TriTeal Enterprise Desktop (TED™) 4.2

TEDFS Installation and User's Guide

Printed in USA

Part No. TED024004200

TriTeal Corporation, 2011 Palomar Airport Road, Carlsbad, CA 92009 U.S.A.
Hewlett-Packard Company, 3000 Hanover Street, Palo Alto, CA 94304 U.S.A.
International Business Machines Corp., Route 100, Somers, NY 10589 U.S.A.
Sun Microsystems, Inc., 2550 Garcia Avenue, Mountain View, CA 94043 U.S.A.
Novell, Inc., 190 River Road, Summit, NJ 07901 U.S.A.
The Santa Cruz Operation, Inc., 400 Encinal Street, Santa Cruz, CA 95060 U.S.A.
Digital Equipment Corp., 111 Powdermill Road, Maynard, MA 01754 U.S.A.
Fujitsu Limited, 1015, Kamikodanaka Nakahara-Ku, Kawasaki 211, Japan
Hitachi, Ltd., 6, Kanda Surugadai 4-Chome, Chiyoda-ku, Tokyo 101, Japan
Network Computing Devices, Inc, 350 North Bernardo Avenue, Mountain View, CA 94043-5207 U.S.A.
SPYRUS, Inc., 2841 Junction Avenue, Suite 110, San Jose, CA 95134, U.S.A.
Spyglass, Inc., 1240 East Diehl Road, Naperville, IL 60563 U.S.A.

Printing History
First Edition ... September 1996
Second Edition ... April 1997

Sales Information and Technical Support
TriTeal Corporation
Tel: (760) 930-2077
Fax: (760) 930-2081
Support E-Mail: support@triteal.com
Sales E-Mail: info@triteal.com
World Wide Web: http://www.triteal.com/
Documentation comments: docs@triteal.com

Please
Recycle

Contents

Welcome

Welcome to TEDFS. This guide introduces you to TEDFS and gets you up and running quickly. You should read it if you're the TEDFS Administrator or an advanced user.

In this guide you'll discover why TEDFS is the easiest way to let PC users access UNIX files and printers, and how to tailor your TEDFS configuration using the TEDFS Profile Editor.

What is TEDFS?

TEDFS turns an Administrator's nightmare into a dream come true. It answers the question: PC or UNIX?

Users prefer PCs on their desktops, with good reason: the productivity tools they want to use are PC programs. But Administrators like UNIX systems for their reliability and configurability. UNIX servers are business critical.

TEDFS lets PC users access files on UNIX servers just as if they were on another PC—through network drives or their Network Neighborhood.

Not only that, with TEDFS PC users can print to UNIX printers—just like any Windows network printer.

For the TEDFS Administrator, there's an easy-to-use Windows program to configure TEDFS—no configuration files to edit, and no new file format to remember.

More importantly, there's no PC installation. Ten minutes is all it takes to give any number of PCs access to a UNIX server, whatever the size of the network.

TEDFS server

The TEDFS server is a UNIX program with a difference—it speaks the same language as PCs for sharing files and printers across a network. This language is a Windows standard, developed originally by Microsoft and Intel, so it's the only sensible choice for integrating UNIX with your Windows users.

TEDFS effectively disguises a UNIX server as a PC. To other PCs and to users, a TEDFS server looks just like any other PC.

TEDFS Profile Editor

The TEDFS Profile Editor is a Windows program, used by the TEDFS Administrator to configure the server.

The Profile Editor is closely integrated with the server, presenting an intuitive interface complete with context-sensitive help and step-by-step instructions. It performs as-you-type validation of many settings, giving instant and helpful visual feedback. Multi-level undo and redo lets you change your mind, and change it back again, many times.

The benefits of TEDFS

Here are some compelling reasons why TEDFS gives the best solution for Windows-to-UNIX integration.

Best for PCs

- There's no need to install complex software on every user's PC, saving a significant amount of time when setting up or upgrading the TEDFS server.
- TEDFS doesn't use any extra valuable memory or disk space on PCs.

Best for UNIX

- UNIX servers are perfect for network installations of disk-hungry products that can be used by multiple users.
- UNIX is robust, reliable and scalable, perfect for coping with the ever-changing, ever-growing world of PCs.

Best for users

- With no special PC software, users don't have to change the way they work, or learn confusing new tools.
- If a program works with Windows, it works with TEDFS.

Best for Administrators

- Installed in just ten minutes; a few questions to answer, no fuss, no bother.
- Highly configurable through the Profile Editor, with fine-grained access control and instant feedback.

Setting up TEDFS

TEDFS is typically installed as an optional package with other TED applications such as NTED or TED for Windows. The installer will automatically install, configure, and execute the TEDFS server.

Once TEDFS is installed, set up and running, you need to log into a PC as the TEDFS Administrator, and use the Profile Editor to configure the server if you want to share files other than each user's home directory. The user's home directory on UNIX is automatically exported with TEDFS.

Make sure you read the README in the TEDFS installation directory. It contains any last-minute information that couldn't make it into this book.

See Also – "A tour of the Profile Editor," in Chapter 1, "The Basics."

PC system requirements

To access the server, PCs must use TCP/IP as their transport protocol. For Windows 95 or Windows NT 4.0, a TCP/IP protocol stack is provided with the operating system.

See Also – The TED CD.

About the TEDFS Administrator

The TEDFS Administrator is a special UNIX user, with sole responsibility for configuring the TEDFS server. You can make any UNIX user the TEDFS Administrator, but choose carefully. Upon installation, the administrator is "root."

The TEDFS Administrator is:

- As powerful as the UNIX superuser.
- The only user allowed to run the Profile Editor.

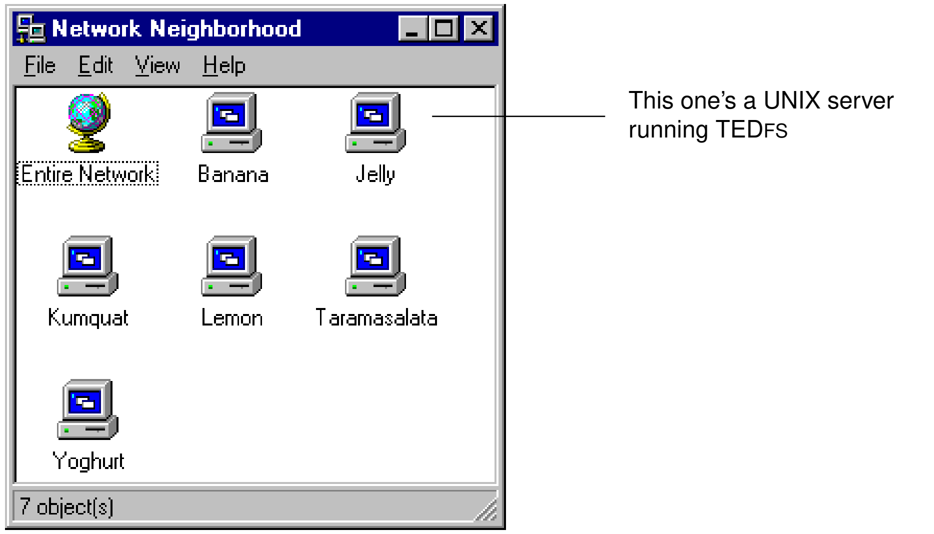
The administrator name can be changed in the Profile Editor. You'll need to log into Windows using the TEDFS Administrator's username to run the Profile Editor if you can not specify "connect as" when mapping a network drive, such as in Windows 95.

Important – The Administrator's UNIX password is as important as the superuser's. Keep it secure.

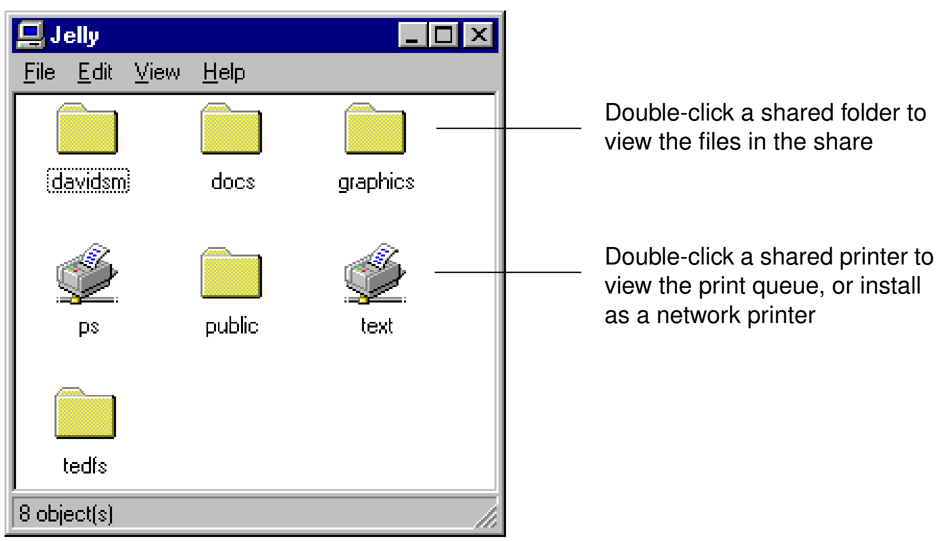
Using a TEDFS server

Setting up a UNIX server to run TEDFS is easy. Using a TEDFS server couldn't be easier.

The TEDFS server appears with other computers in a Windows workgroup. (The only problem might be distinguishing the TEDFS server from the PCs.)



TEDFS makes any number of UNIX directories and printers available for use over the network. These network resources are called *shares*.



The TEDFS Administrator chooses the name for the server, which workgroup it appears in, and sets up the shares people can access.

What's in this book

In **Chapter 1, "The Basics",** you'll take a tour of the Profile Editor and its extensive Help. You'll be introduced to some Windows and TEDFS terms, and create your first share.

In **Chapter 2, "Beyond the Basics",** you'll discover some of the more sophisticated features of the TEDFS Profile Editor and server, including the different access rights you can grant to users.

In **Chapter 3, "The Possibilities",** you'll discover just a few of the ways in which you can take advantage of the flexibility of the Profile Editor and server, using real-world examples.

In **Chapter 4, "Issues for Administrators",** you'll learn how to control the TEDFS server from UNIX, and find information about things that matter to you, such as security, licensing and troubleshooting.

The Basics 1

Look in this chapter to find out how to get started with TEDFS and the Profile Editor.

You'll take a tour of the Profile Editor and its extensive Help. You'll be introduced to some Windows and TEDFS terms, and create your first share.

A tour of the Profile Editor

In this section, we'll point out the important parts of the TEDFS Profile Editor: what you see, and how you use it. We'll show you how the Profile Editor's instant validation and feedback stops you making mistakes, and how to get Help.

Finding and starting the Profile Editor

You use the TEDFS Profile Editor to configure the TEDFS server. A complete TEDFS configuration is called a server *profile*.

The TEDFS Profile Editor is stored on the UNIX server, not on any PC. Every TEDFS server has its own Profile Editor—you can't use the same Profile Editor with more than one server.

You access a TEDFS server's Profile Editor through a share on the server. This share, called TEDFS, is created automatically by TEDFS Setup.

How you access the TEDFS share depends on your version of Windows:

- Windows 95 and Windows NT 4 users can use Network Neighborhood (or Windows Explorer).

- Windows for Workgroups and Windows NT 3.51 users need to map a drive using File Manager.

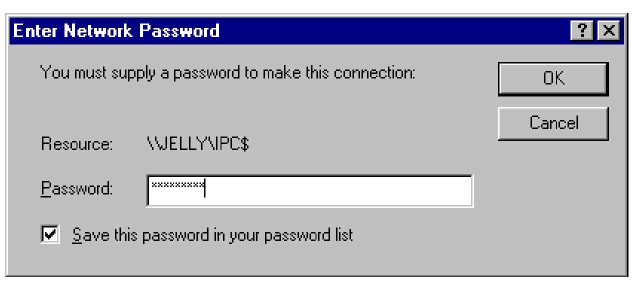
Note – Only the TEDFS Administrator can run the Profile Editor. The Administrator is named during TEDFS Setup; you can change it using the Profile Editor, if you wish.

See Also – "Administrator, naming" in the Help index.

▼ *To find and start the Profile Editor using Network Neighborhood*

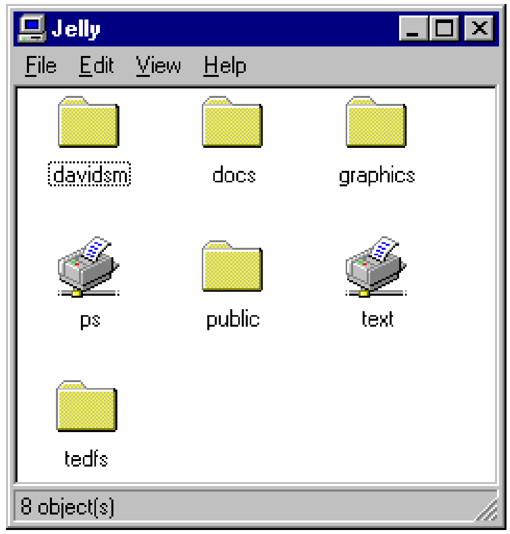
1. Log into Windows as the TEDFS Administrator.
2. Double-click Network Neighborhood and locate the TEDFS server. If it's not in your workgroup, double-click Entire Network and open the workgroup it's in.

3. Double-click the TEDFS server.

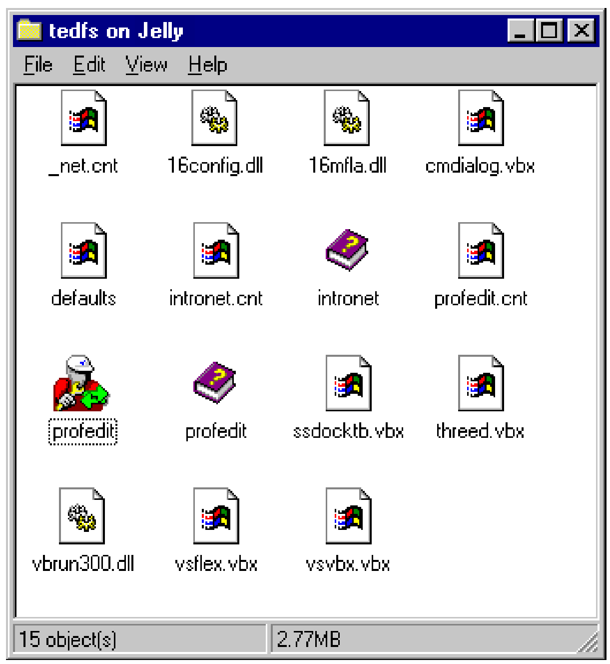


4. If prompted, type the TEDFS Administrator's UNIX password.

5. Double-click the TEDFS share.



6. Double-click the Profile Editor.

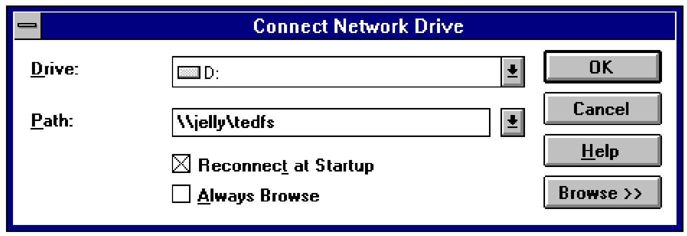


Tip – Create a shortcut to the Profile Editor on your desktop.

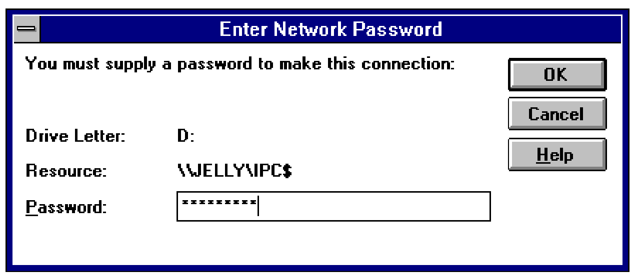
You can also click Start, click Run, then type `\\`*server*`\TEDfs\profedit.exe` (replacing *server* with the name of your TEDFS server) to start the Profile Editor without browsing the network.

To find and start the Profile Editor using File Manager

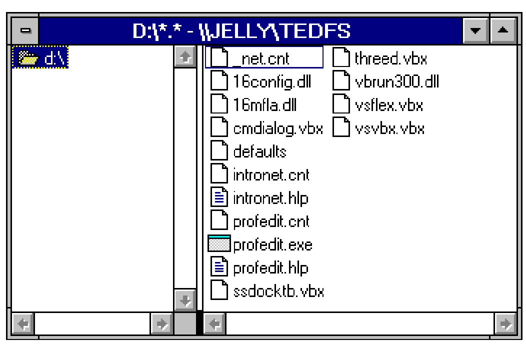
1. Log into Windows as the TEDFS Administrator.
2. In File Manager, click Connect Network Drive on the Disk menu.
3. Choose an unused drive letter.



4. In the Path box, type *server*\TEDFS, replacing *server* with the name of your TEDFS server:
5. Click OK.



6. If prompted, type the TEDFS Administrator's UNIX password.
7. Double-click profedit.exe.



Tip – Create a Program Manager icon for the Profile Editor.

