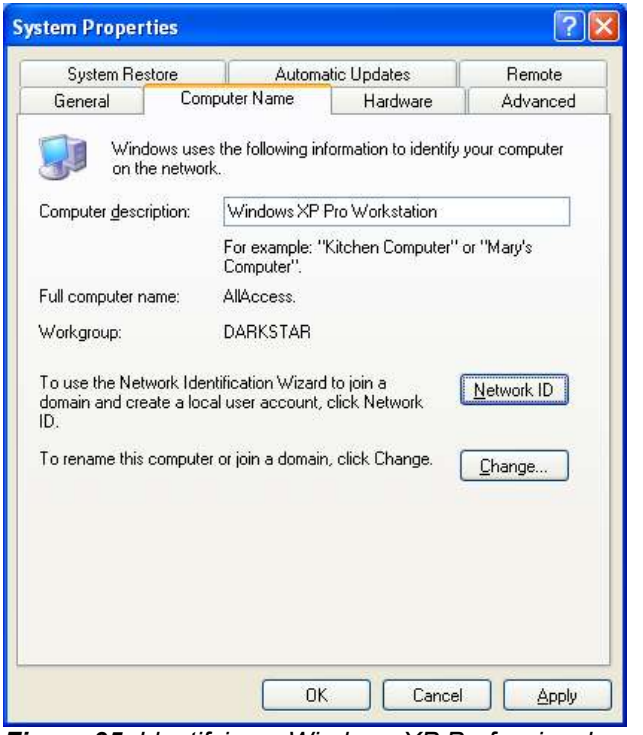

***Figure 35**. Identifying a Windows XP Professional computer.*

3. Click Change to open the Computer Name Changes dialog shown in **Figure 36**.

***Figure 36**. Configuring your computer on a Domain or Workgroup.*

Alternatively, you can click Network ID to start the Network Identification Wizard. The steps you use in this wizard for Windows XP Professional Workstation are identical to those for Windows 2000 Professional.

4. It's easy to configure this computer as part of a Domain. Click the Domain radio button, and then enter the name of the Domain that you want to join. Change the computer name if desired, and then click OK. If your computer detects a PDC on the Domain, it opens the Computer Name Changes dialog.
5. Enter the user name and password of an administrative user on the PDC. Assuming you're using a Linux PDC, enter the root user name and password.

Sometimes a workstation may not find a PDC right away, and you'll see an error. Assuming your network is properly configured, wait a few minutes and try again.

6. You should see a confirmation message such as "Welcome to the GRATEFUL Domain." Click OK.
7. You'll be told that you need to reboot before the changes take effect. Click OK, and then click OK again to exit the System Properties dialog.
8. When prompted, click OK again to reboot your computer.

Setting up a roaming profile

Roaming profiles allow users to get the same look and feel on their desktops, when they log in from any Windows NT/2000/XP workstation on a Domain. Unfortunately, Windows 9x/ME roaming profiles are not interchangeable with those available for Windows NT/2000/XP.

Not all administrators will want to create roaming profiles, because they can get quite large. For example, the roaming profile on my Windows XP Professional workstation is nearly 300MB. If your users have large roaming profiles, that can easily tax the capacity of many networks. In that case, you'll want to read the following with a view toward making sure all user profiles are local.

If you want to configure a roaming profile on a Windows XP Professional computer, follow these steps:

1. Boot your computer. When prompted, log in with an account on the Domain.

If you get an error message during the login process, you may have forgotten to apply the WinXP_SignOrSeal.reg file to your Windows XP computer registry, as described earlier.

2. Click Start. From the Start menu, right-click My Computer, and then click Properties from the pop-up menu that appears. This opens the System Properties dialog.
3. Select the Advanced tab. Under User Profiles, click Settings. Roaming profiles are enabled by default, as shown in **Figure 37**. As you can see, I've logged in as the Domain user **donna**.

