










## Resource Kit Tools for Microsoft® Windows NT™



-  [Computer and Network Administrative Tools](#)
-  [Desktop Tools](#)
-  [File System Tools](#)
-  [Performance and System Monitoring Tools](#)
-  [Registry Tools](#)
-  [Setup and Troubleshooting Tools](#)
-  [Tools for Developers](#)

Many of these tools are designed to work only if you logged on as a member of an Administrators group. Also, some tools have been tested only for running under an administrator account.












These tools were developed internally at Microsoft. Any use of these tools is at your own risk. These tools are provided "AS IS" without warranty of any kind. Microsoft disclaims any implied warranty of merchantability and/or fitness for a particular purpose.

## Resource Kit Tools for Microsoft Windows NT














▼ Expand

- ☐ Computer and Network Administrative Tools
- ☐ CHOICE.EXE: Input from Batch Files
- ☐ DUMPEL.EXE: Dump Event Log
- ☐ FIFO.CMD: Enable FIFO Buffers for Modems
- ☐ FLOPLOCK.EXE: Lock Floppy Disk Drives
- ☐ MIBCC.EXE: SNMP mib compiler
- ☐ NET2COM.EXE: Modem Pooling
- ☐ NETSVC.EXE: Command-line Service Controller
- ☐ NETWATCH.EXE: Net Watcher
- ☐ PERMS.EXE: File Access Permissions per User
- ☐ RASUSERS.EXE: Enumerating Remote Access Users
- ☐ REMOTE.EXE: Remote Command Line
- ☐ SFMATALK.SYS: AppleTalk Protocol and SFM Administrator Tools
- ☐ SLEEP.EXE: Batch File Wait
- ☐ SNMPUTIL.EXE: SNMP Browser
- ☐ SRVMGR.EXE: Server Manager
- ☐ UPEDIT.EXE: User Profile Editor
- ☐ USRMGR.EXE: User Manager for Domains
- ☐ WINAT.EXE: Command Scheduler
- ☐ WINVTP.EXE: Windows NT Communications
- ☐ Desktop Tools
- ☐ File System Tools
- ☐ Performance and System Monitoring Tools
- ☐ Registry Tools
- ☐ Setup and Troubleshooting Tools
- ☐ Tools for Developers



















## Resource Kit Tools for Microsoft Windows NT

-  Computer and Network Administrative Tools
-  Desktop Tools
-  ANIEDIT.EXE: Animated Cursor Creator
-  IMAGEDIT.EXE: Image Editor
-  SCHEMES.CPL: Animated Cursors and Cursor Schemes
-  TOPDESK.EXE: Multiple Desktops
-  File System Tools
-  Performance and System Monitoring Tools
-  Registry Tools
-  Setup and Troubleshooting Tools
-  Tools for Developers











## Resource Kit Tools for Microsoft Windows NT

-  Computer and Network Administrative Tools
-  Desktop Tools
-  File System Tools
-  COMPRESS.EXE: File Compress
-  DELPART.EXE: Delete Partitions
-  DIRUSE.EXE: Directory Disk Usage
-  EXETYPE.EXE: Finding the Executable Type
-  SCOPY.EXE: File Copy with Security
-  WINDIFF.EXE: File and Directory Comparison
-  Performance and System Monitoring Tools
-  Registry Tools
-  Setup and Troubleshooting Tools
-  Tools for Developers














## Resource Kit Tools for Microsoft Windows NT

-  Computer and Network Administrative Tools
-  Desktop Tools
-  File System Tools
-  Performance and System Monitoring Tools
-  BROWMON.EXE: Browser Monitor
-  BROWSTAT.EXE: Browser Status
-  DOMMON.EXE: Domain Monitor
-  DRIVERS.EXE: Device Driver Information
-  FTPCTRS: FTP Server Counters for Performance Monitor
-  PERFMTR.EXE: Performance Meter
-  PMON.EXE: Process Resource Monitor
-  PSTAT.EXE: Process and Thread Status
-  PVIEWER.EXE: Process Viewer
-  QSLICE.EXE: CPU Usage by Processes
-  SMBTRACE.EXE: Server Message Block Tracer
-  Registry Tools
-  Setup and Troubleshooting Tools
-  Tools for Developers

## Resource Kit Tools for Microsoft Windows NT











-  Computer and Network Administrative Tools
-  Desktop Tools
-  File System Tools
-  Performance and System Monitoring Tools
-  Registry Tools
-  GRPTOREG.EXE and REGTOGRP.EXE: Program Manager Groups to Registry
-  REGBACK.EXE: Registry Backup
-  REGENCY.HLP: Windows NT Registry Entries
-  REGINI.EXE: Registry Change by Script
-  REGREST.EXE: Registry Restoration
-  Setup and Troubleshooting Tools
-  Tools for Developers

## Resource Kit Tools for Microsoft Windows NT

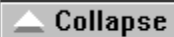
-  [Computer and Network Administrative Tools](#)
-  [Desktop Tools](#)
-  [File System Tools](#)
-  [Performance and System Monitoring Tools](#)
-  [Registry Tools](#)
-  [Setup and Troubleshooting Tools](#)
-  [JUL93HCL.HLP: Hardware Compatibility List](#)
-  [NTCARD.HLP: Adapter Help](#)
-  [NTDETECT.COM: Startup Hardware Detector](#)
-  [REPAIR.EXE: Update Emergency Repair Disk](#)
-  [TROUBLE.HLP: Troubleshooting Flowcharts](#)
-  [WINNTP.EXE: Computer Profile Setup](#)
-  [Tools for Developers](#)












## Resource Kit Tools for Microsoft Windows NT

-  Computer and Network Administrative Tools
-  Desktop Tools
-  File System Tools
-  Performance and System Monitoring Tools
-  Registry Tools
-  Setup and Troubleshooting Tools
-  Tools for Developers
-  OS2.API
-  POSIX Utilities
-  UCONVERT.EXE: Unicode Converter