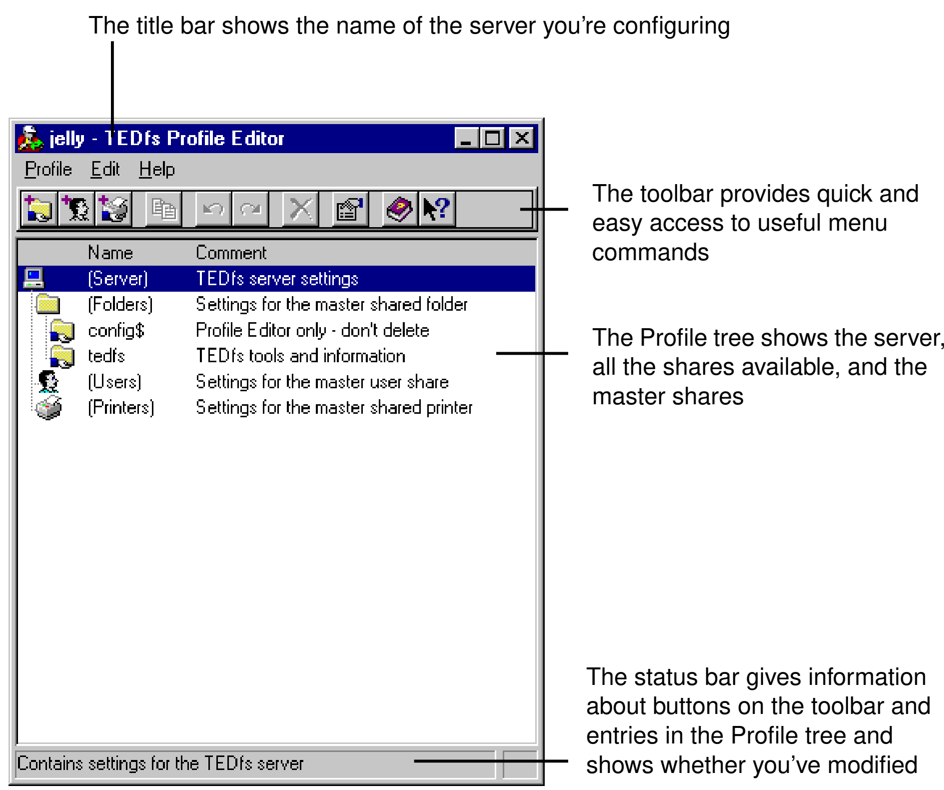
In place of step 4, you can also click Browse, and browse the workgroups for your TEDFS server. When you've found the server, click it, then click TEDFS.

If Windows reports an error when you click the server, or doesn't list any shares, type the path as shown in step 4. You'll need to do this if your UNIX password is different to your Windows password.

Note – This is a Windows problem, beyond the control of TEDFS.

The Profile Editor window

The first time you run the TEDFS Profile Editor, you'll see a window like this:



See Also – "Main window" in the Help index.

Toolbar

To see a brief description of each button in the toolbar, rest the pointer over the button.

Profile tree

The Profile tree gives an overview of your server's configuration. Some parts of the tree are always shown—the entries with names in parentheses (). Other parts depend on the shares you have on your particular server.

At the top of the tree is the server you're configuring. Below that are three groups: *shared folders*, *user shares* and *shared printers*. Each group has a *master share*, handy for more advanced use. We'll explain more about the different types of share later in this book.

See Also – "Master shares","User shares," and "Shared printers," in Chapter 2, "Beyond the Basics".

The first time you run the Profile Editor, you'll see two shared folders in the Profile tree: config$ and TEDFS . You've already used the TEDFS to access the Profile Editor. The config$ share is a special share used by the Profile Editor (Windows doesn't show this share). Both shares are created by TEDFS Setup.

When you add and remove shares, entries appear and disappear in the Profile tree.

Changing settings

The Profile tree is the main starting point for configuring your server. From the Profile tree you can access all settings for your shares and the server.

▼ *To change settings for a share or the server*

1. In the Profile tree, click the entry you want to change the settings for.
2. On the Edit menu, click Properties.

   Note – You cand ouble-click an entry in the Profile tree to show its Properties.

   ---

3. View or change the settings you want. Click the tabs to show all the options you can set.

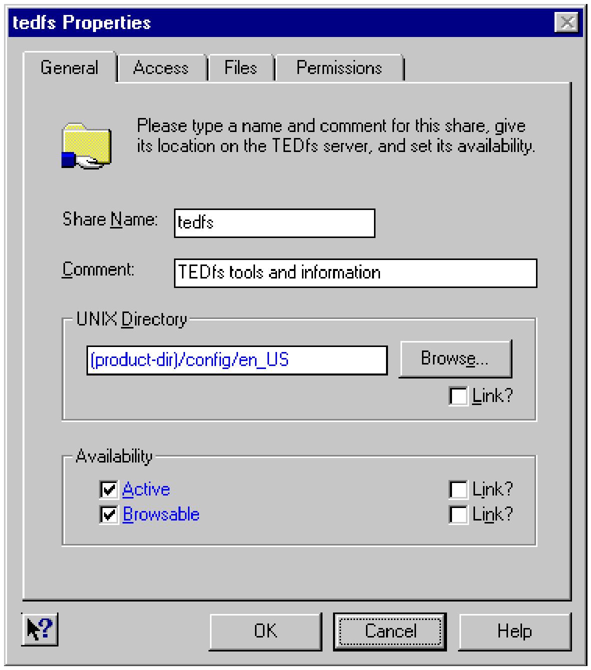
   Some settings let you include placeholders, like (*share-name*) and (*admin-user*), to refer to values that aren't constant.

   See Also – "Placeholders" in the Help index.

   ---

4. When you've finished, click OK to keep any changes you've made. Or click Cancel to close the dialog without making any changes.

When you change a setting in the Profile Editor, it doesn't take effect immediately. An icon in the lower-right corner of the Profile Editor indicates that you've made a change the server doesn't know about yet.



Types of change

There are three types of change you can make:

- A change that affects everybody immediately. For example, creating a new share.
- A change that won't affect people who are already using a share, but will affect new users. For example, changing who can access a share.
- A change that won't affect anybody, unless you restart the server. For example, renaming the server.

For this reason, the Profile Modified icon has three forms:

- A transparent background for changes that can take effect immediately.
- A green background if some changes won't affect users who are already connected to the server.
- A red background if you must restart the server for all changes to take effect.

See Also – "Main window" in the Help index.

Making changes permanent

When you're happy with the changes you've made, you update the TEDFS server with the new profile.

▼ *To update the server with a modified profile*

♦ On the Profile menu, click Update Server.

Depending on the changes you've made, the Profile Editor may offer to restart the server for you. Restarting the server means the Profile Editor must close down.

When you've updated the server, the Profile Modified icon will disappear.

Undoing and redoing changes

If you change some settings by mistake, you can undo those changes easily as long as you haven't updated the server in the meantime. The Profile Editor remembers your last twenty changes.

As long as you don't make any new changes, you can also redo changes you've just undone.

▼ *To undo a change*

♦ On the Edit menu, click Undo. Or click the Undo button on the toolbar.

▼ *To redo a change*

♦ On the Edit menu, click Redo. Or click the Redo button on the toolbar.

When you update the server or exit the Profile Editor, the undo and redo buffers are cleared.

Exiting the Profile Editor

You can exit the Profile Editor at any time.

▼ *To exit the Profile Editor*

♦ On the Profile menu, click Exit.

If you've made any changes to the profile, you'll be asked whether you want to update the server. Click Yes to keep the changes, No to forget them, or Cancel to change your mind and stay working in the Profile Editor.

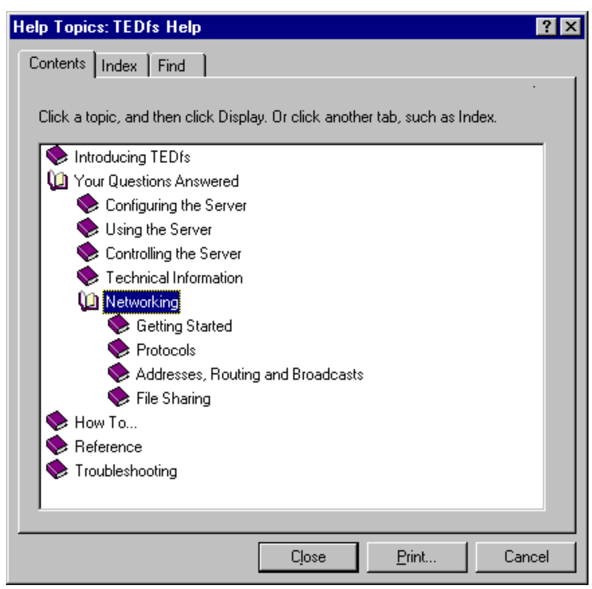
Getting Help

Online Help is the main source of information about TEDFS. All Help is stored with the Profile Editor, in the TEDFS, in Windows Help format. See your Windows manuals for full instructions on using Help.

▼ *To get Help*

♦ In the Profile Editor, click Help Topics on the Help menu. Or click the Help button on the toolbar.

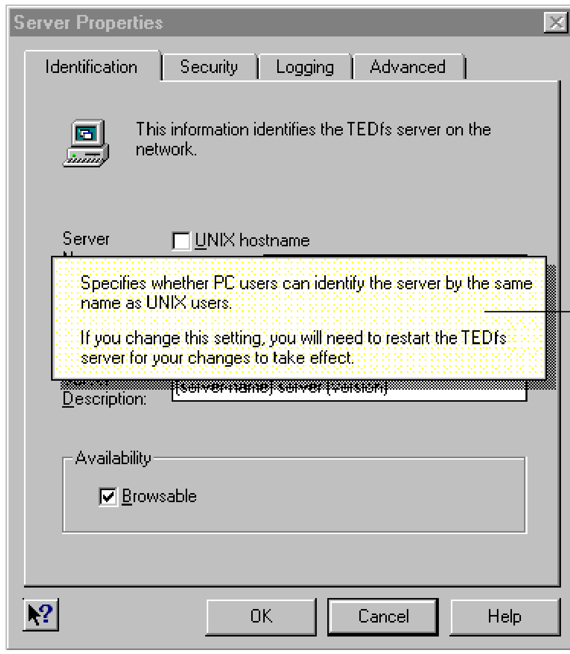
The list of Help topics appears.



- If you use Windows 95 or NT 4, you can use the tabs in Help to search for information in several ways.
- If you use Windows for Workgroups or NT 3.51, you can use the Index to search for information.

▼ *To get Help on a specific item*

♦ For information about an item in the Profile Editor, click and then click the item.



A pop-up explanation appears. Click it to make it disappear

Manipulating shared folders

In this section you'll learn the basics of share management: creating, configuring and deleting shares, accessing them, and how to interpret the files you see in a share.

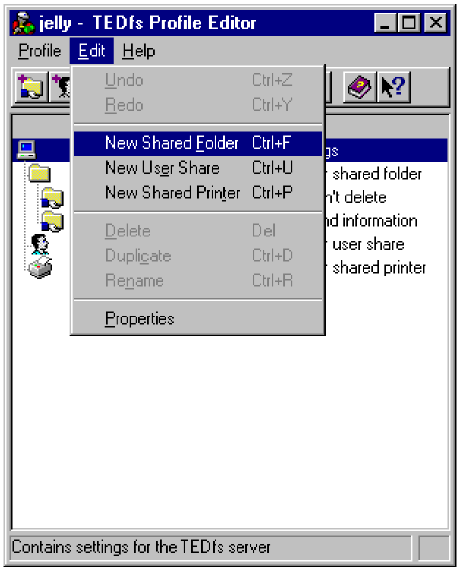
Creating a shared folder

Now you'll use the TEDFS Profile Editor to create your first share—a shared folder, giving access to a UNIX directory. First we'll take you through the process step by step, pointing out how the Profile Editor can help you before you make a mistake. Finally, we'll summarize the procedure.

Starting off

Start the Profile Editor, as described earlier in this chapter.

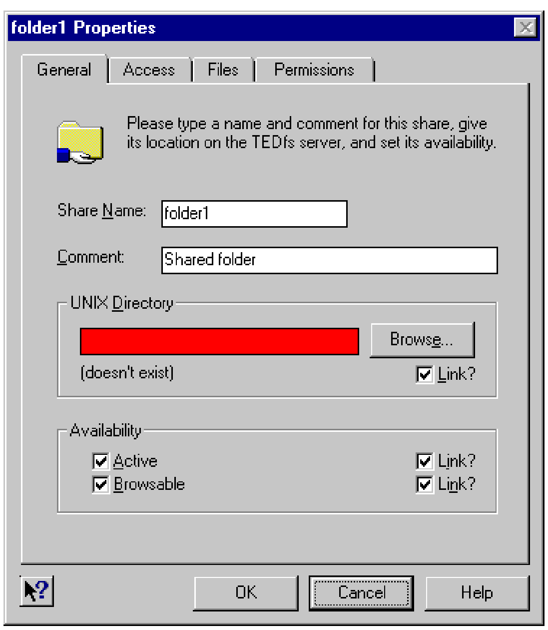
On the Edit menu, click New Shared Folder. Or click the New Shared Folder button on the toolbar.



In the Profile tree, a shared folder called `folder1` appears. The Profile Editor shows properties for the shared folder automatically.

The General tab

Before making any changes, take a look at the General tab.



The first thing you might spot is the UNIX Directory box in red. This is the Profile Editor's way of indicating a problem with a setting—in this case, there's no directory in the text box. Before you can add the share, you must fill in a directory name.

Also, below the text box there's a line saying "(doesn't exist)". This is because the current contents of the box doesn't correspond to a UNIX directory that exists. As you type a directory, the Profile Editor will check silently and report whether it exists.

Many different settings give the same helpful indication and feedback:

- If you see a setting turn red and you don't know why, click OK and the Profile Editor will explain how to fix the problem.
- If you see a comment in parentheses beside a setting, it's the Profile Editor giving instant feedback. It doesn't mean the setting's invalid; just that you might want to think twice before using it.

Filling in the details

First, we'll name the shared folder. Next to Share Name, type temp. People use this name to access the share, so you'll probably want to make the name descriptive. Share names are case-insensitive: the Profile Editor will change any upper-case characters you type to lower-case.

Notice how, as you delete the text `folder1`, the box turns red—this is to remind you that shares must have a name.

Next to Comment, type `Temporary files`. Windows shows comments in browse lists, and when you use Details view in folders.

See Also – "Browse lists," in the Help index.

Next to UNIX Directory, type `/tmp`. This is the UNIX directory you're giving people access to. As you type the first character, the box stops being red to show the setting's valid.

Also, you'll see that as you type each character, the "(doesn't exist)" line appears or disappears. Assuming there's a `/tmp` directory on your UNIX server, the feedback line should disappear when you finish typing.

You can type directories that don't exist, but you'll need to create the directory on UNIX before people can see and access the shared folder.

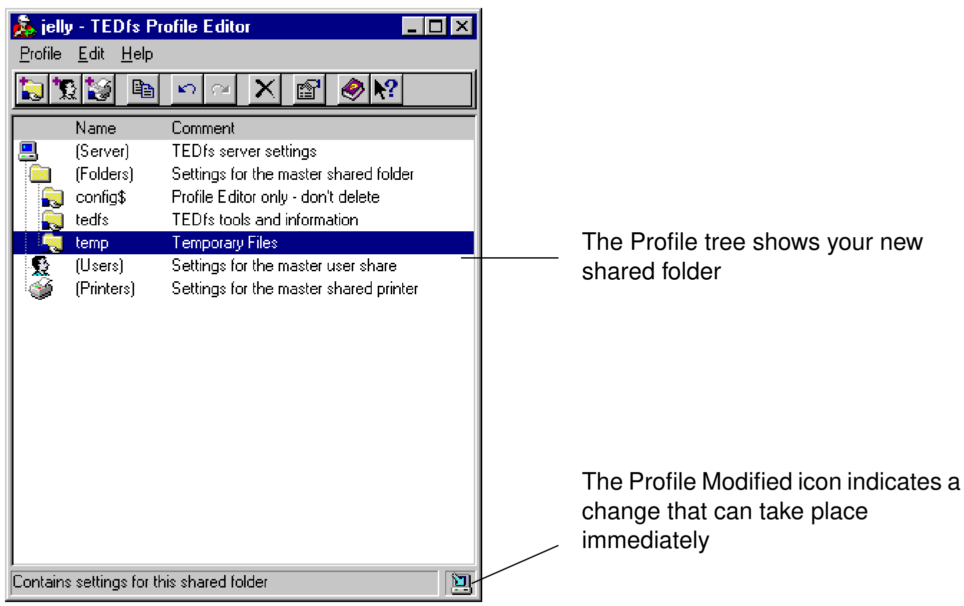
Lastly, notice how the directory you've just typed appears in blue, and the Link box clears. We'll explain blue settings and links in Chapter 2, "Beyond the Basics". For now, don't worry—these aren't errors.

You don't have to remember and type in directory names. If you want, you can click Browse and search for a UNIX directory. Bear in mind that if the UNIX server uses NFS to mount remote UNIX directories in /, it may take a few moments for a directory listing of / to appear.

See Also – "Browsing UNIX directories" in the Help index.

For normal shared folders, you can leave all the other settings as their defaults. Click OK to use these settings.

Making the shared folder available



At the moment, the server doesn't know about the new shared folder. To let people use it, click Update Server on the Profile menu. The Profile Modified icon will disappear.

Summary

In summary, here's how you create a new shared folder.

▼ *To create a new shared folder*

1. In the Profile Editor, click New Shared Folder on the Edit menu. Or click the New Shared Folder button on the toolbar.

   Your new share is shown in the Profile tree, and the shared folder properties dialog is displayed.

2. Type a name, comment and UNIX directory for the share.

3. Change any other properties you want.
4. Click OK.
5. On the Profile menu, click Update Server.

Accessing a shared folder

Now you've created a shared folder, you'll want to make sure you can use it. Accessing a shared folder you've created is very similar to accessing the TEDFS share, described earlier.

See Also – "Finding and starting the Profile Editor," earlier in this chapter.

Windows users will recognize the steps—they're exactly the same steps you use to access shares on other Windows PCs. In fact, any method you use to access other Windows PCs can be used to access a TEDFS server.

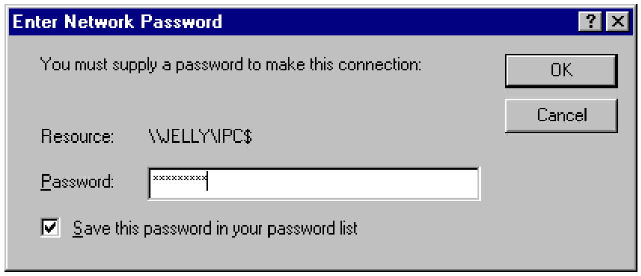
Follow the instructions below for Network Neighborhood or File Manager according to your version of Windows, as before.