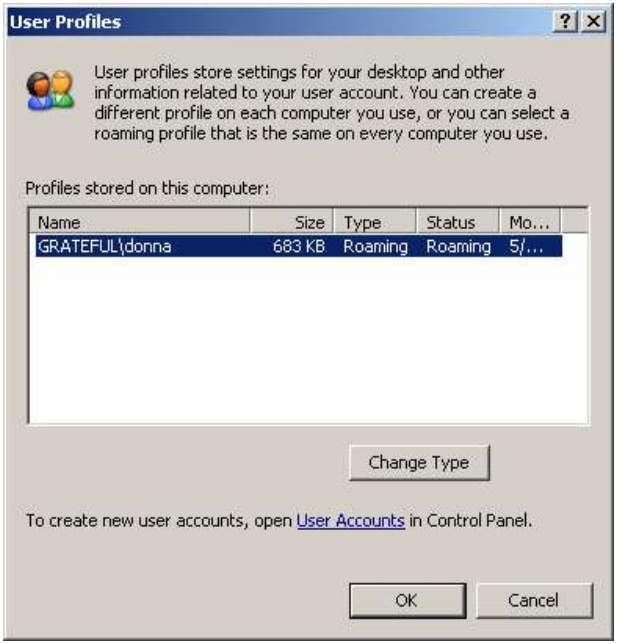

***Figure 37**. Local or roaming profile for a Domain user.*

4. Select the listed user and click Change Type. This opens the Change Profile Type dialog, which allows you to switch the configuration for this Domain user between a local and a roaming profile.

Connecting to a Domain share

The easiest way to connect to a shared directory is through My Network Places in Windows Explorer. You can also mount the shared directory to a Microsoft Windows drive letter. Later in this chapter, I'll show you how to mount to a shared directory from the command-line interface.

To access the Windows Explorer file browser, click Start | All Programs | Accessories | Windows Explorer. You can navigate to the shared directories on the Grateful Domain through My Network Places, as shown in **Figure 38**. In this case, you can see the different files available to this user from the project directory shared from the **nopaws** Linux computer. It's configured with Samba as a Windows Domain member server.

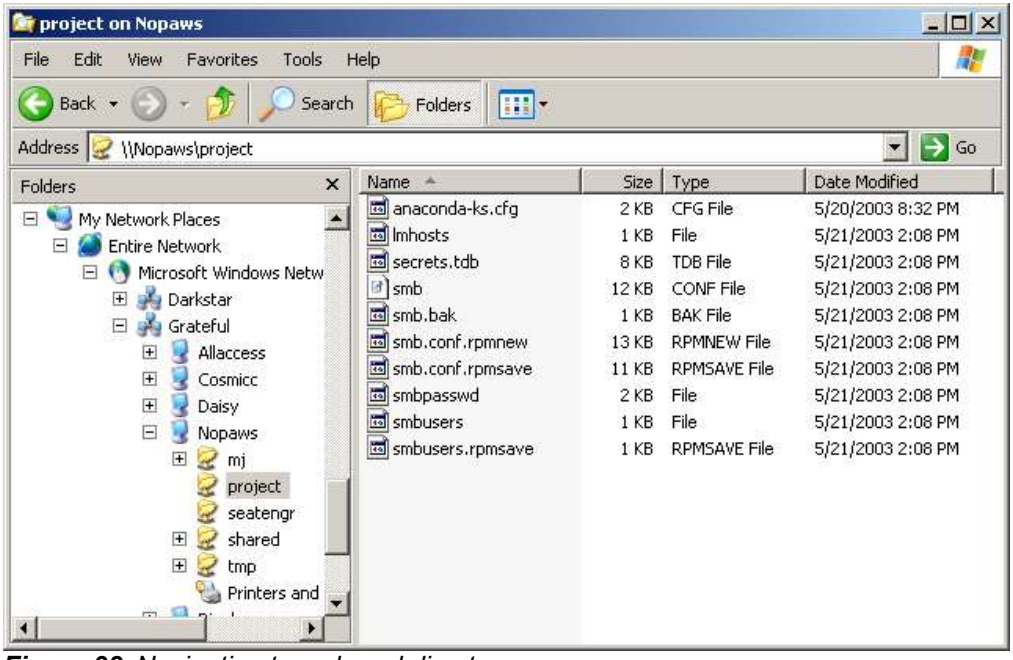

***Figure 38**. Navigating to a shared directory.*

The first time you connect to a shared directory on a Domain through a Windows workstation, you may have trouble connecting to a share where you think you have permissions. You may be prompted to enter your Domain user name and password again. That's OK; it can take a few minutes before other computers on the Domain can find your computer on the browse list.

Finally, you can mount a network drive in a similar fashion from the MS-DOS prompt. Because the text commands apply to all Microsoft operating systems, I describe them in detail later in this chapter.

Connecting to a Workgroup share

If you've configured a Windows XP Professional computer on a peer-to-peer Workgroup, the process for connecting to a share depends on whether you're connecting from a Windows 9x/ME or a Windows NT/2000/XP computer. In either case, open Windows Explorer as described earlier.