## Resource Kit Tools for Microsoft Windows NT



- [!\[\]\(95b42f0077faf7439a26242a54e021ec\_img.jpg\) Computer and Network Administrative Tools](#)
- [!\[\]\(e097ab4c08b8186dd0908330bbc2dc28\_img.jpg\) CHOICE.EXE: Input from Batch Files](#)
- [!\[\]\(1e9d865c5de095f8e3304757c49e79d7\_img.jpg\) DUMPEL.EXE: Dump Event Log](#)
- [!\[\]\(735b10d724a5f0ec5005c4eb3eb9c9d1\_img.jpg\) FIFO.CMD: Enable FIFO Buffers for Modems](#)
- [!\[\]\(e6250f05bc27fa93236b816562b699f9\_img.jpg\) FLOPLOCK.EXE: Lock Floppy Disk Drives](#)
- [!\[\]\(d190cc638f389909d4b049d6c19e4cb2\_img.jpg\) MIBCC.EXE: SNMP mib compiler](#)
- [!\[\]\(4d34001966c7597a8c3e4293694bde37\_img.jpg\) NET2COM.EXE: Modem Pooling](#)
- [!\[\]\(70b9adae95aa76ce55f26a9fb944efce\_img.jpg\) NETSVC.EXE: Command-line Service Controller](#)
- [!\[\]\(7b780ea9fe954e20f3a1890df3f944e2\_img.jpg\) NETWATCH.EXE: Net Watcher](#)
- [!\[\]\(6ed942f5426e79452fae08b09c55d70a\_img.jpg\) PERMS.EXE: File Access Permissions per User](#)
- [!\[\]\(7ac021fc5aa83a899c22bb9c073b3a95\_img.jpg\) RASUSERS.EXE: Enumerating Remote Access Users](#)
- [!\[\]\(ac644161d49341d629aea2711a2e6e9e\_img.jpg\) REMOTE.EXE: Remote Command Line](#)
- [!\[\]\(e0abb7c68acc16c8045492bb231c4b01\_img.jpg\) SFMATALK.SYS: AppleTalk Protocol and SFM Administrator Tools](#)
- [!\[\]\(f7ae7c0fb463c75dd672b3ca6e6deab8\_img.jpg\) SLEEP.EXE: Batch File Wait](#)
- [!\[\]\(a111623bd72ed30a4b87ae6ecc564347\_img.jpg\) SNMPUTIL.EXE: SNMP Browser](#)
- [!\[\]\(0f6f994ab4c227dedcb0120d2e4578f9\_img.jpg\) SRVMGR.EXE: Server Manager](#)
- [!\[\]\(2038f55e2fb2bf049e5d18738c9b6749\_img.jpg\) UPEDIT.EXE: User Profile Editor](#)
- [!\[\]\(8b3bb7af4c88c70f913e0f53c69b584b\_img.jpg\) USRMGR.EXE: User Manager for Domains](#)
- [!\[\]\(de99c425ce328ea1bb5f0281f304fdb8\_img.jpg\) WINAT.EXE: Command Scheduler](#)
- [!\[\]\(57f21dc2759cc317217f298e29d861b3\_img.jpg\) WINVTP.EXE: Windows NT Communications](#)
- [!\[\]\(498433ed7a9755cf68896c0f712aed1a\_img.jpg\) Desktop Tools](#)
- [!\[\]\(383227c1dec52b0f22f83f97aa406e1e\_img.jpg\) ANIEDIT.EXE: Animated Cursor Creator](#)
- [!\[\]\(65aedb68fa51e83ccf661bb7a3a0a21f\_img.jpg\) IMAGEDIT.EXE: Image Editor](#)
- [!\[\]\(40679573b940492a30dc418063bfdb02\_img.jpg\) SCHEMES.CPL: Animated Cursors and Cursor Schemes](#)
- [!\[\]\(4b9bfc11ce7d3ddc27e5e7a95c7e6922\_img.jpg\) TOPDESK.EXE: Multiple Desktops](#)
- [!\[\]\(9704eb012b488ac816df3fdaa985907b\_img.jpg\) File System Tools](#)
- [!\[\]\(e5a15722f65d1beae018a09aa6d476fd\_img.jpg\) COMPRESS.EXE: File Compress](#)
- [!\[\]\(0bf1759a081fb406e6ad1e4591b0458a\_img.jpg\) DELPART.EXE: Delete Partitions](#)
- [!\[\]\(97747483c2b9ecfe7be85588978513e3\_img.jpg\) DIRUSE.EXE: Directory Disk Usage](#)
- [!\[\]\(8b972688b8677de8b6973a85ef0f994f\_img.jpg\) EXETYPE.EXE: Finding the Executable Type](#)
- [!\[\]\(39a7299013c706e78224340d170a1977\_img.jpg\) SCOPY.EXE: File Copy with Security](#)
- [!\[\]\(141ba93fb40a44248d5dabce42c9b33d\_img.jpg\) WINDIFF.EXE: File and Directory Comparison](#)
- [!\[\]\(a13b0ea4cadba7ae6fb3d153b137140e\_img.jpg\) Performance and System Monitoring Tools](#)
- [!\[\]\(98ee1b4bf2609386400763e5075e28b7\_img.jpg\) BROWMON.EXE: Browser Monitor](#)
- [!\[\]\(f91eb58f34b3404e056b491624e9d0fe\_img.jpg\) BROWSTAT.EXE: Browser Status](#)
- [!\[\]\(1d2d8c824f377047f301ba712f01efb3\_img.jpg\) DOMMON.EXE: Domain Monitor](#)
- [!\[\]\(74d25cccc4ea276a1a9fd0fa1991c1ab\_img.jpg\) DRIVERS.EXE: Device Driver Information](#)
- [!\[\]\(4cb6340900379711e854b58ff215072a\_img.jpg\) FTPCTRS: FTP Server Performance Counters](#)
- [!\[\]\(0bb93e3d6aa370dcb627fef61e4351a8\_img.jpg\) PERFMTR.EXE: Performance Meter](#)
- [!\[\]\(61f1e2b35cc205de34de8920a3ee425f\_img.jpg\) PMON.EXE: Process Resource Monitor](#)
- [!\[\]\(33b43ed79959d6acc22dee6b3e825a17\_img.jpg\) PSTAT.EXE: Process and Thread Status](#)
- [!\[\]\(43ee69106a57bc49d37d6d85527340be\_img.jpg\) PVIEWER.EXE: Process Viewer](#)
- [!\[\]\(1fcabdb41dd22f003c5713956e146a2b\_img.jpg\) QSLICE.EXE: CPU Usage by Processes](#)
- [!\[\]\(5af8e3dd4e0d8636221cab4c28d64f3f\_img.jpg\) SMBTRACE.EXE: Server Message Block Tracer](#)
- [!\[\]\(08a5b947c0bfb9accd9cfe5567535ba6\_img.jpg\) Registry Tools](#)
- [!\[\]\(904c4baa5d22fc8b786857a21de4ea64\_img.jpg\) GRPTOREG.EXE and REGTOGRP.EXE: Program Manager Groups to Registry](#)
- [!\[\]\(cabfe3e6bd010b86c753cf14b5d0f525\_img.jpg\) REGBACK.EXE: Registry Backup](#)
- [!\[\]\(7595e3bc8973dce9997bee193cc8fab3\_img.jpg\) REGENCY.HLP: Windows NT Registry Entries](#)
- [!\[\]\(241bae70b0f0e3d4404d4181e67b9a52\_img.jpg\) REGINI.EXE: Registry Change by Script](#)
- [!\[\]\(6186be847d41767d8af6dc2c84c926a2\_img.jpg\) REGREST.EXE: Registry Restoration](#)
- [!\[\]\(71fb8b30e07c2f3e017361de92a5c8d9\_img.jpg\) Setup and Troubleshooting Tools](#)
- [!\[\]\(593453a33b64b591f9c972fbcfd55dd7\_img.jpg\) JUL93HCL.HLP: Hardware Compatibility List](#)

-  NTCARD.HLP: Adapter Help
-  NTDETECT.COM: Startup Hardware Detector
-  REPAIR.EXE: Update Emergency Repair Disk
-  TROUBLE.HLP: Troubleshooting Flowcharts
-  WINNTP.EXE: Computer Profile Setup
-  Tools for Developers
-  OS2.API
-  POSIX Utilities
-  UCONVERT.EXE: Unicode Converter



Aniedit

## **ANIEDIT.EXE: Animated Cursor Creator**

Use the animated cursor creator to draw and edit frames to create animated cursors.