▼ *To access a shared folder using Network Neighborhood*

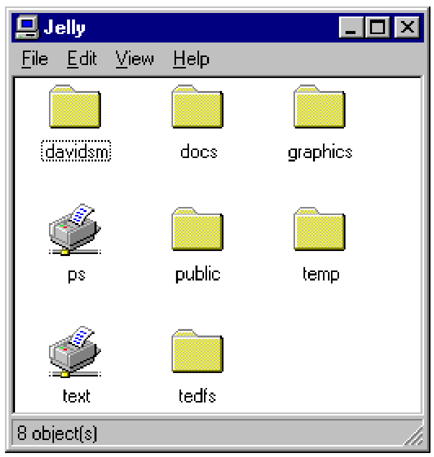
1. Double-click Network Neighborhood and locate the TEDFS server. If it's not in your workgroup, double-click Entire Network.

2. Double-click the TEDFS server.

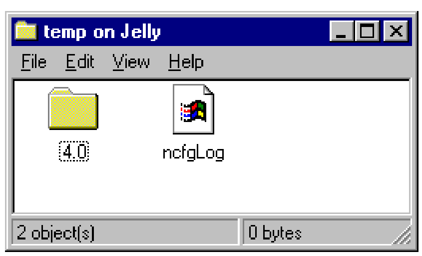
3. If prompted, type your UNIX password.



4. Double-click the shared folder.



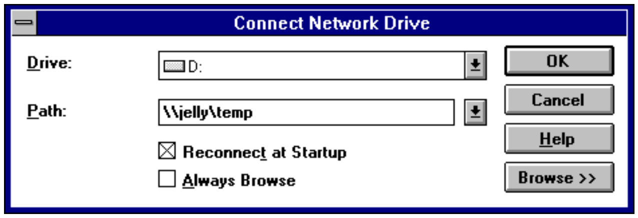
If you're allowed access to the share, you'll see the files and directories it contains.



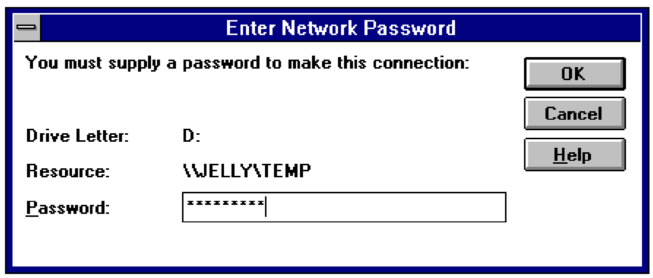
Tip – Use shortcuts to give access to frequently used shares, and files within shares.

▼ *To access a shared folder using File Manager*

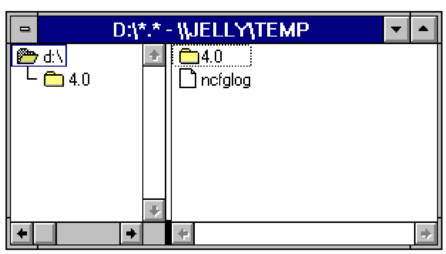
1. In File Manager, click Connect Network Drive on the Disk menu.
2. Choose an unused drive letter.



3. In the Path box, type *server**share*, replacing *server* with the name of your TEDFS server, and *share* with the name of the shared folder.
4. Click OK.
5. If prompted, type your UNIX password.



If you're allowed access to the share, you'll see the files and directories it contains.



In place of step 3, you can also click Browse, and browse the workgroups for your TEDFS server. When you've found the server, click it, then click the shared folder you want to access.

If Windows reports an error when you click the server, or doesn't list any shares, type the path as shown in step 3. You'll need to do this if your UNIX password is different to your Windows password.

Note – This is a Windows problem, beyond the control of TEDFS.

Windows and UNIX filenames

One important difference between Windows and UNIX concerns filenames.

- In Windows, filenames are case-insensitive. For example, Windows treats foo and FOO as identical names.
- On UNIX, filenames are case-sensitive. You can have files called foo and FOO in the same directory.

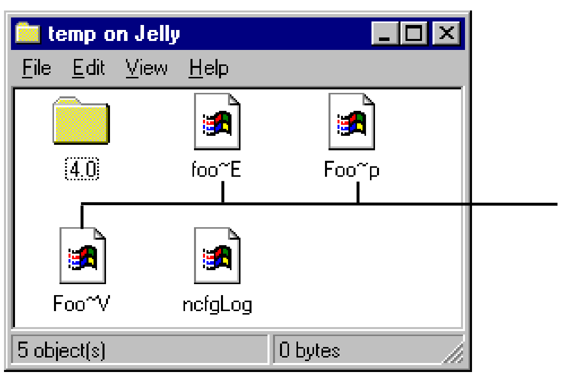
See Also – “PC and UNIX file differences,” in Chapter 4, “Issues for Administrators”.

Also, Windows for Workgroups and NT 3.51 filenames are restricted to “8.3” format—up to eight alphanumeric characters, optionally followed by a dot and an extension of up to 3 alphanumeric characters. UNIX doesn’t have this restriction.

Windows 95 and NT 4 allow filenames of any length, but they’re still case-insensitive.

To ensure that all UNIX files in a directory can be accessed from Windows, the TEDFS server reports different names for some files—the files that Windows wouldn’t be able to tell apart.

Using your newly-created shared folder, you can experiment with filenames to see how TEDFS reports different names. In `/tmp` on your TEDFS server, create three files called `foo`, `FOO` and `Foo`. Then, on Windows, look at that directory using your temp shared folder. If you use Windows 95 or NT 4, you’ll see something similar to the picture below.



The files are reported with different names, formed by adding a unique suffix. The real UNIX filenames haven’t changed

Who's allowed to access a shared folder?

By default, anyone with an account on the UNIX server is given full access to the files and directories in the shared folder, and all actions are performed using their UNIX username.

This means that when a user tries to manipulate files or directories in any way, for example deleting a file, the result would be the same as if they'd performed the action from the UNIX command prompt. If UNIX would deny permission, so will TEDFS.

See Also – "Controlling access," in Chapter 2, "Beyond the Basics".

Deleting a share

Now you'll delete your temp share. Deleting a share doesn't delete any UNIX files or directories—just the fact that people can access those directories from PCs.

▼ *To delete a share*

1. In the Profile tree, click the share you want to delete.
2. On the Edit menu, click Delete. Or click the Delete button on the toolbar.
3. On the Profile menu, click Update Server.

   If you want to disconnect people using this share at the moment, restart the server when the Profile Editor offers.

Note – Don't delete the config$ or TEDFS shares, or configure them so that the TEDFS Administrator is denied access. If you do so, you won't be able to run the Profile Editor. If this happens by accident, run TEDFS Setup to fix your profile.

See Also – "Troubleshooting," in Chapter 4, "Issues for Administrators".

Other share settings

So far, we've only introduced three of the settings you can change for shared folders. If you've explored all the tabs of a properties dialog, you'll know there are plenty of other settings. In this section, we'll introduce two of the most useful.

Enabling and disabling

Sometimes it's useful to be able to take a share out of action temporarily. For instance, you might want to move some files and be sure nobody can access them during that time.

▼ *To disable a share temporarily*

1. In the Profile tree, double-click the share you want to disable.
2. On the General tab, make sure the Active box is not checked.
3. Click OK.
4. On the Profile menu, click Update Server.

Tip – Check the Active box to enable the share again later.

Hiding a share

It's possible to set up shares that don't appear when you double-click a computer in Network Neighborhood (or click the computer in File Manager's Connect Network Drive dialog). To access these hidden shares, you must know the name and type it yourself.

▼ *To hide a share*

1. In the Profile tree, double-click the share you want to hide.
2. On the General tab, make sure the Browsable box is not checked.
3. Click OK.
4. On the Profile menu, click Update Server.

Tip – Check the Browsable box to show the share again later.

Beyond the Basics 2

Look in this chapter when you've mastered the basics, and want to know more about how to configure your TEDFS server.

You'll discover some of the more sophisticated features of the TEDFS Profile Editor and server, including the different access rights you can grant to users.

Modifying the server

In this section you'll learn how to make changes to the TEDFS server itself, including changing its name and workgroup, and hiding the server. You'll also find out how to change the Administrator's username.

Identifying the TEDFS server on the network

Your UNIX server already has different sorts of names, usually (but not limited to) a hostname, an IP address, and a DNS name. It may also have multiple DNS names or multiple IP addresses.

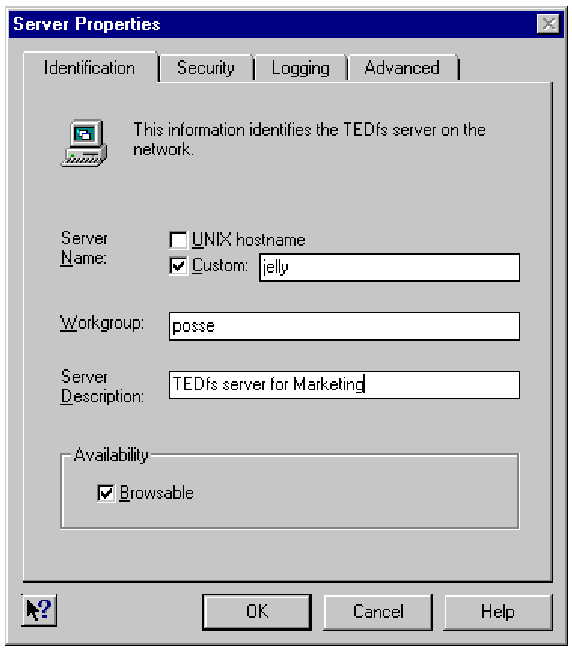
For Windows networking, the TEDFS server uses another name (technically, a NetBIOS name). Other PC users on the network use this name to refer to the server. In fact, you're not restricted to a single name—you can use as many as you wish.

Along with the name, you can provide a comment. Users will see this comment when they look at the TEDFS server on the network.

Computers are grouped into loose collections called *workgroups*. Workgroups don't have a strict hierarchy, and computers can appear in more than one workgroup.

▼ *To identify the TEDFS server on the network*

1. In the Profile Editor, double-click (*Server*) in the Profile tree.
2. Click the Identification tab.



3. Set up the names for your TEDFS server:
   - Check UNIX Hostname to use your UNIX server's hostname.
   - Check Custom and type a name to use that name.

Tip – To use more than one custom name, separate each name with a comma (,).

---

4. Type the workgroup name (or names), and a brief description of your TEDFS server.
5. Click OK, then click Update Server on the Profile menu. You'll need to restart the server for the new names, workgroups and description to take effect.

How are workgroups maintained?

Each workgroup is self-organizing: it automatically elects one of its members to maintain the list of computers in the workgroup. This computer is called the *master browser*.

When you choose to list the computers in your workgroup, your computer contacts the master browser for the details.

If some details change, the master browser may take some time to fully reflect the new information. For example, if you rename your TEDFS server the master browser will show the new name immediately, but the old name may still be displayed for a time.

Hiding the server

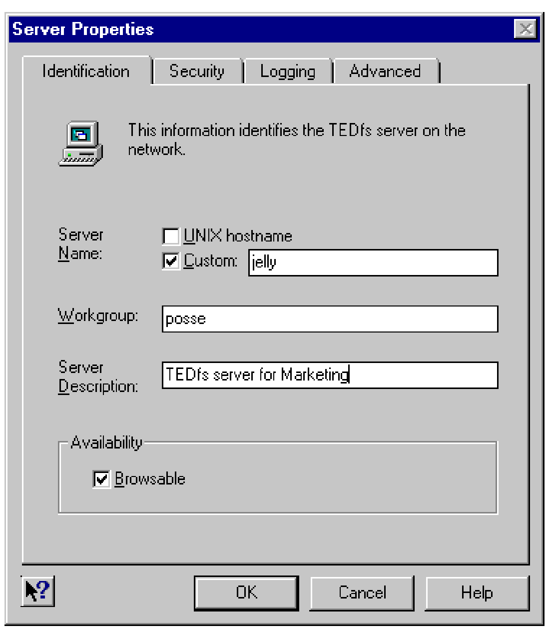
If you want, you can stop the TEDFS server from appearing in any workgroups. Users can still access the server, but they'll need to know the name to do so.

Due to the way workgroups are maintained, it may take some time for the server to disappear from the workgroup.

See Also – "How are workgroups maintained?," earlier in this chapter.

▼ *To hide the server from all workgroups*

1. In the Profile Editor, double-click (*Server*) in the Profile tree.
2. Click the Identification tab.



3. Make sure that Browsable isn't checked.

4. Click OK, then click Update Server on the Profile menu. You'll need to restart the server for the change to take effect.

Tip – Check the Browsable box to show the server again later.

Changing the TEDFS Administrator

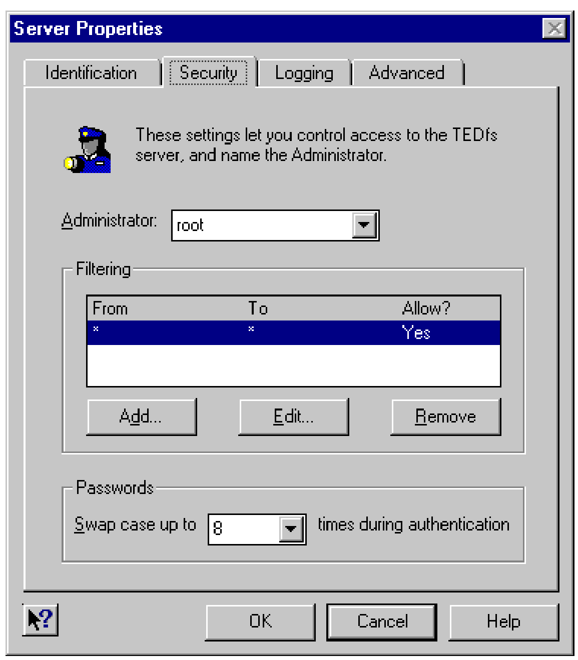
The Profile Editor lets you grant complete access to any files and directories on the UNIX server. For this reason, only one user—the TEDFS Administrator—is allowed to run the Profile Editor. The TEDFS Administrator is as powerful as the UNIX superuser.

Think carefully before changing the Administrator. Once the server has been updated, only the new Administrator can run the Profile Editor. Remember that you need to log in to Windows as the Administrator to run the Profile Editor; choosing root as the Administrator may not be convenient.

▼ *To change the TEDFS Administrator*

1. In the Profile Editor, double-click (*Server*) in the Profile tree.

2. Click the Security tab.

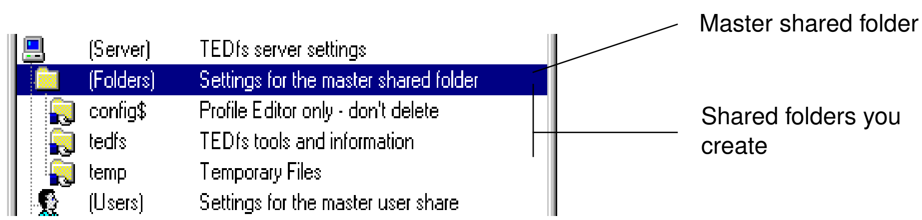


3. Click the new Administrator's username in the list, or type the name yourself. If you type a username that doesn't exist on the UNIX server, you'll see "(doesn't exist)" alongside the box.
4. Click OK, then click Update Server on the Profile menu. You'll need to restart the server for the change to take effect.

Master shares

In Chapter 1, "The Basics", we created and manipulated a shared folder. In this section, we'll introduce master shares, and explain links.

When you created your shared folder, it appeared in the Profile tree alongside the existing shared folders. Just above these folders is the *master shared folder*.



Similarly, there's a master shared printer and a master user share, just above the other types of share in the Profile tree.

See Also – "Shared printers" and "User shares," later in this chapter.

These *master shares* contain the master settings for that type of share. Master shares are only shown by the TEDFS Profile Editor; you can't see a master share from Windows.

The settings for a master share are used:

- As the default settings when you create a new share of that type.
- When you check "Link?" next to an option in a share's properties.
- As the settings for automatically generated user shares and shared printers.

You configure a master share in exactly the same way as you configure a real share, but you can't delete the master shares.

Links

In the Profile Editor, Link boxes appear next to some options in a share's properties. If a Link box is checked, it means the neighboring option is linked to the setting in the master share.

- Linked settings automatically change when you modify the same setting in the master share.

- Settings that aren't linked appear in blue, so you can easily identify differences from the master share.

When you create a new share, all the settings are automatically linked (except for the name and comment, which don't have Link boxes), so the share's effectively a clone of the master share.

If you change a setting in the new share, the Link box automatically clears to remove the link. The setting becomes independent of the master share. You can restore the link by checking the Link box again.

You may like to think of a particular share's configuration as a set of differences from the master share; alternatively, that a share inherits settings from its master share.

Using master shares and links effectively can help you minimize the work needed to change a setting in lots of shares simultaneously.

See Also – "Using links effectively," in Chapter 3, "The Possibilities".

Shared printers

We've already shown how TEDFS lets Windows users access UNIX files and directories just as if they were on another PC on the network. It's the same with UNIX printers.

In this section we'll explain how TEDFS can create shared printers automatically, saving you valuable time and effort, and how to create your own shared printers.

Automatic shared printers

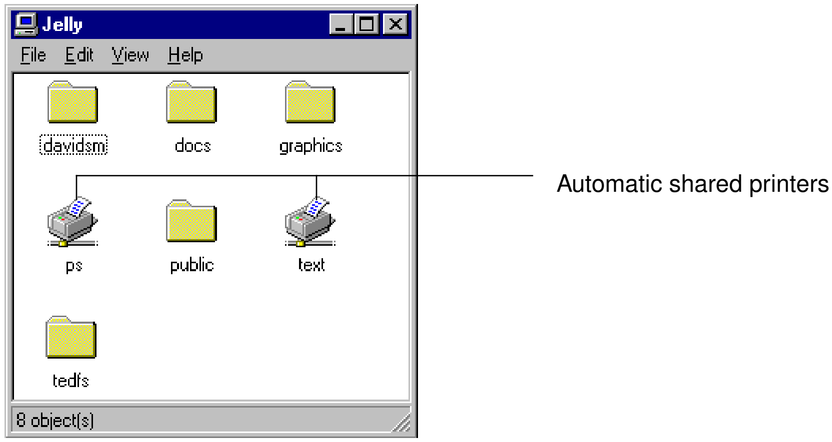
When the TEDFS server starts, it scans the UNIX server for printers, and updates your profile with information about each printer.