For connections to a Windows 9x/ME share, you'll see the Connect to *Computername* dialog shown in **Figure 39**; you're not allowed to enter a user name. Just enter the read-only or full-access password associated with the shared directory and click OK.

***Figure 39**. Connecting to a shared directory on a peer-to-peer Workgroup.*

When you log in to a Windows XP Professional computer in a peer-to-peer Workgroup, the password you use matters. If it's the same password as is used to share a Windows 9x/ME directory, you'll get the Workgroup permissions (Read-only or Full) associated with that password.

On a Windows NT/2000/XP share, connections work slightly differently from a Windows NT 4 Workstation/2000 Professional computer. They depend on the allowed users on the target computer. When you access the remote computer, you're prompted for the user name and password from the target Windows computer.

For example, I have a share named StStephDocs on my Windows 2000 Professional computer named **ststephen**. I limit access to the user named **michael**, as configured on **ststephen**. If I log in to my Windows XP Professional computer as a user named **donna**, Windows XP Professional prompts me for the user name and password on the Windows 2000 Professional computer.

Text-mode network commands

You can connect to shared directories from the command-line interface on a Microsoft Windows workstation. Functionally, the process is similar to the way you can map network drives in Microsoft Windows. The key commands are:

```
C:\> net view
C:\> net use
```

The **net use** command is especially useful, because it's something you can use in a logon script to connect users on your Domain to the shared directories of your choice. I'll show you how these commands work in the following sections.

Both of these commands work from the MS-DOS command-line interface, which you can open in Microsoft Windows. Click Start | Run and then enter **command** in the Open text box.

Viewing computers

Just as you can use the **smbclient** command to view the computers on a Workgroup or a Domain from the Linux command-line interface, you can use the **net view** command to view computers from the MS-DOS prompt.

*Sometimes when you run the **net** command from the DOS prompt, you may get an error message such as "Access is denied." If your connections are sound, don't let that discourage you. If you've just logged in to the computer, it may take a minute or two for the computers to find each other.*

As you can see in **Figure 40**, the **net view** command alone returns a list of the current computers in your Workgroup or Domain. The **net view /domain:darkstar** command is somewhat counter-intuitive; it allows me to view the computers in my **darkstar** peer-to-peer *Workgroup*. The command would work just as well if **darkstar** were the name of a Domain.

*If my computer is a member of a different Workgroup or Domain, I can view the computers in my Grateful Domain with the following command: **net view /domain:grateful**.*

You might note that the STSTEPHEN computer comes up in the browse list in both my Domain and Workgroup. That just tells me that I've recently moved STSTEPHEN from the Workgroup to the Domain (or vice versa).

You can also use the **net view** command to browse shared directories in any accessible computer on a Domain or Workgroup. For example, I can use the following command to view the shared directories on my Linux Domain member server named **nopaws**:

```
C:\> net view \\nopaws
```

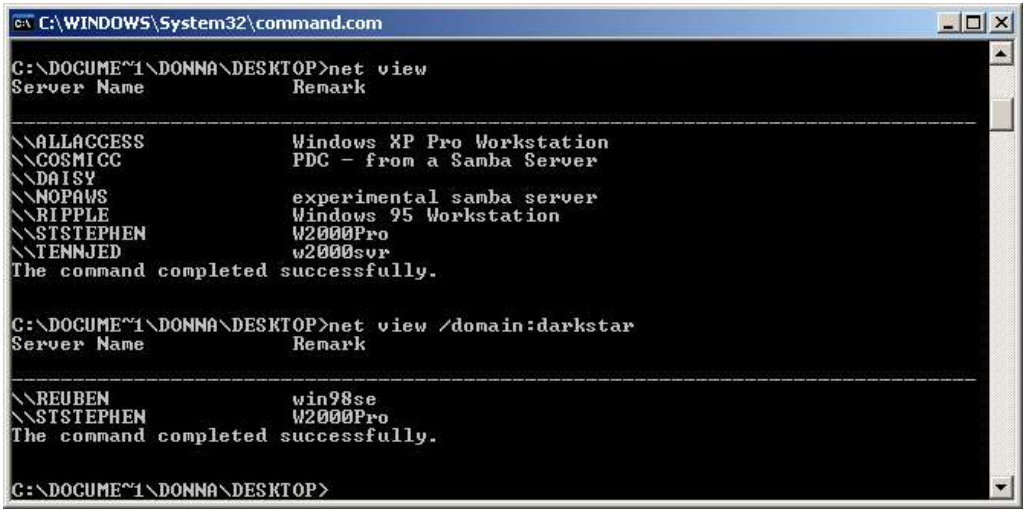

***Figure 40**. Viewing Domain and Workgroup members from the DOS prompt.*

Mounting shared directories

I've shown you the GUI way to mount a shared directory from a remote computer to a Microsoft drive letter. In Chapter 5, you learned to mount a shared directory from a remote computer to a Linux directory. You can incorporate the commands you see in this section into login batch files that run these commands automatically when a user logs in to the Domain.