### **To use the animated cursor creator:**

- 1 Double-click the Animated Cursor Creator icon in the Resource Kit program group.
- 2 From the File menu, choose New to create a new animated cursor sequence, or choose Open to edit an existing animated cursor.
- 3 Use the toolbar or menu commands to edit frames:



File New



Edit Cut



Edit Delete



File Open



Edit Copy



New Frame



File Save



Edit Paste



Edit Frame

Other menu command: Import Frame

- 4 In the edit control, select the number of jiffies (1/16th seconds) that the current frame will appear during the animation sequence.



### Tips for Using Animated Cursor Creator

Files required for the animated cursor creator:

ANIEDIT.EXE  
IMAGEDIT.EXE  
IMAGEDIT.HLP

### **Tips for Animated Cursor Creator**

- Use SHIFT+CLICK or CTRL+CLICK to select multiple frames. Then you can change the speed for several frames at once.
- If you select multiple frames that have different frame speeds, clicking the Up or Down arrow in the Jiffies edit box will change their speed proportionally. That is, they will all increase or decrease  $x\%$ , but an actual number will not appear in the Jiffies box.

### **Import Frame**

Imports a .BMP image to serve as a frame in the sequence for the current file.

**Edit Cut**

Cuts a selected frame and places it on the Clipboard, so that it can be pasted in another position in the sequence or in another animated cursor file.

**Edit Copy**

Copies a selected frame and places it on the Clipboard, so that it can be pasted in another position in the sequence or in another animated cursor file.

**Edit Paste**

Pastes a frame from the Clipboard in the indicated position in the sequence.



**File New**

Displays an empty Frame sequence so that you can create a new animated cursor file.

**File Open**

Displays the Open dialog box so that you can select an animated cursor (.ANI) file to edit.

**File Save and File Save As**

Saves the sequence under the current filename (Save) or under a filename that you supply (Save As).

**Edit Delete**

Deletes a selected frame from the animation sequence.

**Edit Frame**

Runs Image Editor so that you can alter the currently selected frame.

**New Frame**

Runs Image Editor so that you can create a new frame for the animation.



Imagedit

## **IMAGEDIT.EXE: Image Editor**

The Image Editor allows you to create and edit cursors and icons for VGA, monochrome, and other display devices. Use the Image Editor to create custom cursors and icons for Program Manager.

### **To use Image Editor:**

- 1 Double-click the Imagedit icon in the Resource Kit program group.  
Or choose New Frame or Edit Frame in [Animated Cursor Creator](#).
- 2 Press F1 to get online Help for Image Editor.  
When you create, open, or edit a .CUR or .ICO file, the Image Editor allows you to select an image type or an existing image in a file.

Files required for the Image Editor:

IMAGEDIT.EXE  
IMAGEDIT.HLP



Cursors

## **SCHEMES.CPL: Animated Cursors and Cursor Schemes**

You can install a Cursors Schemes icon in the Control Panel to replace the Cursors icon included with Windows NT. The new Cursor Schemes icon lets you group cursors into schemes, similar to the color schemes you can create using the Color icon. Over 150 cursors and cursor schemes are included on the Resource Guide disks to help you customize your desktop.

### **To install the Cursors Schemes icon in Control Panel:**

- 1 Close Control Panel. (You cannot install SCHEMES.CPL if Control Panel is open.)
- 2 Make the \RESKIT directory the current directory, and then run SCHEMES.BAT.

This batch file installs SCHEMES.CPL in the *SystemRoot\SYSTEM32* directory, and renames CURSORS.CPL to CURSORS.OLD so that there can be no conflict between the old and new cursors tools in Control Panel.

### **To define cursor schemes:**

- 1 In the Control Panel, choose the Cursor Schemes icon.
- 2 Select a type of cursor, and double-click it (or choose the Browse button) to bring up the list of cursors. Repeat this process for each type of cursor you wish to change.

### **To save a cursor scheme:**

► After making changes, click the Save Scheme button. Type the name of your scheme and press Enter to save it.

### **To remove a cursor scheme from the list:**

► Click the Remove Scheme button. When prompted, click Yes if you're sure you want to delete the scheme.

### **To remove the Cursors Schemes tool from the Control Panel:**

► If you no longer want to use the Schemes tool, close Control Panel. Then delete or rename the SCHEMES.CPL file in the *SystemRoot\SYSTEM32* directory, and rename the file CURSORS.OLD to CURSORS.CPL.

Files required for animated cursors and schemes:

SCHEMES.BAT

SCHEMES.CPL

\*.ANI (source files for animated cursors)

\*.CUR (source files for static cursors)





Browmon

## **BROWMON.EXE: Browser Monitor**

The Browser Monitor utility monitors the status of browsers on selected domains. Browsers are shown on a per domain and per transport basis.

### **To use Browser Monitor:**

- 1 Double-click the Browser Monitor icon in the Resource Kit program group.
- 2 Press F1 to get online Help while working in Browser Monitor.

Files required for the Browser Monitor:

BROWMON.EXE  
BROWMON.HLP

## BROWSTAT.EXE: Browser Status

BrowStat is a general purpose, character-based browser diagnostic. Use BrowStat to find whether a browser is running and to find active Microsoft Windows for Workgroups 1.0 (WFW) browsers in Windows NT domains. This utility provides information about the state of the browser in a workgroup, including the name of the master browser.

### To view browser information:

- Type **browstat** *Command Options* at the command prompt.

Where *command* is the whole word or abbreviation from the following list:

<u>ELECT (EL)</u>	Force election on remote domain.
<u>GETBLIST (GB)</u>	Get backup list for domain.
<u>GETMASTER (GM)</u>	Get remote master name (using NetBIOS).
<u>GETPDC (GP)</u>	Get primary domain controller name (using NetBIOS).
<u>LISTWFW (WFW)</u>	List active Windows for Workgroups servers.
<u>STATISTICS (STS)</u>	Dump browser statistics. <u>Example</u>
<u>STATUS (STA)</u>	Display status about a domain. <u>Example</u>
<u>TICKLE (TIC)</u>	Force remote browse master to stop.
<u>VIEW (VW)</u>	Retrieve list of servers on a transport or workgroup.