Note – This means if you add a UNIX printer, you'll need to restart the TEDFS server for TEDFS to include the information in your profile.

In Windows, when you list the shares on a TEDFS server you see a shared printer for each UNIX printer (if automatic shared printers are enabled, which they are by default). The share names are the same as the UNIX printer names

---

See Also – "Enabling and disabling automatic shared printers," later in this chapter.



The TEDFS server uses the settings from the master shared printer for automatic shared printers. However, you can override settings for automatic shared printers, if you want.

As automatic shared printers are normally just “clones” of the master shared printer, the Profile Editor doesn’t show them.

See Also – “Configuring shared printers,” later in this chapter.

Creating shared printers

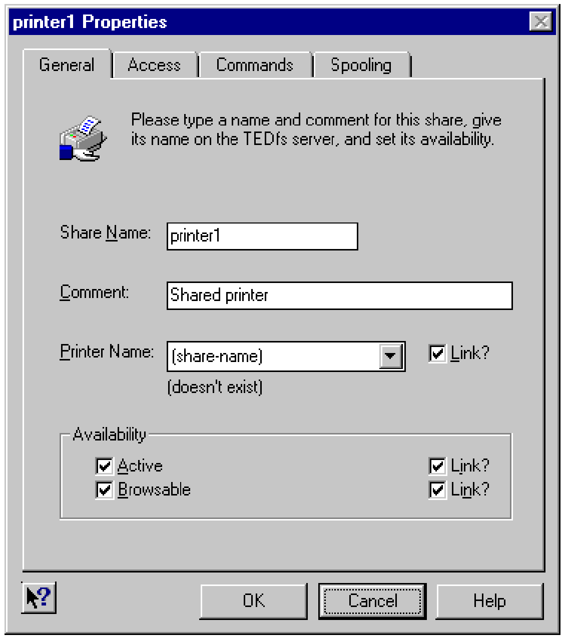
You’re not restricted to the automatic shared printers. If you want, you can create new shared printers that, for example, print (or don’t print) a banner page. Then users can choose a shared printer based on what kind of output they’d like.

See Also – “Using shared printers for custom output,” in Chapter 3, “The Possibilities”.

To create a new shared printer

1. In the Profile Editor, click New Shared Printer on the Edit menu. Or click the New Shared Printer button on the toolbar.

   Your new share is shown in the Profile tree, and the shared printer properties dialog is displayed.



2. Type a name and comment for the share.

3. Click a system printer name in the list, type a printer name, or click the placeholder (*share-name*) to use what's in the Share Name box.

   You'll see "(doesn't exist)" below the system printer name if there's no UNIX printer of that name.

See Also – "Placeholders" in the Help index.

---

4. Change any other properties you want.

5. Click OK, then click Update Server on the Profile menu.

Configuring shared printers

You can configure automatic shared printers and your own shared printers independently. To configure the settings used for all automatic shared printers, you change the master shared printer's properties.

▼ *To configure shared printers*

1. In the Profile Editor, double-click the shared printer you want to configure, or (*Printers*) to configure automatic shared printers.
2. Change any properties you want.
3. Click OK, then click Update Server on the Profile menu.

Customizing one of the automatic shared printers

For automatic shared printers, TEDFS uses the settings in the master shared printer only if there isn't a shared printer with the same name.

To set up special behavior for a particular printer, just create and configure a shared printer using the correct share name, and TEDFS will use those settings instead.

See Also – "Using shared printers for custom output," in Chapter 3, "The Possibilities".

Customizing printing commands

Different flavors of UNIX use varying methods for submitting, listing and removing print jobs. You don't need to worry about this—TEDFS detects which print commands are appropriate for your system when you run TEDFS Setup.

However, you can customize the print commands for your site, for instance to print a banner page before each print job, using the Commands tab of a shared printer's (or the master shared printer's) properties.

See Also – "Using shared printers for custom output," in Chapter 3, "The Possibilities".

Using a shared printer

Before you can print from Windows, you need to set up a network printer, just as you would to use a printer on another PC. How you do this depends on your version of Windows:

- Windows 95 and NT 4 users can use the Add Printer wizard.
- Windows for Workgroups and NT 3.51 users can use Print Manager.

See Also – Your Windows manual.

▼ *To set up a network printer in Windows 95 or NT 4*

1. Click the Start button, point to Settings, and then click Printers.

   Tip – Double-click a shared printer in Windows to install it if it's not already installed.

   ---

2. Double-click Add Printer.
3. Follow the instructions on your screen.

   Choose Network Printer when asked, and enter the name of the server and shared printer, in the form `\\server\printer` (or click Browse, if you prefer). You'll need to know the make and model of the printer.

▼ *To set up a network printer in Windows for Workgroups or NT 3.51*

1. In Print Manager, click Connect Network Printer on the Printer menu.
2. Choose an unused device name.
3. If you don't see a list of computers, click Browse. Locate the TEDFS server. If it's not in your workgroup, double-click the workgroup it's in.
4. Click the TEDFS server, then click the shared printer.
5. Click OK, then click Yes to add a new printer.
6. Use the Printers dialog to install files for the correct make and model of printer.

Once you've set up a network printer for the TEDFS shared printer, you can print to it from your Windows programs. Follow the particular method for each program.

Who's allowed to access a shared printer?

By default, anyone with an account on the UNIX server is allowed to print to the shared printer, and all actions are performed using their UNIX username.

This means that when a user tries to add, list or remove jobs, the result would be the same as if they'd performed the action from the UNIX command prompt. If UNIX would deny permission, so will TEDFS.

See Also – "Controlling access," later in this chapter.

Manipulating shared printers

You can disable, hide, and delete shared printers in the same way as shared folders.

You can't delete the master shared printer.

See Also – "Deleting a share," "Enabling and disabling" and "Hiding a share," in Chapter 1, "The Basics".

Enabling and disabling automatic shared printers

If you don't want TEDFS to generate shared printers automatically, you can easily prevent it from doing so.

▼ *To turn off automatic shared printers*

1. In the Profile Editor, double-click (*Printers*) in the Profile tree.

   You'll see properties for the master shared printer.

2. On the General tab, make sure the Active box is not checked.
3. Click OK, then click Update Server on the Profile menu.

Tip – Check the Active box to turn on automatic shared printers again later.

Turning off automatic shared printers also turns off any shared printers you've created that link the Active setting. You can make a shared printer accessible whether or not automatic shared printers are turned on, by checking the Active box and clearing the adjacent Link box in that shared printer's properties.

User shares

In this section, we'll explain how Windows users can access their UNIX home directories automatically, and how you can customize access for particular users with user shares.

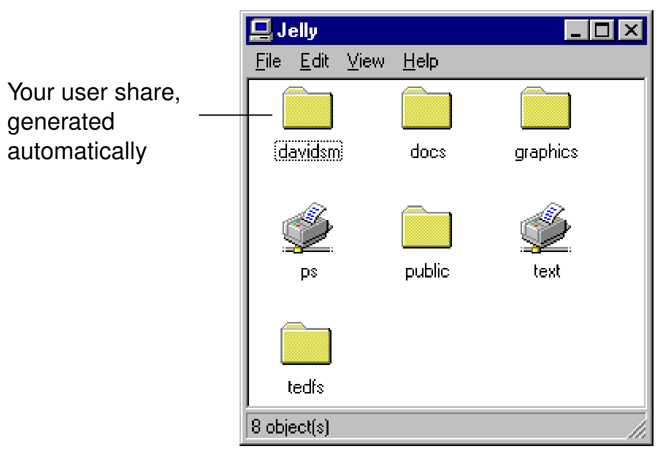
Automatic user shares

Automatic user shares let Windows users with valid UNIX accounts access their home directories.

In Windows, when you list the shares on a TEDFS server you see a share with your username—as long as you use the same Windows username as your UNIX username (and automatic user shares are enabled, which they are by default). Accessing your home directory through your user share is as simple as accessing any other shared folder.

See Also – "Enabling and disabling automatic user shares," later in this chapter and "Accessing a shared folder," in Chapter 1, "The Basics".

You'll only ever see one user share in share lists—for yourself.



The TEDFS server uses the settings from the master user share for automatic user shares. However, you can override settings for automatic user shares, if you want.

See Also – "Configuring user shares," later in this chapter.

As automatic user shares are normally just "clones" of the master user share, the Profile Editor doesn't show them.

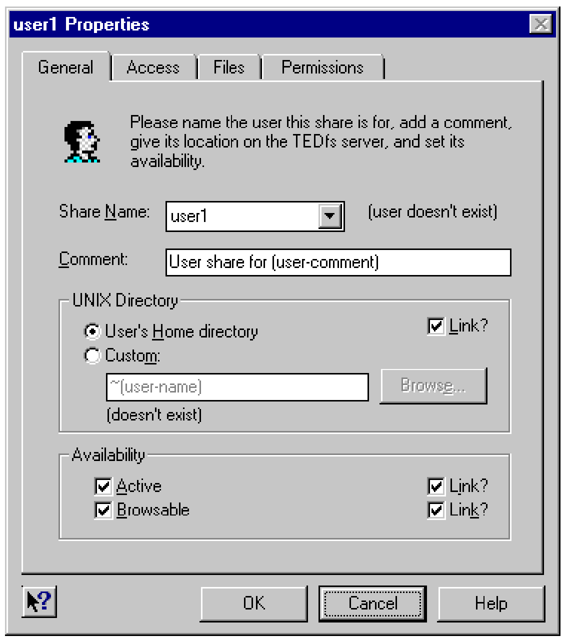
Creating user shares

You're not restricted to the automatic user shares. If you want, you can create new user shares, for example to customize access for a particular user.

▼ *To create a new user share*

1. In the Profile Editor, click New User Share on the Edit menu. Or click the New User Share button on the toolbar.

   Your new share is shown in the Profile tree, and the user share properties dialog is displayed.



2. Click a username in the list, or type a username.

   You'll see "(user doesn't exist)" next to the username if there's no UNIX user of that name.

3. Type a comment for this user. To include the user's full name, use the placeholder (*user-comment*).

4. Do one of the following:
   - To give access to the user's home directory, click User's Home Directory.
   - To give access to a custom directory, click Custom and type the directory. You can include the username as part of the directory, using the placeholder (*user-name*).

See Also – "Placeholders" in the Help index.

---

5. Change any other properties you want.
6. Click OK, then click Update Server on the Profile menu.

Configuring user shares

You can configure automatic user shares and user shares you've created independently. To configure the settings used for all automatic user shares, you change the master user share's properties.

▼ *To configure user shares*

1. In the Profile Editor, double-click the user share you want to configure, or (*Users*) to configure automatic user shares.
2. Change any properties you want.
3. Click OK, then click Update Server on the Profile menu.

Customizing one of the automatic user shares

For automatic user shares, TEDFS uses the settings in the master user share only if there isn't a user share with the same name.

To set up special behavior for a particular user, just create and configure a user share using the correct share name, and TEDFS will use those settings instead.

See Also – "Overriding user shares", in Chapter 3, "The Possibilities".

Manipulating user shares

You can disable, hide and delete user shares in the same way as shared folders. In fact, most of the settings for user shares are exactly the same as for shared folders.

You can't delete the master user share.

See Also – "Deleting a share," "Enabling and disabling" and "Hiding a share," in Chapter 1, "The Basics".

Enabling and disabling automatic user shares

If you don't want TEDFS to generate user shares automatically, you can easily prevent it from doing so.

▼ *To turn off automatic user shares*

1. In the Profile Editor, double-click **(Users)** in the Profile tree.

   You'll see properties for the master user share.

2. On the General tab, make sure the Active box is not checked.
3. Click OK, then click Update Server on the Profile menu.

Tip – Check the Active box to turn on automatic user shares again later.

Turning off automatic user shares also turns off any user shares you've created that link the Active setting. You can make a user share accessible whether or not automatic user shares are turned on, by checking the Active box and clearing the adjacent Link box in that user share's properties.

Who's allowed to access a user share?

By default, anyone with an account on the UNIX server is given full access to the files and directories in their user share, and all actions are performed using their UNIX username. All other users are denied access.

This means that when a user tries to manipulate files or directories in any way, for example deleting a file, the result would be the same as if they'd performed the action from the UNIX command prompt. If UNIX would deny permission, so will TEDFS.

See Also – "Controlling access," later in this chapter.

Controlling access

In this section, you'll learn how to use the Profile Editor to customize exactly who can access a shared folder, shared printer or user share. We'll show you how to allow Guest access, and how to deny access.

Understanding access rights

Each share has an associated list of *access rights*. A particular access right describes:

- A username and UNIX group, for whom this access right applies.
- Whether this right applies if they have an account on the UNIX server.
- Whether this right applies if they *don't* have an account on the UNIX server.
- The actions this user is allowed to perform in the share.
- The username and UNIX group to use when performing the allowed actions.

You set up the access rights for a share using the Access tab of the share's properties.

The access rights determine whether or not TEDFS tries to perform an action in a share. If a particular action is allowed, it doesn't necessarily mean the action will succeed: the UNIX permissions ultimately determine success or failure.

For example, if the access rights for a share grant kevin full access, performing actions as user rod, but the UNIX permissions on the directory only allow read-only access for rod, then kevin won't be able to write to the directory.

See Also – "How to tell if an action will succeed," in Chapter 4, "Issues for Administrators".

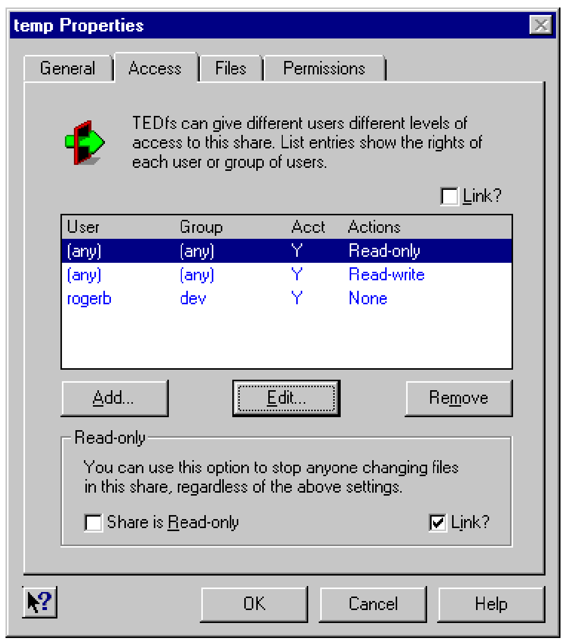
Customizing access

You can customize access to individual shares, or use the master shares to customize access for automatically generated shares, new shares you create, and shares that link the Access settings.

▼ *To customize access to a share*

1. In the Profile Editor, double-click the share or the master share you want to customize access for.

2. Click the Access tab to see the current access rights.



Entries near the top of the list take precedence over those below. To move an entry, drag it up or down.

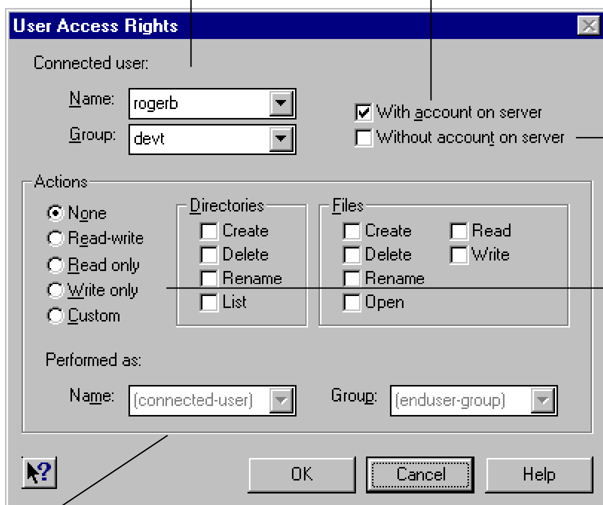
3. Do one of the following:
   - To add a new access right, click Add.
   - To edit an existing access right, click it in the list, then click Edit.
   - To remove an access right, click it in the list, then click Remove.

4. If you're adding or editing an access right, fill in the details in the User Access Rights dialog.

In the top section, set up the circumstances in which this right should apply. Then in the bottom section, set up the actions allowed under those circumstances, and who the actions are to be performed as.

Click or type the user's name and group.
Click **(any)** to mean any user or group.

Check this to mean the right applies if the user has an account on the server.



Check this to mean the right applies if the user doesn't have an account on the server.

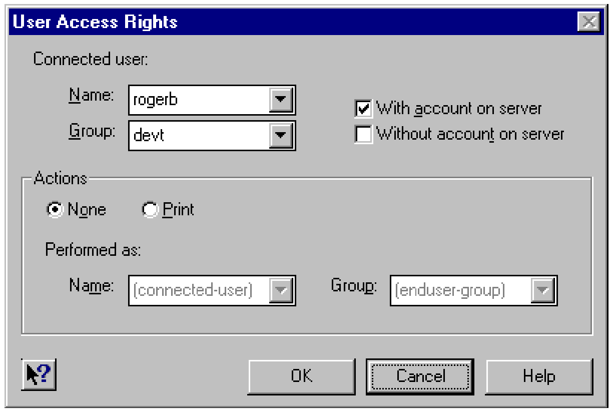
Click the type of actions the user will be allowed to perform. You can customize these with the Directories and Files boxes.

Click or type the UNIX username and group all actions will be performed as.
Click **(connected-user)** and **(enduser-group)** to mean the user accessing the share.

Tip – Check both With and Without Account On Server if you want the access right to apply whether or not the user has an account.

Tip – Use the placeholder (*admin-user*) to refer to the TEDFS Administrator.

The picture above shows the User Access Rights dialog for shared folders and user shares. For shared printers, the dialog looks like this:



Ordering the access rights

The order of access rights in the list on the Access tab is important. When someone tries to use a share, the TEDFS server checks the details against the list. The first entry that matches determines the actions that user is allowed to perform. If no entry matches, the user is not allowed to access the server.