As described earlier in this chapter, the batch files are located on the PDC, as defined by the [netlogon] share.

If you want to mount to a directory shared from a Domain member server, you need to log in to the workstation as the desired user. Then you should be able to mount the desired share easily with the following command:

```
C:\> net use L: \\nopaws\tmp
The command completed successfully.
```

In this case, I've mounted the [tmp] share from the **nopaws** member server on the Microsoft Windows workstation L: drive.

Sometimes on a Domain, user names and passwords don't get to the server in time; you'll need to re-enter your Domain user name and password one more time. If this happens with a login batch file, you may need to run the batch file directly from the Linux PDC's [netlogon] share, or reconnect to shared directories manually.

If you want to mount to a directory shared from a computer on a peer-to-peer Workgroup, the commands are similar. For example, I've shared the My Documents directory from my

Windows 98 computer named **reuben**. I can mount it to the M: drive with the following command (don't worry about the "password is invalid" error message):

```
C:\> net use M: \\reuben\my documents"
The password is invalid for \\reuben\my documents.

To connect reuben\Guest to reuben, press ENTER, or type a new user name:
Enter the password for 'reuben\Guest' to connect to 'reuben':
The command completed successfully.
```

The **net use** command first tries the user name and password you used to log in to the workstation. Unless there's an exact match, you'll see the "password is invalid" error message. Because **reuben** is a Windows 98 computer on a peer-to-peer Workgroup, no user name is required. Don't enter anything when prompted for a user name. When prompted for the password, enter the read-only or full-access password for that shared directory.

Observe how I've used double-quotation marks to make sure Windows reads the full name of the shared My Documents directory. Otherwise, because "My Documents" includes a space, the ***net use*** *command would look for a shared "My" directory.*

Troubleshooting

Whenever Samba starts, it's recorded in the logs. Whenever something goes wrong with Samba, it's recorded in the logs. Whenever another computer has a problem connecting to a Samba server, it's recorded in the logs. In Red Hat Linux, the Samba log files are stored in the /var/log/samba directory.

But before looking at the logs, there's the matter of syntax.

Samba syntax

It's easy to check the syntax of the Samba configuration file, smb.conf. All you need is the **testparm** command. If you have a problem with one of your variables, it should show up in the output. For example, if I misspelled something in my smb.conf file, I might see the following output:

```
# testparm -x | more
Load smb config files from /etc/samba/smb.conf
Unknown parameter encountered: "workgrou"
Ignoring unknown parameter "workgrou"
Processing section "[homes]"
Processing section "[printers]"
Processing section "[tmp]"
Processing section "[shared]"
Processing section "[project]"
Processing section "[seatengr]"
Loaded services file OK.
Press enter to see a dump of your service definitions
```

I used the **testparm -x** command to show all Samba variables that don't correspond to the default, and the **| more** switch so I can scroll through the list of variables one at a time. And

testparm identified the syntax error in my smb.conf file; I misspelled "workgroup." When you press Enter, **testparm** scrolls through the Samba variables, one page at a time.

*If you leave out the **-x** switch, **testparm** lists all variables associated with your Samba configuration. Try it out; you'll see that it's a long list.*

Samba logs

If there are problems with Samba, and the syntax of the smb.conf file checks out, the next step is to check the logs. Typical Samba problems include invalid passwords, services such as Winbind that are not running, and browser elections that are not working. Linux log files are stored in the /var/log directory; Samba log files are stored in the /var/log/samba directory. Access to these files is limited to the root user on the Linux PDC.