### Show BrowStat Listing Flags

You can find which transports are on a computer by using the **net config rdr** command and examining the result. In this discussion, *transport* refers to the case-insensitive Windows NT device name for the specified transport, in one of the following formats:

\Device\*transport* (For example: \device\Nbf\_elnk1601)

\*transport*

*transport*

File required for BrowStat:

BROWSTAT.EXE

## Flags in BrowStat Listings

AFP = AFP Server	PBR = Potential Browser
BBR = Backup Browser	PDC = Primary Domain Controller
BDC = Backup Domain Controller	PQ = Print Server
DL = Dial-in Server	S = Server
DMB = Domain Master Browser	SQL = SQL Server
MBC = Member Domain Controller	TS = Time Source
MBR = Master Browser	VMS = Vax VMS Server
NT = Windows NT	W = Workstation
NV = Novell	WFW = Windows for Workgroups
OSF = OSF Server	XN = Xenix

## BROWSTAT ELECT (EL)

This command forces a master browser election on the domain using the transport specified.

Usage:

**browstat elect** *transport domain*

## BROWSTAT GETBLIST (GB)

This command retrieves a list of the backup browsers on the domain with the specified transport.

Usage:

**browstat getblist** *transport [domain] [refresh]*

## BROWSTAT GETMASTER (GM)

This command uses NetBIOS to retrieve the name of the browser master for a transport for a workgroup.

Usage:

**browstat getmaster** *transport domain*

## BROWSTAT GETPDC (GP)

This command uses NetBIOS to retrieve the name of the primary domain controller for a transport for a workgroup.

Usage:

**browstat getpdc** *transport domain*

## BROWSTAT LISTWFW (WFW)

This command finds WFW computers that are currently running the browser. If you have a mixed workgroup-and-domain network, you can disable the browser in Windows for Workgroups.

Usage:

**browstat listwfw** *domain*

## BROWSTAT STATISTICS (STATS or STS)

This command dumps various useful browser statistics. The `\\Server` switch allows it to be pointed to a specific server.

Usage:

**browstat stats** [\server] [clear]

### **BROWSTAT STATUS (STA)**

This command dumps browser status for the specified workgroup on all local transports. It also includes the build number of the browser master and how many servers are in the workgroup.

Usage:

**browstat status** [-v] *workgroup*

### **BROWSTAT TICKLE (TIC)**

This command stops the browse master for the specified workgroup on the specified transport. It can be used to reset a computer that has been determined to be "bad."

Usage:

**browstat tickle** *transport domain*

### **BROWSTAT VIEW (VW)**

Usage:

**browstat view** *transport*

**browstat view** *transport domain* | *server* [/DOMAIN]

**browstat view** *transport server* /DOMAIN *domain*

This command retrieves the list of servers or domains for a specified server. for a specified transport or workgroup.

### **BROWSTAT STATS Example**

C>BROWSTAT STATS

Browser statistics since 20:40:54.705 on 6/9/1993	Time statistics were last cleared
NumberOfServerEnumerations:	237
NumberOfDomainEnumerations:	235
NumberOfOtherEnumerations:	6
NumberOfMailslotWrites:	1974
NumberOfServerAnnouncements:	0
NumberOfDomainAnnouncements:	0
NumberOfElectionPackets:	27425
NumberOfGetBrowserServerListRequests:	0
NumberOfMissedGetBrowserServerListRequests:	0
NumberOfMissedServerAnnouncements:	0
NumberOfMissedMailslotDatagrams:	0
NumberOfFailedMailslotAllocations:	0
NumberOfFailedMailslotReceives:	0
NumberOfFailedMailslotWrites:	0
NumberOfFailedMailslotOpens:	37415
NumberOfFailedServerAnnounceAllocations:	0
NumberOfMasterAnnouncements:	0
NumberOfIllegalDatagrams:	0

Meaning:

NumberOfServerEnumerations: = Number of browse requests for servers

NumberOfDomainEnumerations: = Number of browse requests for domains

NumberOfOtherEnumerations: = Number of "other" browse requests

NumberOfMailslotWrites: = Number of mailslot writes received  
NumberOfServerAnnouncements: = Number of server announcements received  
NumberOfDomainAnnouncements: = Number of domain (workgroup) announcements received  
NumberOfElectionPackets: = Number of browser election packets received  
NumberOfGetBrowserServerListRequests: = Number of GetBrowserServerList received  
NumberOfMissedGetBrowserServerListRequestNumber of Missed GetBrowserServerList requests  
NumberOfMissedServerAnnouncements: = Number of server announcements dropped  
NumberOfMissedMailslotDatagrams: = Number of mailslot writes that were dropped  
NumberOfFailedMailslotAllocations: = Number of mailslot writes dropped due to memory  
NumberOfFailedMailslotReceives: = Number of mailslot writes dropped due to transport  
NumberOfFailedMailslotWrites: = Number of mailslot writes failed in the file system  
NumberOfFailedMailslotOpens: = Number of mailslot writes for mailslots not present  
NumberOfFailedServerAnnounceAllocations: = Number of server announcements dropped due to  
memory  
NumberOfMasterAnnouncements: = Number of WAN master browser announcements  
NumberOfIllegalDatagrams: = Number of illegal datagrams received

## BROWSTAT STATUS Example

```
C>BROWSTAT STATUS NTLAN
Status for domain ntlan on transport \Device\Nbf_Lance01
  Master browser name is: MSTRBROWSER1
    Master browser is running build 1.511.1
  3 backup servers retrieved from master MSTRBROWSER1
    \\MYPC
    \\MSTRBROWSER1
    \\BACKUPSVR1
  There are 20 servers in domain ntlan on transport \Device\Nbf_Lance01
  There are 1074 domains in domain ntlan on transport \Device\Nbf_Lance01
Status for domain ntlan on transport \Device\Streams\NWNBLINK
  Master browser name is: MSTRBROWSER1
    Master browser is running build 1.511.1
  3 backup servers retrieved from master MSTRBROWSER1
    \\BACKUPSVR1
    \\MSTRBROWSER1
    \\MYPC DEBUG
  There are 8 servers in domain ntlan on transport \Device\Streams\NWNBLINK
  There are 36 domains in domain ntlan on transport \Device\Streams\NWNBLINK
Status for domain ntlan on transport \Device\Streams\NBT
  Master browser name is: MSTRBROWSER1
    Master browser is running build 1.511.1
  3 backup servers retrieved from master MSTRBROWSER1
    \\MSTRBROWSER1
    \\BACKUPSVR1
    \\MYPC
  There are 11 servers in domain ntlan on transport \Device\Streams\NBT
  There are 101 domains in domain ntlan on transport \Device\Streams\NBT
```

## CHOICE.EXE: User Input for Batch Files

CHOICE prompts the user to make a choice in a batch program by displaying a prompt and pausing for the user to choose from among a set of keys. You can use this command only in batch programs.

### To use Choice:

- ▶ In a batch file, include **choice** with the appropriate switches:

**choice** [/c[:]choices] [/N] [/S] [/T[:]c,nn] [text]

Where:

**/c[:]choices**

Specifies allowable keys in the prompt. When displayed, the keys will be separated by commas, will appear in brackets ([ ]), and will be followed by a question mark. If you don't specify the **/C** switch, CHOICE uses YN as the default (which displays as [Y, N]). The colon (:) is optional.