When adding or editing access rights, make sure the entry appears in the correct place in the list: entries for specific users should appear *before* entries for any user.

If an access right for any user appears before one for a specific user, the specific one will be ignored.

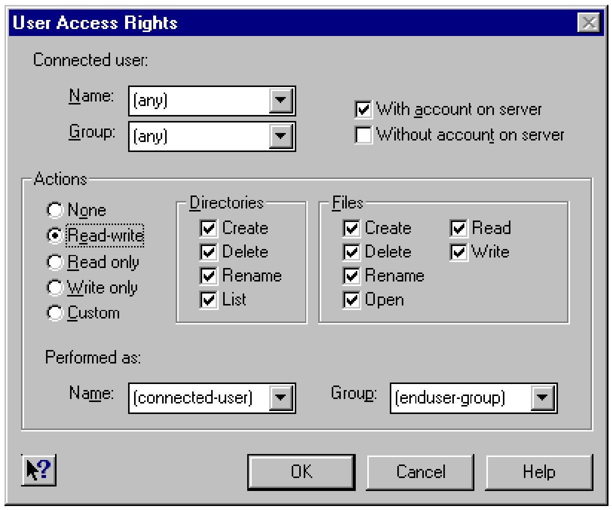
▼ *To move an access right*

♦ Drag the entry up or down in the list.

Common types of access

The following sections give examples of common types of access you’re likely to want to use.

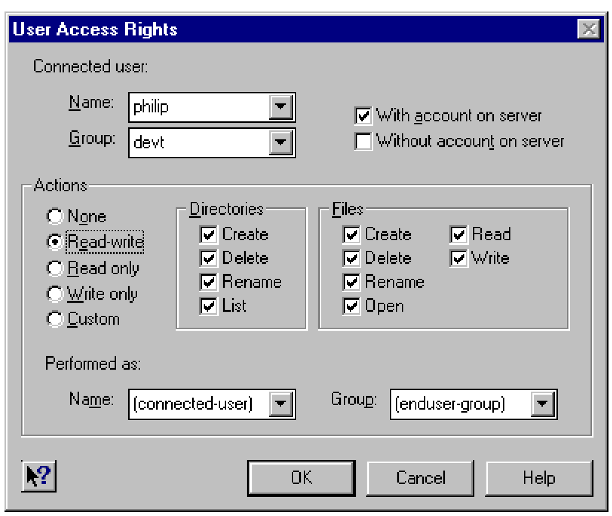
Granting full access for all users with accounts



Allow all users who have an account full access as themselves.

Remember to put this access right below any access rights for particular named users.

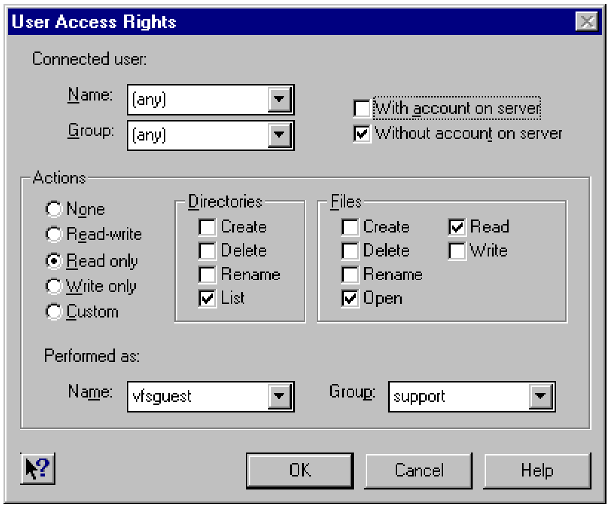
Granting full access for one user



Allow this user who has an account full access as that user.

Remember to put this access right above any access rights for multiple users.

Allowing guest access



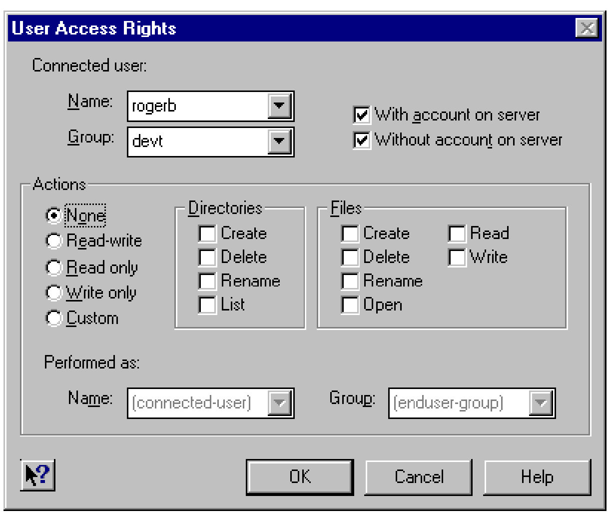
Allow all users who don't have an account read-only access as user "vfsguest".

UNIX actions must always be performed by a valid UNIX user. For guest users—users without an account on the server—you have to specifically name a UNIX user to use for these actions.

You may want to set up a special UNIX account, called for example "vfsguest", to use for guest users.

Remember to put this access right below any access rights for particular named users.

Denying access by one user



Allow this user who might or might not have an account no access.

In this case, you don't need to decide who to perform actions as, as no actions are allowed.

Remember to put this access right above any access rights for multiple users.

More share settings

In this section, we'll briefly cover the remaining settings for shared folders and user shares.

Showing and hiding symbolic links

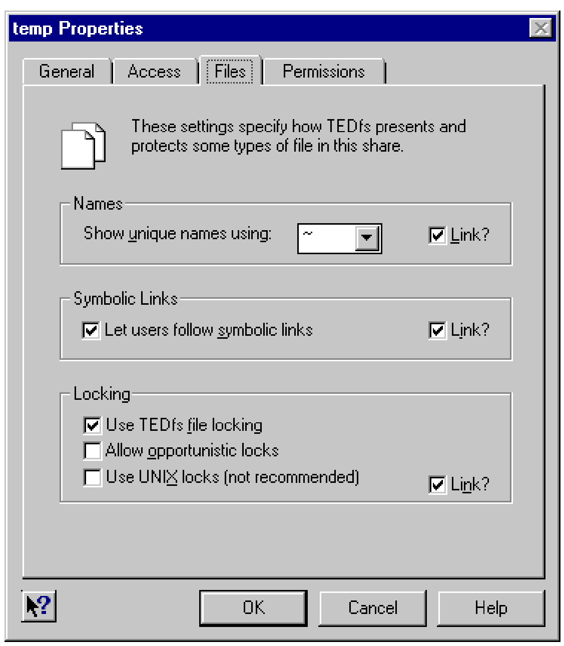
UNIX symbolic links can lead anywhere in the UNIX filesystem. If you're not careful, you could inadvertently grant access to directories you thought were safely hidden from your PC users.

You can configure the TEDFS server to hide all symbolic links in a share, ensuring that PC users are restricted only to the files and directories in that share.

▼ *To hide symbolic links in a share*

1. In the Profile Editor, double-click the share (or master share) in which you want to hide symbolic links.

2. Click the Files tab.



3. Make sure the Let Users Follow Symbolic Links box is not checked.
4. Click OK, then click Update Server on the Profile menu.

Tip – Check the Let Users Follow Symbolic Links box to show symbolic links again later.

Changing the special character for unique names

As mentioned in Chapter 1, “The Basics”, Windows and UNIX treat filenames differently. Windows filenames are case-insensitive, and depending on your version of Windows may be restricted in length. UNIX filenames are case-sensitive, and may be any length.

To ensure that all UNIX files in a directory can be accessed from Windows, the server adds a special character and code to the files that Windows wouldn’t be able to tell apart (truncating to short “8.3” format if necessary).

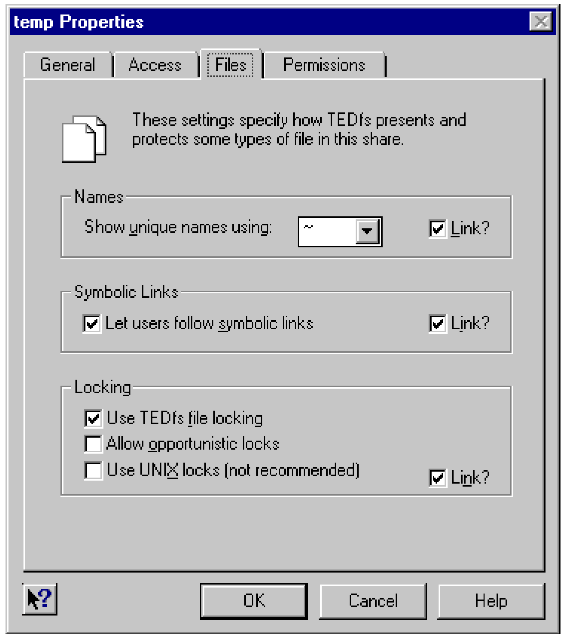
See Also – “PC and UNIX file differences,” in Chapter 4, “Issues for Administrators”.

For each share, or for the master shares, you can change the special character used to introduce the unique code.

▼ *To change the special character for unique names*

1. In the Profile Editor, double-click the share (or master share) to change the special character for.

2. Click the Files tab.



3. In the list, click the character to use for unique names.
4. Click OK, then click Update Server on the Profile menu.

Specifying permissions for new files and directories

When you create new files and directories in a share, they must be given standard UNIX file permissions.

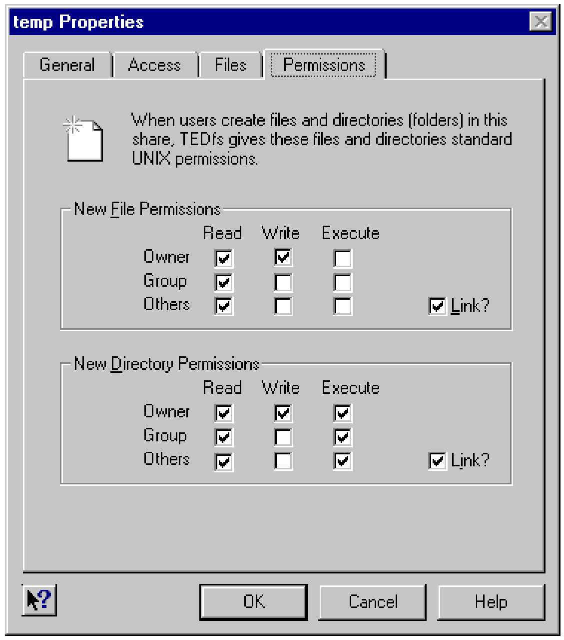
For each share, or for the master shares, you can decide which permissions to give.

Note – These are the permissions you would see from UNIX. A user's umask does not affect these settings.

▼ *To change the permissions for new files and directories*

1. In the Profile Editor, double-click the share (or master share) to change the new file/directory permissions for.

2. Click the Permissions tab.



3. Check the boxes at the intersections of rows and columns to set the appropriate permissions.

4. Click OK, then click Update Server on the Profile menu.

Making a share read-only

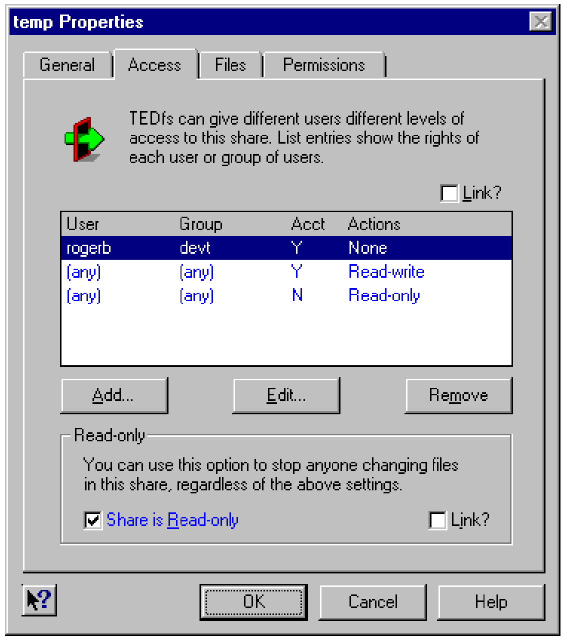
TEDFS access rights give you complete control over who can access shares. This is extremely useful, but sometimes you might want to make a share read-only for everyone—for instance, you might want to make a backup and be sure no files change while it's in progress.

Instead of laboriously changing each access right to read-only, one-by-one, you can change just one setting in the share. You can also make lots of shares read-only at once, if they link the Share Is Read-only setting, by changing the master share.

▼ *To make a share read-only*

1. In the Profile Editor, double-click the share (or master share) you want to make read-only.

2. Click the Access tab.



3. Make sure the Share Is Read-only box is checked.

4. Click OK, then click Update Server on the Profile menu.

   You'll probably want to restart the server, so that all users are affected.

Tip – Clear the Share Is Read-only box to make a share writable again later.

File locking

File locking lets a program be sure it can read and write to a file (or part of a file) without another program doing so at the same time. UNIX and Windows use different locking mechanisms.

The TEDFS server implements full Windows-style locking, but also allows you to use UNIX file locking at the same time, if you wish. However, UNIX file locking is not recommended.

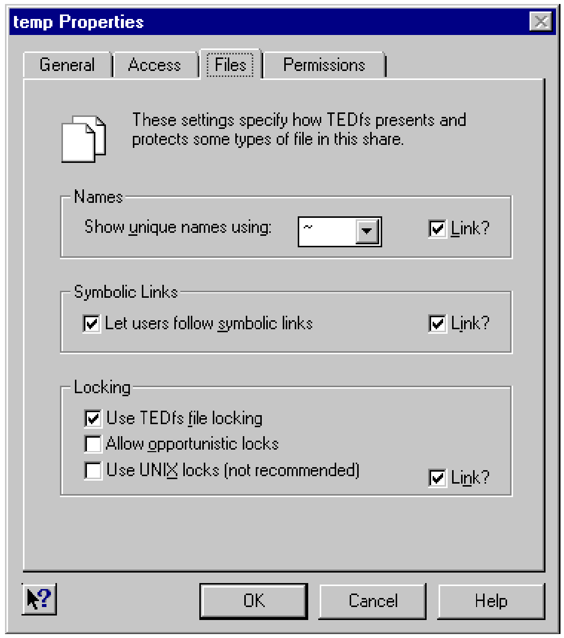
See Also – "File locking," in Chapter 4, "Issues for Administrators".

You can use different locking mechanisms for each share, if you wish, and use the master shares with links to change lots of shares at once.

▼ *To change how TEDFS locks files in a share*

1. In the Profile Editor, double-click the share (or master share) you want to change the locking mechanism for.

2. Click the Files tab



3. Click the locking settings you want. Use Help to find out about each setting.

4. Click OK, then click Update Server on the Profile menu.

The Possibilities 3

Look in this chapter when you're ready to examine the world of possibilities offered by TEDFS.

Using real-world examples, you'll discover just a few of the ways in which you can take advantage of the flexibility of the Profile Editor and server.

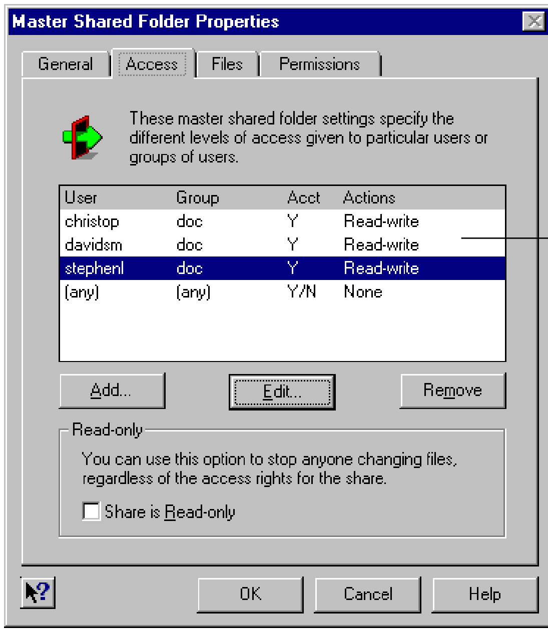
Using links effectively

Links and master shares let you change settings in lots of shares at once. In fact, you'll probably want to leave most of your settings linked to the master share, and only change the settings that make the share different from the master.

If master shares reflect the most common settings for your site, making across-the-board changes is as simple as editing just the master shares.

For example, consider a TEDFS server dedicated to a particular team. In this case, the TEDFS Administrator should set up access rights for the team in the master share, as shown in the picture on the following page.

In this way, when someone joins or leaves the team, the Administrator just adds or removes an access right in the master share, and all linked shares are automatically updated.



The master shared folder's access rights

Also, when the Administrator adds a new share, it automatically has the correct access rights for the team.

As a general rule, treat the shares you create as a collection of differences from the master shares, and use links as a form of "inheritance".

Overriding automatic user shares

Automatic user shares use the settings from the master user share, unless you've created a specific user share.

In other words, creating user shares for specific users lets you override the settings for those users. You only need to override the settings you want to—all the others retain their links to the master share.

So if you want to change just one setting for a particular user, it's as simple as creating a user share for that user, and changing that one setting.

The following sections show a few of the overrides you might want to use.

Show symbolic links for an advanced user

Symbolic links could lead anywhere on the server. You can easily hide symbolic links in users' home directories, by clearing the Let Users Follow Symbolic Links box in the master user share. In this way, you can be sure that all automatically generated user shares will only grant access to subdirectories of home directories.

However, you can trust your advanced users—yourself included! You don't mind if those users can see symbolic links.

To give this custom behavior to your advanced users, all you need to do is create a user share for each one, and check the Let Users Follow Symbolic Links box.

See Also – "PC and UNIX file differences," in Chapter 4, "Issues for Administrators".

Give a user access to another directory

If a particular user wants their user share to access a directory other than their home directory on the UNIX server, create a user share for that user and change the UNIX Directory setting.

By default, UNIX Directory is set to User's Home Directory. Next to Custom you'll see this setting shown another way, as ~(user-name). This uses the UNIX "~" notation for indicating home directories, together with the Profile Editor's placeholder (user-name) meaning the user the share's for.

To change the directory, click Custom, then type the directory. You can keep or remove the placeholder if you like.