Take a look at these files in **Figure 41**. As you can see, it includes log files named after every computer that I've connected to this Samba server.

```
[root@CosmicC root]# cd /var/log/samba/
[root@CosmicC samba]# ls
allaccess.log    delilah.log.1    reuben.log.1    smbmount.log.2
allaccess.log.1  localhost.log    reuben.log.2    smbmount.log.3
allaccess.log.2  localhost.log.1  reuben.log.3    smbmount.log.4
allaccess.log.3  log.nmbd         reuben.log.4    ststephen.log
allaccess.log.4  log.smbd         rh9test.log     ststephen.log.1
althea.log       log.winbindd     rh9test.log.1   ststephen.log.2
cosmicc.log      nmbd.log         ripple.log      ststephen.log.3
cosmicc.log.1    nmbd.log.1       ripple.log.1    ststephen.log.4
cosmicc.log.2    nmbd.log.2       smbd.log        sugaree.log
cosmicc.log.3    nmbd.log.3       smbd.log.1      sugaree.log.1
daisy.log        nmbd.log.4       smbd.log.2      tennjed.log
daisy.log.1      nopaws.log       smbd.log.3      tennjed.log.1
daisy.log.2      nopaws.log.1     smbd.log.4      tennjed.log.2
daisy.log.3      nopaws.log.2     smbmount.log    tennjed.log.3
delilah.log      reuben.log       smbmount.log.1  tennjed.log.4
[root@CosmicC samba]#
```

***Figure 41**. Samba log files on a Linux PDC.*

Primary log files

The primary log files in /var/log/samba tell you about problems with the Samba, NetBIOS, and Winbind daemons. Here are more details about each of these log files:

- **log.smbd**: Every time the Samba daemon is started, you'll see another entry in this file, with date and time.
- **log.nmbd**: On a Linux PDC, this file includes master browser data, including elections. Every time the NetBIOS daemon, **nmbd**, is started, you'll see another entry in this file. (For more information on browser elections, see Chapter 3, "Setting Up Your Server File System.")
- **log.winbindd**: On a Linux/Samba Domain member server, this file records problems with connections to a PDC on a Microsoft Windows computer.

- **smbd.log**: Any problems with the Samba daemon are recorded in this file.

*In Red Hat Linux 9, there is a bug with the standard C language libraries, as defined by the **glibc** RPM package. This bug results in frequent error messages about "oplocks." Use the Red Hat Update Agent described in Chapter 2 to update your system. Make sure you've downloaded a **glibc** RPM package later than 2.3.2. If you don't upgrade, connection performance to a Samba PDC may be intermittent. For more information, see **bugzilla.redhat.com/bugzilla/show_bug.cgi?id=90036**.*

- **nmbd.log**: This file stores NetBIOS queries and master browser election information.

Workstation log files

For each workstation on your network, you may see two common errors:

```
Error = No route to host
Error = Connection reset by peer
```

These standard errors happen when you connect a computer to a Domain, when network connections fail, or after you restart Samba. In any of these cases, it may take some time before the PDC, master browser, and workstation computers can find each other, which leads to these noted errors. In other words, you need not worry about these errors unless the computers on your network aren't able to find each other for a reasonable period of time, such as 15 minutes. This may vary depending on your hardware setup and the number of computers on your network.

Most of the log files in my /var/log/samba directory are associated with specific workstation computers on my network. Red Hat Linux by default keeps up to five weeks of log files, rotated with a numeric extension. For example, I have five weeks of logs for connections from the computer named **ststephen**. The current week's log for connections from that computer are stored in ststephen.log. The previous week's log is stored in ststephen.log.1, and so on.

In other words, you can collect a history of logs, to check the performance of your Domain on the Linux computer that you've configured as the PDC.

Conclusion

With Samba, you can configure a Microsoft-style Domain with users on Microsoft workstations who log in to a Linux Primary Domain Controller. Before you start the process, you should check the configuration on your PDC for computer accounts, logon scripts, and user profiles as needed.

Then you can start configuring Microsoft Windows workstations to join your Domain. The steps you take for Microsoft Windows 95, 98, and ME are quite similar. While the steps you take for Windows NT, 2000, and XP are somewhat different, the basic concepts are the same:

- Make sure that networking is properly configured.
- Set up the connection to the Domain. On Microsoft Windows 9x/ME, you can do this through the network properties for the workstation. On Microsoft Windows NT/2000/XP, the steps vary.
- Once you're connected to the Domain, set up roaming profiles if desired.
- Connect to shares as needed.
- Share the directories of your choice with other users on the Domain.

You can perform many of these functions from the MS-DOS command-line interface. For example, you can browse Domains and Workgroups with the **net view** command. You can connect to shared directories with the appropriate **net use** command.

You can also set up Microsoft and Linux computers in a peer-to-peer network. When you connect to shared directories in a peer-to-peer network, the steps vary depending on whether the source is a Windows 9x/ME, Windows NT/2000/XP, or a Linux member server.

Updates and corrections to this chapter can be found on Hentzenwerke's Web site, **www.hentzenwerke.com**. Click "Catalog" and navigate to the page for this book.