**/N**

Causes CHOICE not to display the prompt. The text before the prompt is still displayed, however. If you specify the **/N** switch, the specified keys are still valid.

**/S**

Causes CHOICE to be case sensitive. If the **/S** switch is not specified, CHOICE will accept either upper or lower case for any of the keys that the user specifies.

**/T[:]c,nn**

Causes CHOICE to pause for a specified number of seconds before defaulting to a specified key. The values for the **/T** switch are as follows:

- **c** = the character to default to after *nn* seconds. The character must be in the set of choices specified in the **/C** switch.
- ▢ **nn** = the number of seconds to pause. Acceptable values are from 0 to 99. If 0 is specified, there will be no pause before defaulting.

**Text**

Specifies text you want to be displayed before the prompt. Quotation marks are necessary only if you include a switch character (*/*) as part of the text before the prompt. If you don't specify text, CHOICE displays only a prompt.

▢ [Choice Examples](#)

▢ [Choice ErrorLevel Note](#)

File required for Choice:

CHOICE.EXE

## **Choice ERRORLEVEL Note**

ERRORLEVEL is set to the offset of the key that the user presses in choices.

The first key you assign returns a value of 1, the second a value of 2, the third a value of 3, and so on. If the user presses a key that is not among the keys you assigned, CHOICE sounds a warning beep (that is, it sends a BEL, or 07h, character to the console).

If CHOICE detects an error condition, it returns an ERRORLEVEL value of 255.

If the user presses CTRL+BREAK or CTRL+C, CHOICE returns an ERRORLEVEL value of 0.

When you use ERRORLEVEL parameters in a batch program, list them in decreasing order.



## CHOICE Examples

### What the user sees when you use CHOICE in a batch file...

If you use the following syntax in a batch file:

```
choice /c:ync
```

then the user sees the following when CHOICE is started:

**[Y,N,C]?**

If you add text to the syntax:

```
choice /c:ync Yes, No, or Continue
```

then the user sees the following when CHOICE is started:

**Yes, No, or Continue [Y,N,C]?**

### What the user sees if you leave out a prompt...

If, as in the following example, you use the /N switch to leave out the prompt in a batch program:

```
choice /n Yes, No, or Continue?
```

then the user sees only the text you specified when CHOICE is started:

**Yes, No, or Continue?**

### What the user sees if you use the /T switch...

If you use the following syntax in a batch program:

```
choice /c:ync /t:n,5
```

then the user sees the following when CHOICE is started:

**[Y,N,C]?**

If, after 5 seconds, the user hasn't pressed a key, CHOICE chooses N and returns an ERRORLEVEL value of 2. If the user presses a key before 5 seconds, CHOICE returns the value corresponding to the user's choice.

## COMPRESS.EXE: File Compress

This command-line utility can be used to compress one or more files. Compress can also be used to make custom Setup floppies as described in Chapter 3 of the *Windows NT Resource Guide*.

### To use Compress:

- At the command line, type **compress** with the appropriate switches:

**compress [-r] [-d] source destination**

**compress -r [-d] source [destination]**

Where:

**-r**

Renames compressed files.

**-d**

Updates compressed files only if out of date.

*source*

Specifies the source file. The \* and ? wild cards can be used.

*destination*

Specifies the destination file or path. The *destination* can be a directory. If *source* specifies multiple files and **-r** is not specified, then *destination* must be a directory.

You can type **compress -?** at the command prompt to get help.

File required for Compress:

COMPRESS.EXE


## DELPART.EXE: Delete Partitions

DelPart deletes one or more partitions, including NTFS partitions. This allows you to remove an NTFS partition without reformatting your hard drive. (You cannot use the MS-DOS **fdisk** command to remove NTFS partitions.)

DelPart deletes whole partitions, and also can delete individual logical drives in extended partitions.

Because DelPart is an MS-DOS-based program, you can run this application from a bootable floppy disk if the computer also has MS-DOS installed.

### To use DelPart:

 At the command prompt, type **delpart**

**Important:** DelPart also provides full UNDO capability. You can delete all partitions on a hard disk, and physically write that information out, then choose Edit Undo and be able to restore the hard disk partitioning scheme back to what it was before DelPart was run.

**Note:** You can also delete partitions by running the Windows NT Setup program from disks, CD, or the network. You can quit Setup after completing the early steps to delete partitions or create new ones.

File required for DelPart:

DELPART.EXE

## DIRUSE.EXE: Directory Disk Usage

DirUse shows disk space used per directory. You can use this utility to check how disk space is being used in users' home directories. If you run DirUse while logged on as a member of the Administrators group, you can check the use of disk space in directories on an NTFS partition, even if you don't have access rights to those directories.

### To use DirUse:

At the command prompt, type **diruse** with the appropriate switches:

**diruse** [/S] [/Q:#] [/M | /K | /B] [/A] [/D] [/O] [\*] [*dirs*]

Where:

**/S**

Specifies whether subdirectories are included in the output.

**/Q:#**

Marks directories that exceed the specified size with a "!". (If **/M** or **/K** is not specified, then bytes is assumed.)

**/M**

Displays disk usage in megabytes.

**/K**

Displays disk usage in kilobytes.

**/B**

Displays disk usage in bytes.

**/A**

Specifies that an alert is generated if specified sizes are exceeded. The Alerter service must be running, and the alert appears only when you are using DirUse.

**/D**

Displays only directories that exceed specified sizes.

**/O**

Specifies that subdirectories are not checked for specified size overflow.

**/\***

Uses the top-level directories residing in the specified *dirs*.

*dirs*

Specifies a list of the paths to check.

You can type **diruse /?** or **diruse /H** to see online Help. Parameters can be typed in any order, and you can use the "-" symbol rather than the "/" symbol. For example:

**diruse c:\winnt -s -m -q:1.5**

File required for DirUse:

DIRUSE.EXE



Dommon

## **DOMMON.EXE: Domain Monitor**

Domain Monitor monitors the status of servers in a domain and the secure channel status to the domain controller and to domain controllers in trusted domains. Domain Monitor displays various status errors, plus the domain controller name and a list of trusted domains.

### **To use the Domain Monitor:**

- 1 Double-click the Domain Monitor icon in the Resource Kit program group.
- 2 Press F1 to get online help while working in Domain Monitor.

Files required for the Domain Monitor:

DOMMON.EXE  
DOMMON.HLP

## DRIVERS.EXE: List Loaded Drivers

The Drivers tool displays character-based information about the installed device drivers. There are no command-line arguments.

### To use the Drivers tool:

 Type **drivers | more** at the command prompt.

#### Format for Drivers data:

<i>ModuleName</i>	The drivers filename.
<i>Code</i>	The executable code in the <u>image</u> .
<i>Data</i>	The non-BSS data in the image.
<i>Bss</i>	The .BSS section from the image. This is data that is initialized to 0.
<i>Paged</i>	The size of the data that is paged.
<i>Init</i>	The size of the file on disk.
<i>LinkDate</i>	The date that the driver was linked.