See Also – "Placeholders" in the Help index.

Customize access to home directories

The most common ways to customize access involve granting read-only access, or denying access completely. You can use access rights to do these, but quicker—and easily reversible—ways exist:

- Deny access by clearing the Active option on the General tab.
- Give read-only access by checking the Read-only option on the Access tab.

Using placeholders

Placeholders give an extra level of control over some settings. For example, you can include the name of the TEDFS Administrator in access rights using (*admin-user*), so that if the Administrator changes, you don't need to change any other settings.

Also, using placeholders in master shares means that some settings in automatically generated shares can include share-dependent information, like the name of the share. In this way automatic user shares give access to the user's home directory, using the placeholder (user-name) in the master user share's UNIX Directory setting.

See Also – "Placeholders" in the Help index.

Give all users a special Windows home directory

Normally, user shares give access to UNIX home directories. You might want to keep UNIX home directories safe, and set up a special "Windows home directory" for each user to keep their Windows files in. Placeholders let you do this easily.

In the master user share's properties, click Custom for the UNIX Directory. Then change the directory shown, preserving the placeholder (user-name) in some way.

For example, to keep UNIX home directories and "Windows home directories" independent, change the setting to something like `/winhome/(user-name)`. Then create directories under `/winhome` on your UNIX server for each user, and give them appropriate UNIX permissions and ownership—usually, exactly the same as the original UNIX home directories.

Alternatively, you could change the setting to `~(user-name)/windows` and create a directory called `windows` (with the right permissions) in every user's UNIX home directory.

Using shared printers for custom output

By default, TEDFS creates shared printers automatically for the printers on your server. When users list the shares on the TEDFS server, they'll see a shared printer for every system printer, with the same name as the system printer in each case.

If you want, you can add shared printers that use the system printers in different ways. The details of how you do this will vary depending on your particular circumstances—for example, your flavor of UNIX, and whether the system printer is directly or indirectly connected to the UNIX server.

In general, you just need to change the command used to print the job. This might be by adding a flag to the existing command, or by using a different command entirely—preprocessing the print job using a separate program. Check your UNIX documentation for the different types of output you can generate.

Some special printing placeholders are defined so you can include useful information in the commands, for example the name of the user who submitted the job.

Using this technique, you can set up shared printers that:

- Print a banner page before every job.
- Print a header on each page.
- Print multiple pages on a single sheet of paper.
- Print landscape instead of portrait orientation.

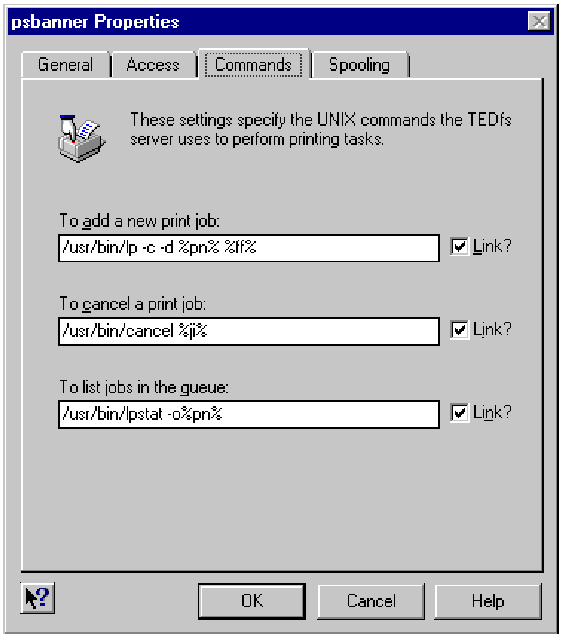
Remember also that you can set up different shared printers with different access rights. For example, you could reserve some special output only for particular users, or restrict access to the color printer.

See Also – "Placeholders, printers" in the Help index.

▼ *To customize printing commands*

1. In the Profile Editor, double-click the shared printer you want to customize printing commands for. To customize automatic shared printers, double-click (*Printers*).

2. Click the Commands tab.



3. Make any changes you want. Use Help to find out about the special printing placeholders you can use.
4. Click OK, then click Update Server on the Profile menu.

Allowing multiple NetBIOS applications

Technically, TEDFS is a NetBIOS application that runs over TCP/IP—sometimes called an *NBT application*. When you look in your Network Neighborhood or use the Connect Network Drive dialog, the names you see and use are NetBIOS names.

If you want to use only a single NBT application on the server—TEDFS—then it works, straight out of the box. You don't need to make any changes at all.

However, you might want to run more than one NBT application on the server. You can set this up using the Profile Editor.

About NBT applications

Each NBT application:

- Has at least one name, to identify the application.
- Listens for connections on a UNIX port.

An NBT application uses network broadcasts to announce, or *advertise*, its names. The naming doesn't need to be handled by the same program that listens for connections; they're independent. This means you could have an entirely separate program to advertise the names used by all NBT applications on the server. In fact, if you want to use more than one NBT application on the server, only one of the applications can advertise the names, as explained below.

With TEDFS, the naming process is part of, but distinct from, the rest of the server. The Profile Editor lets you name the server, and turn off naming altogether. It also lets you set up which port the server listens to.

Primary and secondary NBT applications

If you have more than one NBT application on the UNIX server, then one of them must advertise all the names used for all the NBT applications on the server. This is because only one application is allowed to use the appropriate naming ports, UDP ports 137 and 138 (different to the TCP port used to listen for connections).

Similarly, there's a standard port used for NBT connections, TCP port 139. Only one application can listen on this port. This application must make sure that connections intended for the other applications are rerouted, or *redirected* to the port each is listening on.

The NBT application that advertises the names and handles redirections is called the *primary* application. All others are called *secondary* applications.

In effect, the primary NBT application acts as a *proxy* for the other applications, while remaining an NBT application in its own right. All connections go through the primary, using the default port, 139. Some (or all) may be redirected to other ports, used by the other, secondary NBT applications.

In summary:

- The primary NBT application advertises all names, and handles all redirections to secondary NBT applications, by listening on the appropriate ports.
- The secondary NBT applications don't advertise any names, and listen for connections on custom ports, relying on the primary NBT application to redirect connections intended for them.

By default, TEDFS tries to run as the primary NBT application, as it's highly likely there'll only be a single NBT application on the server. If another NBT application is already running as the primary, the TEDFS server won't start and will generate an error message explaining the problem.

Working with multiple NBT applications

The first task is to decide which of the NBT applications is to be the primary, and which will be secondaries. You can configure TEDFS to be either; check the documentation for your other NBT applications to see if they prefer to be the primary or a secondary.

Once you've decided, make sure your users know the NBT applications will be out of action for a time—you'll need to stop TEDFS and the other NBT applications temporarily.

To find out whether a TEDFS server is running as the primary or a secondary application, start its Profile Editor and check the Port number on the Advanced tab of Server properties: if it's 139, the server's the primary application; otherwise, it's a secondary.

▼ *To set up TEDFS as the primary NBT application*

1. If TEDFS is already running as a secondary application, use the Profile Editor to change the port it listens on to the default port, 139. You change the port in Server properties, on the Advanced tab. Update the server, but *don't* restart it.

2. Find out the port numbers used by all secondary applications, and make sure any existing primary application is set up to run as a secondary application.

   Each secondary application must use a unique port number. Check the documentation for an application to find out how to configure the port number. Remember that port numbers less than 1024 are reserved for applications started as root.

3. Stop the TEDFS server if it is already running, then start it again. You can do this from the UNIX command line (remember to close the Profile Editor if it's running), or in the Profile Editor by clicking Restart Server on the Profile menu.

   The server will now run as the primary application.

4. Use the Profile Editor to advertise the names used by each secondary application. You list all the names next to Custom on the Identification tab of Server properties. Separate each name with a comma, ",".

5. On the Advanced tab of Server properties, add redirections for connections intended for the secondary applications.

6. Update the TEDFS server, and click Yes when the Profile Editor offers to restart it.

See Also – "Identifying the TEDFS server on the network," in Chapter 2, "Beyond the Basics" and "Adding redirections," later in this chapter.

▼ *To set up TEDFS as a secondary NBT application*

1. Make sure you stop any existing primary application before starting the TEDFS server.

   The default profile tells the server to start as a primary application. If there's another primary application, the server cannot start. You need to be able to run the server to change some settings using the Profile Editor.

2. Use the Profile Editor to stop the TEDFS server from advertising any names. To do this, open Server properties, click the Identification tab, and make sure both UNIX Hostname and Custom are not checked.

3. On the Advanced tab of Server properties, set up the server to listen for connections on a different port from the default, 139. Remember that TEDFS must be started as root, so choose a port number less than 1024.

4. Update the server, and click Yes when the Profile Editor offers to restart it.

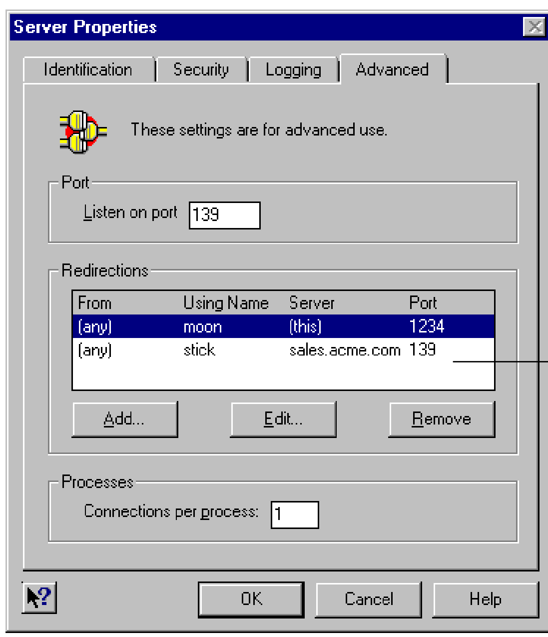
5. Now start your primary NBT application. Remember to set up the primary to redirect connections intended for the secondary applications to the ports they listen on, and to advertise all the names used for the secondary applications.

Adding redirections

Redirections allow a TEDFS server to act as a proxy for other NBT applications. You can set up TEDFS to redirect connections based on the name of the computer connecting to the TEDFS server, or the name it's trying to connect using. You can redirect connections to particular ports on the same UNIX server, or to other servers anywhere on the network.

▼ *To redirect a connection to another NBT application*

1. In the Profile Editor, open Server properties and click the Advanced tab.

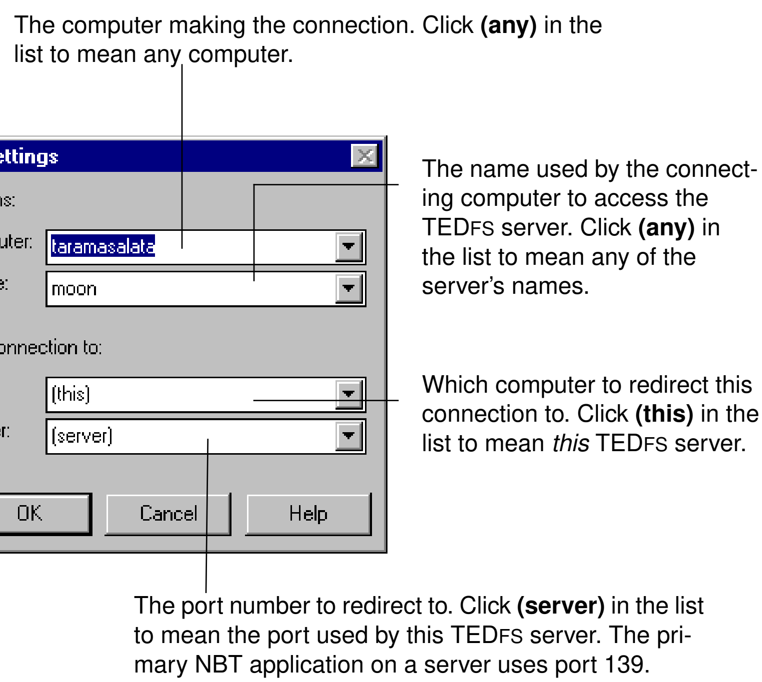


The list of current redirections. Redirections earlier in the list take precedence.

2. To add a new redirection, click Add.

   To edit an existing redirection, click it in the list, then click Edit.

3. In the Redirection Settings dialog, first specify which connections are affected by this redirection. Then, specify where those connections are redirected to.

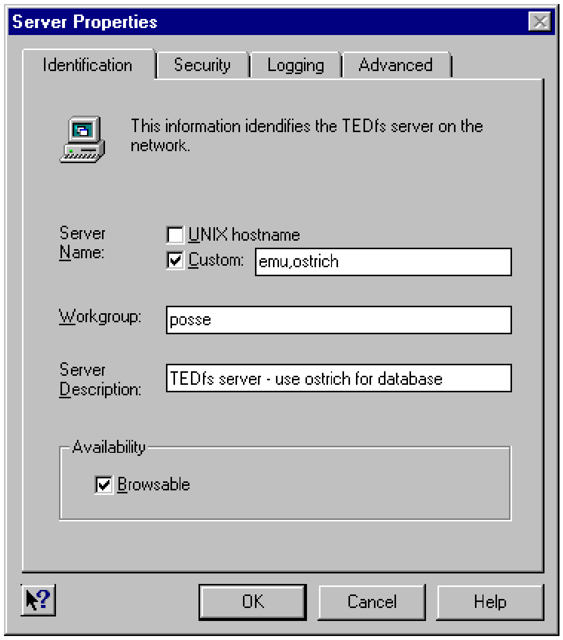


On the Advanced tab, redirections that appear earlier in the list take precedence over redirections that appear later in the list. To move a redirection, drag the list entry up or down.

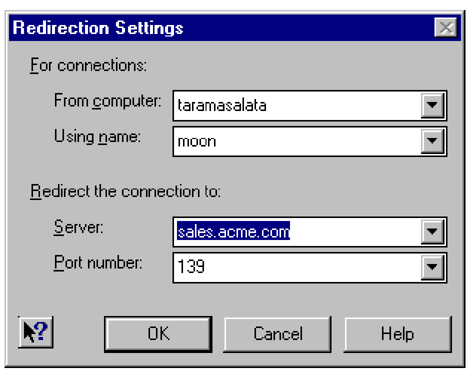
Example

For example, consider a UNIX server running two NBT applications: TEDFS (the primary), and a database server (a secondary, running on port 2345). You'd like the TEDFS server to use the NetBIOS name "emu", and the database server to use "ostrich".

In this case, you'd first set up the TEDFS server to advertise the names **emu** and **ostrich**, using the Identification tab of Server properties as shown below:



Then, add a redirection, so that people can use the database server. On the Advanced tab of Server properties, click Add. In the Redirection Settings dialog, specify that connections from any computer, that use the name **ostrich** to make the connection, are redirected to this server, but on port 2345. Your Redirection Settings dialog should look like this:



Finally, you need to update and restart the server for the new names and redirections to take effect.

Limitations

You can't "chain" redirections, for example so that a particular TEDFS server redirects to another TEDFS server, which redirects to somewhere else. This is a limitation of Windows, which will only redirect once.

If Windows is asked to redirect more than once, it will display an error message indicating that the computer did not receive the request.

Issues for Administrators 4

Look in this chapter to learn how to control the TEDFS server from UNIX, and find information about things that matter to you, such as security, licensing and troubleshooting.

Controlling TEDFS on UNIX

TEDFS Setup installs some files in a shared directory, by default `/usr/dt/TEDfs`. You use one of these files, a program called `bin/tedfs`, to control TEDFS.

Note – Don't try to control the server by running binaries directly, or by using kill. Using the TEDFS command is the only supported way of controlling the server.

The TEDFS command

The `TEDfs` command has the following syntax:

```
tedfs option [ option-specific-arguments ]
```

The various options let you control the server in different ways, or produce information about the server. The table briefly describes each option.

start	Starts the TEDFS server
stop	Stops the TEDFS server
status	Reports whether the TEDFS server is running
checkpoint	Stops the TEDFS server, archives the log files, then starts the server
query	Examines the TEDFS server's log files
archive	Archives the TEDFS server's log files
netinfo	Gives information about UNIX network interfaces
lockinfo	Reports which files are locked, and in what way
setup	Modifies or fixes the TEDFS server configuration
uninstall	Uninstalls the TEDFS server

See Also – "Logging" and "Licensing," later in this chapter.

Security and Authentication

In this section, we'll give information about how TEDFS authenticates users, and how you can be sure your server is as secure as possible while making access by your users as transparent as possible.

How users are authenticated

Authentication effectively starts when the user logs into the PC with a particular username. Not only is this username the name by which the PC knows the user, it's also the name by which the *network* knows the user, and plays an important part in authenticating the user to the TEDFS server.

This is because TEDFS uses *user level* security: the user must be authenticated by the server (logged in) before access is granted, but once authenticated can connect to any shares on the server, assuming the access rights for each share allow. The PC username is sent to the server during authentication, as described below.

User level security contrasts with *share level* security, which allows for different passwords for each share, and doesn't involve usernames. This means that actions aren't associated with a particular user, making it impossible to distinguish between users. For example, Windows for Workgroups operates in share level security.

In general, authentication involves these steps:

- The user tries to connect to the TEDFS server in some way, for example displaying the list of shares or trying to access a share.
- Windows and the server negotiate, and attempt to authenticate the user using the PC username. The user may be prompted for a password (or denied access).
- The server tries to match the user's username and password against entries in the UNIX user database, and grants or denies access according to the results.

Negotiation

The negotiation phase of authentication and its outcome has many variables, for example the action the user's trying to perform, the version of Windows, and the user's passwords on Windows and UNIX.

In some cases, Windows doesn't give users the option of entering a password if the passwords it tries aren't accepted. For example, File Manager on Windows for Workgroups will display an Access Denied dialog if you try to list the shares on a server before you've been authenticated.

In general, connecting to a share by name—using *server**share*—will work: either the user will be authenticated, or prompted for a password.

Usernames

The usernames sent by Windows are case-insensitive. However, the usernames stored in the UNIX user database are case-sensitive.

The TEDFS server will convert all usernames to lower-case before checking against the UNIX user database. This means that only users whose usernames on the TEDFS server are entirely composed of lower-case characters will be allowed to access the server with their own Windows usernames.