File required for Drivers:

DRIVERS.EXE

## Format for Drivers data

ModuleName   Code   Data   Bss   Paged   Init   LinkDate

## DUMPEL.EXE: Dump Event Log

Dump Event Log is a command-line utility that can be used to dump an event log for a local or remote system into a tab-separated text file. This utility can also be used to filter for certain event types or to filter out certain event types.

### To use Dump Event Log:

At the command prompt, type **dumpel** with the appropriate switches:

**dumpel [-s server] [-f file] [-l log [-m source] [e n1 n2 n3...] [-r] [-t]**

Where:

#### **-s server**

Specifies the server for which you want to dump the event log. Leading backslashes on the servername are optional.

#### **-f file**

Specifies the filename for the output file. The default is STDOUT.

#### **-l log**

Specifies which log (system, application, security) to dump. If an invalid *logname* is specified, the application log will be dumped.

#### **-m source**

Specifies which source (such as Rdr, Serial, ...) to dump records of. Only one source can be supplied. If this switch is not used, all events are dumped. If a source is used that is not registered in the Registry, the application log will be searched for records of this type.

#### **-e n1 n2 n3 ...**

Filters for event ID *nn* (up to 10 can be specified). If the **-r** switch is not used, only records of these types are dumped; if **-r** is used, all records except records of these types are dumped. If this switch is not used, all events from the specified *sourcename* are selected. You cannot use this switch without the **-m** switch.

#### **-r**

Specifies whether to filter for specific sources or records, or to filter them out.

#### **-t**

If this is specified, individual strings are separated by tabs. If not, they are specified by spaces.

### Dumpel Examples

File required for Dumpel:

DUMPEL.EXE

## Dumpel Examples

To dump the system event log on server \\EVENTSVR to a file named EVENT.OUT, use:

**dumpel -s eventsvr -l system -f event.out**

To dump the local system event log, but only get Rdr events 2013, use:

**dumpel -l system -m rdr -e 2013**

To dump the local application log, and get all events except ones from the Garbase source, use:

**dumpel -l application -m garbase -r**

## EXETYPE.EXE: Finding the Executable Type

ExeType is an MS-DOS-based application that identifies the operating system environment and processor required to run a particular executable file.

### To use ExeType:

- 1 Make sure the EXETYPE.INI file is somewhere in your PATH (if unsure, type **path** at the command prompt).
- 2 Type **exetype filename** at the command prompt.  
The *filename* is the name of the executable file you want information about. ExeType does not support wild cards for filenames.

You can use ExeType to determine:

- ▣ Why an application will not run (perhaps because a subsystem is failing).
- ▣ Whether to force MS-DOS mode if it is a bound OS/2 application.

### Example of ExeType output

Files required for ExeType:

EXETYPE.EXE  
EXETYPE.INI



## EXETYPE.INI File

The format of EXETYPE.INI is as follows:

```
[entry1]
TYPE    string
ADDRESS hex value or **
OFFSET  hex value
MASK    hex value
VALUE   hex value or !<hex value>
ADDRESS hex value or **
OFFSET  hex value
MASK    hex value
VALUE   hex value
```

```
[entry2]
TYPE    string
HEADER  hex value or **
OFFSET  hex value
MASK    hex value
VALUE   hex value
```

Each Entry has one TYPE field and from 1 to 5 groups of ADDRESS, OFFSET, MASK and VALUE fields.

**Note:** All hex values must have an even number of characters (in bytes). If an odd number of nybbles is needed, please pad with zeros.

### **[entry] in EXETYPE.INI**

This denotes the start of a new entry. 80 characters maximum.

### **TYPE in EXETYPE.INI**

A description of the file type, 80 characters maximum.

### **ADDRESS in EXETYPE.INI**

Microsoft .EXE files have the address of the header at a certain location. This value is kept at offset 3c from the beginning of the file. If your entry does not use this, put in \*\* and it will be ignored.

### **OFFSET in EXETYPE.INI**

This hex value denotes the offset from the beginning of the header.

### **MASK in EXETYPE.INI**

This bitmask lets you screen which bits of the bytes you want to look at.

### **VALUE in EXETYPE.INI**

This is the actual value you are comparing to the value of the bits at OFFSET strained through MASK.

The "!" denotes a logical NOT; in other words, this file fits the criteria only if the value in the file does NOT match this value.

### **Example of ExeType output**

```
The file MYFILE.EXE is of the following type:
Windows NT
32 bit machine
Executable File
Built for the Intel 386 processor
Runs under the Windows GUI subsystem
```

## **FIFO.CMD: FIFO Buffer Support for Modems**

The FIFO tool enables FIFO communications for modems that support FIFO buffers.

### **To turn on FIFO communications:**

- Make \RESKIT the current directory, and then type **fifo on** at the command prompt.

For example, a modem that includes a National Semiconductor 16550 AUART chip can take advantage of FIFO buffer support. This capability is not turned on by default in Windows NT, however. If you are using such a modem, you can use the FIFO tool to turn on support.

### **To turn off FIFO communications:**

- Make \RESKIT the current directory, and type **fifo off** at the command prompt.