Similarly, Windows and UNIX usernames should be identical, otherwise it won't be possible to identify whether a particular Windows user has an account on the UNIX server.

If a username doesn't exist in the UNIX user database, the user is granted guest access to the server (in this case, the password doesn't matter). You can set up separate access rights for guest users, on a share-by-share basis, using the Profile Editor.

If a username does exist in the UNIX user database, TEDFS checks the password.

See Also – “Controlling access,” in Chapter 2, “Beyond the Basics”.

Passwords

The passwords sent by Windows are case-insensitive. However, UNIX passwords are case-sensitive.

TEDFS solves this problem by trying all possible combinations of upper- and lower-case characters in the password. For example, if Windows sends the password `FOO`, TEDFS would try the passwords `foo`, `foO`, `fOo`, `Foo`, `fOO`, `FoO`, `FOo`, and `FOO`.

You can use the Profile Editor to specify how many characters the TEDFS server will change the case of to try to match the password. By default this is 8, as most UNIX systems only use the first eight characters of passwords for authentication. Check your UNIX documentation to see if you should change it.

See Also – “UNIX and Windows passwords,” later in this chapter and “Passwords, case sensitivity” in the Help index.

If one of the password combinations matches, the user is authenticated. If no match is found, the user is denied access.

The UNIX user database

The TEDFS server uses the standard UNIX mechanisms for accessing the UNIX user database. For example, it doesn’t matter if your server uses /etc/passwd or NIS—TEDFS will check whatever you’re using.

Summary

- You should use identical lower-case usernames in UNIX and Windows.
- A user without an account on the UNIX server can be logged in with guest permissions (that you define), and the supplied password is ignored.
- A user with an account on the UNIX server is authenticated if the supplied password matches the UNIX password, allowing for different capitalization.
- A user with an account on the UNIX server is denied access if the supplied password doesn’t match the UNIX password, even after allowing for different capitalization.

UNIX and Windows passwords

You should be aware of some general UNIX and Windows password issues that affect the security of your UNIX server.

Case changing

During the authentication process, Windows sends a case-insensitive password to the TEDFS server, which must try to match against a case-sensitive UNIX password. Consequently, TEDFS tries different capitalizations of the password.

This decreases security, by effectively making UNIX passwords case-insensitive, and increases the time taken to authenticate users. There is no real alternative solution other than enforcing lower-case UNIX passwords, which decreases security still further.

However, you can use the Profile Editor to reduce the number of characters in the password the TEDFS server will change the case of. This increases security (as the server will try fewer passwords) and reduces the time taken to authenticate users, but restricts the acceptable range of passwords.

For example, if you change the setting to 2, then TEDFS will match a password with at most two upper-case characters in an otherwise lower-case password (or two lower-case characters in an upper-case password).

See Also – "Passwords, case sensitivity" in the Help index.

Using identical UNIX and Windows passwords

Often, Windows sends the TEDFS server a user's Windows password to try to authenticate the user. This means that if a user's Windows password is the same as their UNIX password, the user might not be prompted for a password.

However, using identical passwords decreases security:

- If the user leaves their PC unattended, other users can access the UNIX server.
- In some cases, the Windows password list can be decrypted easily.

The Windows password list

A security flaw present in Windows for Workgroups, and the first release of Windows 95, means it is computationally trivial to decrypt the Windows password list file. This file contains all the passwords that Windows caches for the user, including the TEDFS server password. Password caching is enabled by default.

This flaw is fixed by the Windows 95 Service Pack 1, which you can obtain from the Microsoft World-Wide Web site www.microsoft.com.

Windows NT does not use password lists, and so does not have this flaw.

If you use versions of Windows that suffer from this security flaw, you can disable password caching so that the TEDFS server password is not stored on the PC, and can't be decrypted in this way.

To disable password caching in Windows for Workgroups

1. Delete all `username.pwl` files in the Windows directory.
2. Add two lines to the `system.ini` file, in the `[network]` section:

```
CacheThisPassword=No
PasswordCaching=No
```

To disable password caching in Windows 95

1. Delete all `username.pwl` files in the Windows directory.
2. Create a file called `nocache.reg`, containing the following:

```
REGEDIT4


[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows
\CurrentVersion\Policies\Network]
"DisablePwdCaching"=dword:00000001
```

Important – The text between `[` and `]` must be on a *single line* in the file.

3. Double-click the `nocache.reg` file.

How passwords are sent

Windows and UNIX both provide facilities for the encryption of passwords. Unfortunately, the encryption mechanisms used are incompatible. This means passwords are sent over the network in plain text. Although this may present a security problem in environments where very high security is required, in most environments it does not affect the security of your system. Using TEDFS is no less secure than using the UNIX telnet program, for instance.

Filtering unwanted connections

You may only want to allow computers in your company to access your TEDFS server, and automatically reject all other computers. Similarly, you may want to allow access through a particular network interface on the server, and reject accesses using other interfaces.

The Profile Editor lets you do both of these easily, by setting up a list of *filters*.

When a computer tries to access the server, TEDFS checks the list of filters. The first filter that matches determines whether the connection is allowed or denied.

If none of the filters match, the connection is denied. Be careful not to delete all the filters, as this would deny all connections and mean the TEDFS Administrator couldn't run the Profile Editor.

See Also – "Troubleshooting," later in this chapter.

▼ *To filter unwanted connections*

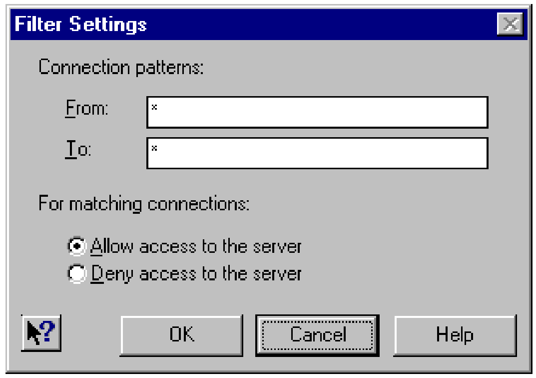
1. In the Profile Editor, open Server properties and click the Security tab.

The list of current filters. Filters earlier in the list take precedence.

2. To add a new filter, click Add.

   To edit an existing filter, click it in the list, then click Edit.

3. You'll see the Filter Settings dialog.



- Type the IP address or DNS name of the computer trying to connect to the TEDFS server.
- Type the IP address or DNS name of the network interface on the UNIX server that the computer's connecting through.
- Specify whether connections that match are allowed or denied.
- You can use the wildcard "*" to match more than one IP address or DNS name.

On the Security tab, filters that appear earlier in the list take precedence over filters that appear later in the list. To move a filter, drag the list entry up or down.

How to tell if an action will succeed

When a user tries to perform an action in a share on the TEDFS server, whether that action succeeds depends on a number of things:

- Whether connections are automatically rejected.
- How the user logged into the PC.
- How the user was (or wasn't) authenticated by the TEDFS server.
- What that user's access rights are in the share.
- What the UNIX permissions are.

Whether connections are automatically rejected

You can configure whether connections from particular IP addresses are automatically filtered out, and never allowed access to the server. Also, you can disallow access based on the IP address used to access the server.

How the user logged into the PC

Users must use the same usernames for Windows and UNIX, otherwise they'll only be allowed guest access.

How the user was (or wasn't) authenticated by the TEDFS server

If the user has an account on the UNIX server, they will be authenticated as that user if they supply the correct password, or rejected otherwise.

If the user doesn't have an account, they're given guest access.

What that user's access rights are in the share

You can set up different access rights for users with and without accounts.

You can also restrict the actions that the TEDFS server will allow them to perform, for example by allowing read-only access.

The access rights also determine which UNIX user the actions are performed as.

TEDFS uses the access rights to decide whether to *attempt* an action.

What the UNIX permissions are

Finally, the UNIX permissions determine whether the action can be performed by the UNIX user defined in the appropriate access right.

The result of the action will be the same as if the user had logged into the UNIX server and tried the action from the command line.

PC and UNIX file differences

In this section, we'll cover the differences between how PCs view files, and how UNIX views them. We'll explain how TEDFS helps smooth the way, and where circumstances conspire to make it impossible.

Filenames

With TEDFS, each file or directory effectively has three names:

- The UNIX filename.
- A long name, reported to Windows 95 and Windows NT.
- A short name, reported to all versions of Windows.

When a long or short name is used in Windows, TEDFS uses the correct UNIX filename.

Potentially, there are fewer long names than UNIX filenames, and fewer short names than long names. This means that the process of converting UNIX filenames to long names, then to short names, may result in two unique UNIX filenames losing their uniqueness. TEDFS ensures that this can't happen by adding unique suffixes to the basename where a clash would occur.

The next sections describe the two conversion processes, and how unique suffixes are generated. The conversion processes are identical to those used by Windows.

Converting UNIX filenames to long names

This involves the following steps:

- Removing trailing dots.
- Removing the characters not allowed in long names: `? " / \ < > * | :`
- Prefixing the DOS reserved basenames with "_". These basenames are: `aux com1 com2 com3 com4 con lpt1 lpt2 lpt3 nul prn clock$`
- Adding unique suffixes to the basename to avoid name clashes, as described below. A name would clash if it's the same as another name, ignoring case.

Converting long names to short names

This involves the following steps:

- Removing all dots but the last.
- Removing the additional characters not allowed in short names: `[ ] ; = + ,`
- Converting to upper case.
- Truncating the basename to 8 characters, and the extension to 3 characters.
- Adding unique suffixes to the basename to avoid name clashes, as described below. A unique suffix is also added if there's no basename—for example, one would be added for the UNIX filename `.cshrc`.

Making filenames unique

To make a set of clashing long or short names unique, TEDFS first generates a 32-bit checksum for each, based on the original UNIX filename. The checksum is converted to a printable form.

Then, a suffix is formed for each clashing name, using the special character for the share (by default, ~) and the last character of the printable checksum (or however many characters are needed to form unique suffixes for all clashing names).

Finally, the suffixes are added to the basenames of the clashing names, replacing the last two (or more) characters to ensure the basenames don't exceed 8 characters in length.

Permissions

UNIX permissions are richer than Windows file attributes, but Windows file attributes aren't a subset of UNIX permissions.

Of the four standard Windows file attributes—Archive, Hidden, System and Read-only—TEDFS supports the Read-only attribute. The other attributes don't map onto UNIX permissions.

The Read-only attribute of a Windows file is set if the user accessing a share isn't allowed to write to the file, according to the UNIX permissions.

Similarly, changing the Read-only attribute in Windows sets or clears the three UNIX write permissions for owner, group and others.

Remember that even if the access rights for a user allow a particular action, the UNIX permissions determine whether that action succeeds or fails.

Semantics

Some actions on files will have different effects in Windows than UNIX users might expect.

Deleting symbolic links

On UNIX, deleting a symbolic link deletes the directory entry, and doesn't affect the file or directory referenced by the symbolic link. However, Windows doesn't have the concept of a symbolic link.

For files shown in Windows that, in reality, are symbolic links on the TEDFS server:

- Deleting a symbolic link to a file will just delete that link. The referenced file is unaffected.
- Deleting a symbolic link to a directory will delete all files and subdirectories in the directory pointed to by the link, then delete the symbolic link. This is because Windows tells TEDFS to delete the contents of the directory, then delete the directory.

Important – For an experienced UNIX user this is not the expected behavior, but is beyond the control of TEDFS. Be careful to ensure that directories are not deleted accidentally. If necessary, hide symbolic links in shares.

See Also – "Showing and hiding symbolic links," in Chapter 2, "Beyond the Basics".

Renaming and deleting files

On UNIX, the permissions of a directory determine whether you can rename or delete files in that directory.

In Windows, the Read-only attribute on a file determines whether you can rename or delete it.

Deleting files in use

On UNIX, you can delete or rename a file that's in use. In Windows, you can't.

This is because UNIX keeps file information separately from directory information, whereas Windows keeps them together.

TEDFS allows files to be deleted or renamed while in use.

Free disk space

Windows will report the free disk space based on the root directory of each share on the TEDFS server. However, there may be symbolic links in the share pointing to other filesystems, which means the effective free disk space is larger.

Also, as UNIX allows multiple directory entries per file, deleting a file may not necessarily increase the amount of available disk space.

File locking

In this section, we'll describe TEDFS file locking, which lets a program be sure it can read and write to a file (or part of a file) without another program doing so at the same time.

Overview

When you edit a file, you perform at least three actions on that file: you read the file into your editing program, you make changes, and finally you save the file.

Imagine that two people want to edit the same file. The first reads the file into an editor and starts making changes. Then the second user reads the file into an editor, makes changes and saves the file before the first person has finished editing. Some time later, the first user finishes editing the first copy of the file and saves it.

The second person's changes have been lost, as they weren't in the file when the first user started editing. Ideally, the second user should have been prevented from editing the file until the first user had finished.

The process of controlling which actions users are allowed to perform on a file while another user is performing an action on a file is called `file locking`.

For example, if you're reading a file but have no intention of making changes, it may be acceptable for others to read the same file. However, it may be unacceptable for others to modify the file while you're reading it.

File locking lets users be sure that files they work with are up-to-date, and that all their changes will be preserved.

UNIX and Windows file locking

UNIX and Windows offer different file locking facilities. UNIX offers only rudimentary file locking, which isn't used by many applications. In contrast, Windows provides rich and flexible file locking.

Comparing Windows and UNIX file locking in detail reveals two important differences:

- UNIX doesn't allow open locks or opportunistic locks. Windows does.
- Windows uses 32-bit range locks, whereas UNIX is restricted to 31-bit range locks. Also, the NFS lock daemon becomes unreliable with range locks above 29 bits long.

The different types of lock are explained in more detail below.

TEDFS file locking

TEDFS has an independent component, called the *lock daemon*, which provides full Windows locking semantics. The lock daemon manages lock requests from PCs and maintains a lock database which contains information about all file locks which are currently in place. When a PC requests access to a file in a share, TEDFS consults the lock daemon for the availability of that file, and access is granted or denied as appropriate.

TEDFS provides three types of locks for PCs to use with files.

Lock type	Description
Open	Used when a file is first opened. These let you specify exactly which actions other users are allowed to perform on a file while you are using it.
Record	Used to prevent other users from accessing a particular portion of a file while you are using it.
Opportunistic	Gives the user complete control over a file while they are using it. If another user needs to edit the file, the client with the opportunistic lock is asked to finish editing for the moment and relinquiah lock.

Should I use opportunistic locks?

Opportunistic locks give great performance benefits, as one user has complete control over the entire file at one time and can edit a local copy, only updating the file itself immediately before the lock is removed.

However, opportunistic locks don't provide protection from simultaneous editing by a PC user and a UNIX user.

You should use opportunistic locks in the following cases:

- If the files in a share will only be modified by PC clients.
- If the files in a share will be read by PC and UNIX clients, but never modified by either, for example with a CD-ROM drive.

You shouldn't use opportunistic locks if the files in a share might be modified by both PC and UNIX clients.

Why UNIX locking is inadequate

In detail, these are the technical reasons why UNIX locking is inadequate for use with Windows programs:

- A read-write lock can't be established on a file only opened for reading.

  TEDFS can't open all files for both reading and writing, as the user may not have permission to do so. As a result, all locks must be treated as exclusive locks.

- With range locks, the 32-bit offsets from Windows must be mapped onto 31-bit offsets for UNIX (UNIX uses the most significant bit as the sign). Also, the NFS lock daemon becomes unreliable with range locks above 29 bits long.

  The problem can be overcome by removing two bits (29 and 30) and putting the two bits from 31 and 32 in their place. This reduces the largest possible file size to 256Mbytes before the areas of the file overlap. OLE2 makes extensive use of full 32-bit values.

- UNIX doesn't perform reference counting of locks (a side-effect of UNIX coalescing locks).

  If UNIX locks the same area twice (or has two locks with overlapping areas), only one unlock is needed to remove both the locks.

- UNIX does not provide whole-file locks.

  It's not possible to implement this by placing a range lock from the beginning of the file up to the maximum file size, as this lock will then conflict with the real offset/range locks made by the client.

- Two common UNIX bugs make life difficult:
  - If a file is opened twice (so you have two file handles) and either file handle is closed, then all locks issued on both file handles are released.
  - UNIX will often report that unlocks have succeeded on areas that weren't even locked.
- UNIX doesn't provide opportunistic locks, nor does it provide any means to implement them.

Logging

TEDFS log files contain useful information about who uses (and who tries and fails to use) the server. In this section, you'll learn how to examine the TEDFS log files, how to customize what's logged, and how to archive log files.

Checking the logs

You use the `TEDfs` command's `query` option to examine the TEDFS log files, and find out about server usage.

Remember that you can customize what's logged: if the information doesn't seem to be correct, make sure the server was logging the information in the first place.

Arguments to the query option let you view the information in different ways, as described in the table.

See Also – "The TEDFS command," earlier in this chapter and "Customizing what's logged," later in this chapter.

Note – Each argument is prefixed by two dashes, "--", not one.

Argument	Description
--conns	Displays information about successful and failed connections to the TEDFS server, and shows the maximum number of simultaneous connections.
--now	Displays which computers are currently connected to which shares. Updated every five minutes.
--err	Displays the error log.
--names	Displays information about the success or failure of the server's attempts to acquire its NetBIOS names.
--shares	Displays information about usage at the share level, showing counts and percentages. You must supply one of the arguments --byuser, --byshare or --bymachine, to show the information by user, share or machine.

--ops Displays information about usage at the operation level, with success/failure ratios shown for each operation. You must supply one of the arguments --byuser, --byshare or --bymachine, to show the information by user, share or machine.

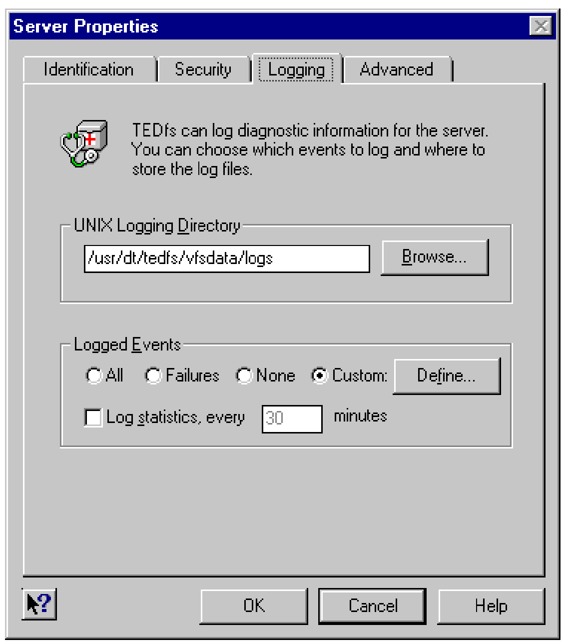
--all Displays all information (the default). You can omit a section of information from the output by using the appropriate argument with the --all flag. For example, TEDFS query --all --err displays all information except the error log.

Customizing what's logged

You use the Profile Editor to customize the information the TEDFS server writes to log files. If you want, you can choose to log just failed operations, or just particular operations you're interested in.

▼ *To customize what's logged*

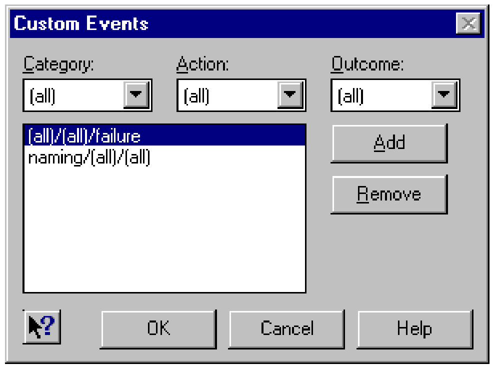
1. In Server properties, click the Logging tab.



2. In Logged Events, click Custom, then click Define.

Tip – Click All, Failures or None in the Logged Events box to use the most common logging settings.

3. In the Custom Events dialog, do the following:
   - To log something new, click the Category, Action and Outcome you want to log, then click Add. Click **(any)** in the lists to mean any Category, Action or Outcome as appropriate.
   - To remove something you don't want to log, click an entry in the list, then click Remove.



4. When you're done, click Update Server on the Profile menu, and restart the server.

Archiving and checkpoints

Log files can use a significant amount of disk space on the server. Archiving the logs compresses the files, freeing disk space. TEDFS Setup lets you configure your system to perform a "checkpoint" every week at the same time, which stops the server, archives the log files, then restarts the server.

Log files are stored in the TEDFS data directory you specified during TEDFS Setup, by default in a subdirectory logs. Each server uses multiple log files, which it may write to at any time. If you want to examine the logs, use the TEDFS query command.

Note – Do not delete any files in the logs directory. If you delete a file the server still has open, the directory entry will disappear but the disk space won't be freed until the server exits. To reclaim some disk space, first archive the logs, then delete the archive.

▼ *To archive the logs if the server's running*

1. Log in as root on the UNIX server.

2. Type the following, replacing `tedfs_dir` with the name of the Vision family shared directory, by default `/usr/dt/TEDfs`:

   ```
   tedfs_dir/bin/TEDfs checkpoint
   ```

   This stops the server, archives the logs, then restarts the server.

To archive the logs if the server's not running, just type TEDFS archive.

Archiving the logs compresses the files and moves them to a numbered subdirectory of the `logs` directory. A file `summary.txt` alongside the compressed log files contains the results of performing `TEDFS query` at the time of the archive.

You can delete archived log files without affecting the server.

Naming across an intranet

In this section, we'll describe the problems you may encounter trying to enable access to TEDFS servers (and other NetBIOS applications) across an *intranet*, or Wide Area Network (WAN), and present some possible solutions.

Overview

When a client wants to access a remote server or application, the client must have a mechanism for identifying and contacting the service in the first instance. Service identification is commonly referred to as *naming*.

Windows uses NetBIOS names to identify applications and servers on a network. NetBIOS names are purely tags for identifying a service; they are not hierarchical, and do not contain any further information about that service. In contrast, DNS names are hierarchical. For example, a DNS name might be jelly.sales.acme.com, whereas a NetBIOS name would just be jelly.

NetBIOS names are well-suited to Local Area Networks. Usually, all the computers on a LAN have different names, so there's no problem referring to each computer by its name, such as jelly. On an intranet, however, there may be many computers called jelly. How do we contact the specific one we want? DNS names solve the problem: jelly.sales.acme.com refers to a specific computer called jelly in a specific part of the world.

Also, Windows uses broadcasts to locate services. Broadcasts are restricted to LANs (it wouldn't be very practical if every time anyone started a network application a broadcast was sent to every computer in the world). So Windows is unable to locate services across intranets.

In summary, naming across an intranet presents the following problems:

- NetBIOS names must be unique, which is impractical on an intranet.
- NetBIOS naming uses broadcasts, which is impractical on an intranet.

Solutions

There are a number of solutions to this problem, which are appropriate in different circumstances.

Use DNS for NetBIOS name resolution

The first solution is to tell Windows to use DNS for NetBIOS name resolution. This option lets you supply a list of suffixes which Windows appends to a NetBIOS name to try to form a DNS name.

For instance, if you're looking for a computer called jelly, and you've supplied the suffixes sales.acme.com, research.acme.com and marketing.acme.com, Windows will try to locate computers with the DNS names jelly.sales.acme.com, jelly.research.acme.com, and jelly.marketing.acme.com.

To tell Windows to use DNS for NetBIOS name resolution, and supply a list of DNS name suffixes, follow the instructions for your version of Windows:

On…	Do this…
Windows for Workgroups	Display your TCP/IP protocol settings. Click Advanced and make sure Enable DNS for Windows Name Resolution is checked. Click DNS and add your DNS name suffixes.
Windows 95	Display your TCP/IP protocol settings. On the DNS tab, add your DNS name suffixes. You can check that NetBIOS name resolution uses DNS by running the winipcfg program.
Windows NT 3.51	Display your TCP/IP protocol settings. Click Advanced and make sure Enable DNS for Windows Name Resolution is checked. Click DNS and add your DNS name suffixes.
Windows NT 4	Display your TCP/IP protocol settings. On the WINS Address tab, make sure Enable DNS for Windows Resolution is checked. On the DNS tab, add your DNS name suffixes.

This solution is best if you're using DNS and have unique computer names in the domains in which you are searching.

Use multiple server names

The second solution is to use multiple names for each server, and set up TEDFS to redirect connections made using these names to the appropriate server.

For example, if you have three TEDFS servers on three separate LANs—in London, New York and Berlin, called london, newyork and berlin—each server would have two redirection patterns.

The server in London would need a local name for itself, and redirection patterns for the names newyork and berlin, redirecting to the servers in New York and Berlin respectively. Similarly, the servers in New York and Berlin would each have redirection patterns for the other two servers.

This solution is best for sites where there are few remote servers but many clients, since all administration is done on the servers.

See Also – “Adding redirections,” in Chapter 3, “The Possibilities”.

Use the LMHOSTS file

The third solution is to edit the `LMHOSTS` file, which contains instructions about how to map NetBIOS names to remote IP addresses. If the `LMHOSTS` file doesn’t exist, copy `LMHOSTS.SAM` to `LMHOSTS`.

In Windows for Workgroups and Windows 95, you’ll find both files in the Windows directory. In Windows NT, you’ll find them in the \system32\drivers\etc subdirectory of the Windows directory.

Once you’ve edited the LMHOSTS file, you need to instruct Windows to use the file. How you do this depends on your version of Windows:

On…	Do this…
Windows for Workgroups	Display your TCP/IP protocol settings and make sure Enable LMHOSTS Lookup is checked.
Windows 95	Nothing. LMHOSTS lookup happens automatically.
Windows NT	Display your TCP/IP protocol settings and make sure Enable LMHOSTS Lookup is checked. Click Import LMHOSTS to locate the LMHOSTS file.

This solution is best for sites with few PC clients needing access to few remote servers, because all clients need to be configured.

Use a WINS server

The final solution is to use a WINS server on each LAN. The WINS servers must be configured to communicate with each other, so they can manage the flat namespace of NetBIOS names to allow access to servers on remote networks. Each client needs to be configured with the address of its local WINS server.

Most machines can use both WINS and broadcasts to locate services. The WINS server is asked to locate the service first; broadcasts are used if the WINS server fails to locate the server. This means this solution is compatible with all of the above solutions.

WINS servers aren't supported by TEDFS. You should use this solution in conjunction with one of the other solutions: PCs can communicate with other PCs using WINS, while the other solution lets them communicate with the TEDFS server.

This solution is best if you have mobile users who travel between sites.

Note – WINS is likely to be replaced by Dynamic DNS in future versions of Windows.

Troubleshooting

As soon as you connect two computers together, you need to make careful decisions about how the computers interact. As networks grow, the potential for trouble increases dramatically.

TEDFS works seamlessly for most networking environments. But no two networks are identical. In this section, we'll point out some of the areas that could lead to problems.

Be sure to check the Help for other troubleshooting information about the TEDFS server and the Profile Editor.

See Also – "Troubleshooting" in the Help index.

General problems

If you can't start the TEDFS server or the Profile Editor, the profile may be corrupt, or you may have accidentally changed some settings that mean you are no longer allowed to access the Profile Editor.

You can fix a profile by running TEDFS Setup again.

▼ *To fix a corrupt or invalid profile*

1. On UNIX, type the following, replacing `tedfs_dir` with the name of the Vision family shared directory, by default `/usr/dt/TEDfs`:

   ```
   tedfs_dir/bin/TEDfs setup
   ```

2. Follow the instructions on your screen.

See Also – “The TEDFS command,” earlier in this chapter.

Remember to check the README file (installed in the TEDFS server’s main installation directory) for late-breaking information that couldn’t make this book or the online Help.

Finding and browsing for computers

In Windows 95 and NT 4, you can click Find Computer on the Start menu to locate computers on the network.

In Windows for Workgroups and NT 3.51, you can click Browse in File Manager’s Connect Network Drive dialog to browse workgroups for computers.

Both methods can locate TEDFS servers on the network. However, there are some Windows problems that in some cases can stop TEDFS from appearing, or listing shares.

Find Computer

A problem with Windows 95 means that a TEDFS server won’t be located unless at least one of the following is true:

- You’ve connected to the server in that Windows session.
- Windows has cached your UNIX password.
- Your Windows password is the same as your UNIX password.

Windows NT 4 successfully locates TEDFS servers.

Clicking Run on the Start menu and typing *server* works more reliably than searching for a computer named ***server*** in the Find Computer dialog.

Browsing in Connect Network Drive dialog

A problem with both Windows for Workgroups and NT 3.51 means that shares for a TEDFS server won't be listed unless at least one of the following is true:

- You've connected to the server in that Windows session.
- Windows has cached your UNIX password.
- Your Windows password is the same as your UNIX password.

Typing share names in the Path box works more reliably than browsing.

CD-ROM drives

You can set up a shared folder that accesses your UNIX CD-ROM drive, if you have one.

On some flavors of UNIX, files on the CD-ROM have version numbers, such as ";1", added to the end of the name. This can cause problems with Windows, which uses the file extension to determine the type of a file. For example, using Windows 95 a file called program.exe might appear as program.exe;1, and Windows would not recognize it as an executable program.

If possible, make sure the CD-ROM is mounted so that versions numbers aren't shown. Check your UNIX documentation for the correct mount command to use.

Printing

UNIX printer settings may cause problems when printing from Windows.

Printer driver

Make sure your users have installed the correct Windows printer driver for your UNIX printer. Be careful to give the correct make and model. In some cases, print jobs may fail to appear even with a similar make and model of printer.

Double-conversion

Windows printer drivers output data intended for the printer itself. Similarly, UNIX printers can use "filters" to convert print jobs to printable data, in much the same way. When printing using TEDFS, it's important that the UNIX system doesn't perform any filtering. Otherwise, the print output may not appear as expected.

If you see, for example, a PostScript file printed as text, then your UNIX printer is filtering raw printer data unnecessarily.

The default TEDFS print command for shared printers tries to use the "raw" mode, that bypasses any filters you might have for that printer. However, not all systems support filter bypassing.

Watch out for filters that identify a PostScript print job by looking for "%!" as the first two characters in the job: Windows often puts a CTRL+D character, or other data, before the PostScript.

Feedback from printers

Some printers return information about their current status, for example that they're out of paper. Unfortunately, this information is not returned to UNIX, and so doesn't appear in Windows when printing using TEDFS.

Other problems

If you're having problems printing using TEDFS, try printing to a file from Windows. Then you can copy the resulting file to UNIX, and try different variations of print command. When you find one that works, you can set up the shared printer to use that print command.

See Also – "Customizing printing commands," in Chapter 2, "Beyond the Basics".

Networking

Naming and broadcasts

TEDFS uses NetBIOS naming, which broadcasts names using the UNIX server's broadcast address. It's important that this broadcast address is correct, or the TEDFS server won't appear in workgroups.

You can use the TEDFS netinfo command to display information about the network interfaces on your system.

See Also – "The TEDFS command," earlier in this chapter.

The UNIX server and your PCs must be using the same subnet mask. Some flavors of UNIX use the wrong subnet mask by default. See your UNIX documentation on the ifconfig command to find out how to set broadcast addresses and subnet masks.

PC networking commands

Windows and DOS contain some useful tools for mapping drives, and changing and displaying a PC's network settings. You may find the following commands helpful:

- `net`
- `nbtstat`
- `winipcfg` on Windows 95
- `ipconfig/all` on other versions of Windows

In each case, you can use the command line option /? to get help on the command.

Names

In networking, a computer may have many different names, each used in particular circumstances. This section summarizes the types of name a computer may have, and where each type fits with the other types.

A computer may have:

- Exactly one hostname.
- Zero or more network interfaces (for example, ethernet adapters), each with a single hardware address. Typically, there'll be a loopback interface (a software interface local to the computer, used to do networking to itself), and a physical network card (for networking with other computers).
- Zero or more IP addresses, each mapping to a single hardware address. Typically, there'll only be a single IP address.
- Zero or more DNS names, each mapping to one or more IP address. Typically, there'll only be a single DNS name.

TEDFS will work as expected, no matter how many DNS names, IP addresses and network interfaces you have on your UNIX system.

Additionally, an application may have zero or more NetBIOS names. (NetBIOS names belong to applications, not computers, though typically there's only a single NetBIOS application per computer.)

TEDFS advertises all NetBIOS names over all network interfaces that support broadcasts. When other computers try to access TEDFS, the NetBIOS name will be resolved to the relevant IP address (as TEDFS uses NetBIOS over TCP/IP).

A typical UNIX system might use its different names in the following way:

- A hostname `jelly`.
- An IP address, 192.168.5.44, that maps to a single ethernet address corresponding to the UNIX system's only ethernet card.
- A DNS name, `jelly.sales.acme.com`, that maps to the single IP address.

The UNIX system may also be running a single NetBIOS application, with name `jelly`, that maps onto the single IP address.

Working with your existing software

In this section, we'll explain how TEDFS can work alongside your networking software.

PC TCP/IP stacks

PCs and the TEDFS server communicate using the SMB protocol, which runs over NetBIOS.

Technically, NetBIOS can run over multiple network transports, for example NetBEUI, IPX/SPX and TCP/IP. In fact, any number of transports could be used simultaneously.

To connect to a TEDFS server, a PC requires a TCP/IP stack that supports the NetBIOS protocol. Both Windows 95 and Windows NT are supplied with a compatible TCP/IP stack, and a version for Windows for Workgroups is distributed with TEDFS.

TCP/IP stacks from other vendors will work if they support the NetBIOS interface. You may find that although a particular PC can communicate with another PC, it can't connect to a TEDFS server. This may be because the PCs are using NetBIOS over NetBEUI or another network transport. In this case, the TCP/IP stack does not fully support the NetBIOS interface.

Other file and printer sharing programs

TEDFS will coexist with other file and printer sharing programs on UNIX without customization, as long as those programs don't conflict with TEDFS as the primary NetBIOS over TCP/IP (NBT) application.

See Also – "Allowing multiple NetBIOS applications," in Chapter 3, "The Possibilities".

You can set up TEDFS to work alongside other NBT applications, by designating one as the primary and the others as secondaries.

PCs can run more than one file and printer sharing client at the same time.

Performance

A number of factors may affect the apparent performance of the TEDFS server.

- The theoretical maximum bandwidth for standard ethernet traffic is 1Mbyte/second (10Mbps).
- In practice, the real speed is dependent on the speed of the client and the speed of the UNIX filesystem. If NFS is also involved (for example, a shared folder might access a directory mounted over NFS) speeds will reduce further.
- TEDFS does not buffer data. However, both UNIX and Windows do.

Connections per process

Each TEDFS process on UNIX can handle a number of connections from PCs, up to a maximum. Once the maximum is exceeded, a new process starts. Only one connection can be attended to in each process at a time.

By changing the maximum number of connections per process, you can trade off response time against resource usage. More connections per process makes more efficient use of server resources, but may result in decreased performance for users. Fewer connections per process increases performance for users, but uses server resources less efficiently.

▼ *To optimize process usage*

1. In the Profile Editor, open Server properties, then click the Advanced tab.
2. In Processes, type the maximum number of connections allowed per process in the space provided.
3. Click OK, then click Update Server on the Profile menu and restart the server.

Locking

Two changes can improve the performance of the TEDFS server:

- If a share gives access to a CD-ROM drive then no files will ever be modified, so you can turn off locking completely for that share.
- If you are sure that files in a share won't be edited by UNIX users as well as PC users, then turn on opportunistic locking in that share.

See Also – "TEDFS file locking," earlier in this chapter.

International characters

Windows is available localized into multiple languages. Your users may want to name files using characters not found in English, for example characters with accents. If not, you can skip this section.

TEDFS operates in code page 850. This is the normal character encoding for most DOS and Windows installations.

- If your UNIX system supports code page 850, then foreign characters will appear correctly in both UNIX and Windows directory listings.
- On systems that don’t support code page 850, UNIX directory listings may show filenames using different characters. However, the filenames reported to Windows will be correct.

We recommend that your users use code page 850 on Windows.

Index

R

S

W