**Note:** You can also change the value of **ForceFifoEnable** to 1 to turn on FIFO support under this subkey in the Windows NT Registry:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \Serial
        \Parameters
```

Files required for the FIFO tool:

FIFO.CMD  
FIFO\_ON.INI  
FIFO\_OFF.INI  
REGINI.EXE

## FLOPLOCK.EXE: Lock Floppy Disk Drives

FloppyLock is a service that controls access to the floppy drives of a computer. When the service is started on Windows NT, only members of the Administrators and Power Users groups can access the floppy drives. When the service is started on Windows NT Advanced Server, only members of the Administrators group can access the floppy drives.

If the FloppyLock service is configured to start automatically, the lock stays in place even after the computer is restarted. This service can be used to prevent the unauthorized installation of software or the introduction of viruses via the floppy disks.

To use the FloppyLock service, you must first install FLOPLOCK.EXE as a service on every computer where you want to lock the floppy drives. Then use Control Panel to configure this service to start automatically.

Installing the FloppyLock service is a separate task after you install the Resource Kit tools.

### To install the FloppyLock service:

■ At the command prompt, type **instsrv *exe-location***

Where:

*exe-location* is a fully qualified path and drive letter that specifies a fixed, local drive that contains FLOPLOCK.EXE. For example: **instsrv c:\reskit\floplock.exe**

### To configure the FloppyLock service:

- 1 In Control Panel, choose the Services icon, and select the FloppyLock service name.
- 2 Choose the Startup button, and select System Account (to ensure that the FloppyLock service starts properly).
- 3 If you want the service to start automatically, select Automatic, to ensure the service will work across restarts and logoffs.
- 4 Restart the computer, log on as Guest, and try to access the floppy drives as a test.

### To remotely turn on or off the FloppyLock service on a computer:

■ Use Server Manager to remotely stop and start the FloppyLock service when you are logged on as an Administrator. Stopping the FloppyLock service unlocks the floppy drives; restarting the service locks them again.

### To remove the FloppyLock service:

- 1 Stop the service, using the Services icon in Control Panel.
- 2 At the command prompt, type **instsrv remove**

Files required for FlopLock:

FLOPLOCK.EXE  
INSTSVR.EXE

## FTPCTRS: FTP Server Counters for Performance Monitor

You can install the FTP Server Performance Counters so that you can use the Windows NT Performance Monitor (PERFMON.EXE) to monitor the activity of the Windows NT FTP Server. This can make FTP Server administration more convenience, since Performance Monitor can be used to view the activity on remote Windows NT computers. The counters are used to graph statistics such as bytes transferred per second, connected users, maximum concurrent users, and files transferred.

Installing the FTP Server performance counters is a separate task after you install the Resource Kit tools.

### To install FTP Server performance counters:

- Switch to the \RESKIT directory. From the command prompt, run FTPINST.BAT.

### To remove FTP Server performance counters:

- Switch to the \RESKIT directory. From the command prompt, run REMOVE.BAT.

**Important:** These .BAT files must be run from the RESKIT subdirectory. Also, TCP/IP and the FTP Server must be installed, and LODCTR.EXE, UNLODCTR.EXE, and REGINI.EXE must be on the path.

### FTP Server Attributes That Are Monitored

Files required for FTP Server Performance Counters:

In the user's path:

LODCTR.EXE  
UNLODCTR.EXE  
REGINI.EXE

In the FTPSVR subdirectory:

FTPCTRS.H  
FTPCTRS.INI  
FTPCTRS.REG  
FTPINST.BAT  
REMOTE.BAT

## **FTP Server Attributes That Are Monitored**

The following attributes can be monitored:

### **0. Bytes Sent/sec**

This is the rate that data bytes are sent by the FTP Server.

### **1. Bytes Received/sec**

This is the rate that data bytes are received by the FTP Server.

### **2. Bytes Total/sec**

This is sum of Bytes Sent/sec and Bytes Received/sec. This is the total rate of bytes transferred by the FTP Server.

### **3. Files Sent**

This is the total number of files sent by the FTP Server.

### **4. Files Received**

This is the total number of files received by the FTP Server.

### **5. Files Total**

This is sum of Files Sent and Files Received. This is the total number of files transferred by the FTP Server.

### **6. Current Anonymous Users**

This is number of anonymous users currently connected to the FTP Server.

### **7. Current NonAnonymous Users**

This is number of non-anonymous users currently connected to the FTP Server.

### **8. Total Anonymous Users**

This is total number of anonymous users that have ever connected to the FTP Server.

### **9. Total NonAnonymous Users**

This is total number of non-anonymous users that have ever connected to the FTP Server.

### **10. Maximum Anonymous Users**

This is maximum number of anonymous users simultaneously connected to the FTP Server.

### **11. Maximum NonAnonymous Users**

This is maximum number of non-anonymous users simultaneously connected to the FTP Server.

### **12. Current Connections**

This is the current number of connections to the FTP Server.

### **13. Maximum Connections**

This is maximum number of simultaneous connections to the FTP Server.

### **14. Connection Attempts**

This is the number of connection attempts that have been made to the FTP Server.

### **15. Logon Attempts**

This is the number of logon attempts that have been made to the FTP Server.

## GRPTOREG.EXE and REGTOGRP.EXE: Program Manager Groups to Registry

This tool creates group files for Program Manager and converts them to the Registry for use in Windows NT.

- REGTOGRP.EXE creates a Windows NT .GRP file in the current directory for each of your Program Manager groups.

- GRPTOREG.EXE converts a .GRP file created by RegToGrp into the Registry.

These tools can be used by an administrator to easily give everyone a particular program group without having to use user profiles. For example, you might create a group that contains icons for company tools, a message of the day, the organizational chart, and other special items.

The advantage of this method over creating user profiles with [User Profile Editor](#) is that you can give users a single group and then let the users have control over the rest of their configuration.

### To use RegToGrp:

- Set up the program groups that you want. Then, at the command prompt, type **regtogrp**

**Note:** The **regtogrp** utility can only create .GRP files for use by **grptoreg**. You cannot use the resulting .GRP files in MS-DOS Windows.

### To use GrpToReg:

- In each user's logon script, add the **grptoreg** command to load the desired program group into the user's profile. The command switches include the following:

**grptoreg [/o] [/c] *groupfiles***

Where:

*groupfiles*

Specifies the path to a .GRP file created by **regtogrp**. If you want to specify more than one file, separate filenames with spaces.

**/o**

Specifies that any existing Program Manager group with the same name should be overwritten.

**/c**

Creates a Common group from the *groupfile*; otherwise, a personal group is created. For the **/c** switch to work, you must be logged on as a member of the Administrators group.

**Note:** The **grptoreg** utility can only use .GRP files created by using **regtogrp**. You cannot specify .GRP files from MS-DOS Windows.

Files required for these tools:

GRPTOREG.EXE  
REGTOGRP.EXE

## **JUL93HCL.HLP: Hardware Compatibility List**

The Hardware Compatibility List is a help file that describes the computers and peripherals that are compatible with Windows NT version 3.1.

### **To use Hardware Compatibility Help:**

- 1 Double-click the Hardware Compatibility List icon in the Resource Kit program group.
- 2 Select a topic.

Files required for Hardware Compatibility Help:

JUL93HCL.HLP  
JUL93HCL.IND



## MIBCC.EXE: SNMP MIB Compiler

MibCC compiles SNMP MIB files.

The SNMP service for Windows NT supports multiple MIBs using an extension-agent API interface. Two extension-agent DLLs come with Windows NT to support the Internet MIB II and a subset of the LAN Manager MIB II objects. Other extension-agent DLLs can be added.

### To use MibCC:

At the command prompt, type **mibcc** with the correct switches:

**mibcc [-e] [-l] [-n] [-ofilename] [-t] [-w] [files...]**

Where:

**-eX**

Stop after X errors. (Default = 10)

**-l**

Do not print logo.

**-n**

Print each node as it is added.

**-ofilename**

Output filename. (Default = MIB.BIN)

**-t**

Print the MIB tree when finished.

**-wX**

Set warning level, where 1=errors and 2=warnings.

**files**

Specifies the source filenames.

You can type **mibcc -?** to display help.

**Note:** The SNMP service does not need to be running on the computer that is executing **mibcc**, but the proper files must be copied to that computer. In particular, MGMTAPI.DLL and MIB.BIN must be present, and TCP/IP must be installed on the computer.

Files required for MIBCC:

MIBCC.EXE

Source files (LMMIB2.MIB, MIB\_II.MIB, and SMI.MIB are included on the Resource Guide disks)  
(SNMP service must be installed)

**SNMP:**

Simple Network Management Protocol (a network management protocol installed with TCP/IP).

**MIB:**

Management Information Base (the entire set of objects that a protocol or service uses).



## Net2set **NET2COM.EXE: Modem Pooling**

Net2Com allows MS-DOS or Microsoft Windows 3.x clients to use modems attached to a Windows NT server for outbound modem services. You can use WINVTP.EXE, the Windows NT communications tool with Net2Com. Windows NT clients can also be used, but only via Windows or MS-DOS-based terminal emulation products such as DynaComm from Futuresoft.

**Note:** You cannot use Windows Terminal with Net2Com. Net2Com only works with software that talks directly to NetBIOS.

Installing Net2Com is a separate task from installing the Resource Kit tools.

### **To install Net2Com from the Resource Guide disks:**

- 1 Place the Resource Guide disk #2 in the drive. Then make that the active drive -- for example, by typing **a:** at the command prompt.

Or place the Resource Guide compact disc in the drive. Then switch to the directory that contains the utilities -- for example, *driveletter:\i386* for an x86-based computer.

- 2 At the prompt, type **net2set.exe**
- 3 Follow the directions on screen.

The Net2Com setup program installs all the necessary files in the directory you specify (typically, *SystemRoot\SYSTEM32\NET2COM*).

### **To start the Net2Com service on the Windows NT server:**

- ☐ Choose the Services icon in the Control Panel, select Net2Com, and choose the Start button.  
Or at the command prompt, type **net start net2com**  
About Using Net2Com

### **To remove Net2Com:**

- 1 Choose the Services icon in Control Panel, and stop the Net2Com service.
  - 2 In File Manager, double-click NET2SET.EXE.
  - 3 Choose the Remove button.
- ☐ Registry Values for Net2Com

Files required for Net2Com:

NCB2PTH.DLL  
NET2ALL.DLL  
NET2COM.EXE  
NET2MST.DLL  
NET2PSP.DLL  
NET2SET.EXE (setup program)  
PTH2COM.DLL  
NET2COM.INF

## About Using Net2COM

The client establishes a NetBIOS session between the client computer and the Windows NT computer running the Net2Com service. The NetBIOS session is considered "Raw" in that no higher level protocol is used.

Many terminal emulators that claim support for NetBIOS sessions use a proprietary protocol known as ACS. These terminal emulators will not work with Net2Com.

No formal testing has been done on Net2Com to verify which communications packages work with it. The only communication packages that will work with this version of Net2Com are those that talk directly to NetBIOS. Of these, Dynacomm 3.0z has gotten the most testing, and WINVTP.EXE has been used with some success.

## Registry Values for Net2Com

The following two Registry values can be set using REGEDT32.EXE to configure Net2Com.

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services  
    \Net2Com\ComMechDLL

### Value: COM

This is a multiple-line entry. The fifth line (first is line 1) is used to tell Net2Com the COM port that will be used by default. If this line is not present, then Com2 is used by default.

### Value: NetBIOS

This is a multiple-line entry. The fifth line (first is line 1) is used to tell Net2Com the NetBIOS name that will be used by Net2Com by default. This name will be the used by the client software to establish a connection to this Net2Com server. By default, this value is **LIONHEART**. The value can be any combination of letters and numbers (must start with a letter) and be 15 characters or less.

## NETSVC.EXE: Command-line Service Controller

You can use the Command-line Service Controller to remotely start, stop, and query the status of services from the command line.

### To use the Command-line Service Controller:

- At the command prompt, type **netsh** with the appropriate syntax and switches, in any order:  
**netsh servicename \\computername switch**

Where:

*servicename*

The name of the service you want to control. You can enter either the service names as defined in the Registry or the display names, as shown when you choose the Services icon in Control Panel. If the service name has embedded spaces, place quotes around the service name.

*computername*

The name of the computer whose services you want to control.

*switch*

**/query, /start, /stop, /continue, /list, or /pause.** No *servicename* is required when you use the **/list** switch. All switches are case-insensitive.

For example:

**netsh /query \\anniep "Network DDE"**

You can type **netsh** with **/?** or **/help** to get help at the command prompt.

You can use **netsh** without special rights, unless the owner of a computer has blocked out all users. To use the **/stop** or **/start** options, you must have an account on the target computer with privileges that allow starting or stopping services.

Because **netsh** uses the service names as listed under HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services in the Registry, the service names displayed with **netsh** can be slightly different from those displayed with **net start**.

- [Notes on Netsh](#)

File required for the Command-line Service Controller:

NETSVC.EXE

## Netsvc Notes

Not all services can be stopped directly (such as the Workstation service), although you can query it. If the user has a lot of active connections, you cannot force the service to shut down remotely, although you can pause or query it. Other services have services that are dependent on them; trying to shut down such services will fail unless the dependent services are shut down first.

For example, the ClipBook Server requires the Network DDE Service; so you cannot shut down NetDDE without first shutting down ClipBook Server. This is true even with **net stop**.

Therefore, you must be familiar with the services you are shutting down.

If you try to stop a non-stoppable service, **net** simply reports that the service is running. For example:

```
net /stop \\machine TargetService
```

Reply:

```
Service is running
```

This indicates that the service rejected the request to stop. It is impossible for **net** to know why the service refuses to be stopped, since the reason varies from service to service.

Likewise, some services cannot be PAUSED or CONTINUED, because PAUSE or CONTINUE logic is not included in the service program. If you try to pause a non-pausable service, **net** reports the "Service is running" immediately afterwards.



Net Watcher

## NETWATCH.EXE: Net Watcher

The Net Watcher tool shows who is connected to shared directories.

**Note:** The Server service must be started to use Net Watcher. You must be logged on as a member of the Administrators group.

### To use Net Watcher:

- 1 Double-click the Net Watcher icon in the Resource Kit program group.  
Or at the command prompt, type **netwatch** *\\computername*
- 2 From the Options menu, choose the Show Open Files, Show Hidden Shares, or Show In Use Shares Only command to indicate the network connections you want to watch.

Some tips for using Net Watcher:

- You can keep Net Watcher open on the desktop with no menu bar or title bar, by choosing No Menu Bar and Title from the Options menu.
- Net Watcher updates the list of connected users every 30 seconds or whenever you press F5.
- You can view more details on a resource by double-clicking it or by selecting it and pressing ALT+ENTER.

File required for Net Watcher:

NETWATCH.EXE

## **NTCARD.HLP: Adapter Help**

The Windows NT Adapter Help file was created by the Microsoft Product Support Services (PSS) group to assist you in the setup of network adapters, SCSI adapters, and sound cards for Windows NT. This Help file provides IRQ, I/O Base, RAM Base Address and other settings, along with illustrations that show the location for jumper settings on the cards.

### **To use Windows NT Adapter Help:**

- 1 Double-click the Adapter Help icon in the Resource Kit program group.
- 2 Click a card name to see more information.
- 3 Click the Control Panel, Setup, Notepad, or RegEdit button at the top of the Help window to run the related application.

**Note:** This Help file is not intended as a replacement for the documentation provided with your adapter(s). It is provided as a convenience, with the hope that it will help you get your adapter(s) configured more quickly. For information not covered in NTCARD.HLP, consult the documentation supplied with your adapter.

The products described in NTCARD.HLP are manufactured by vendors independent of Microsoft; Microsoft makes no warranty, implied or otherwise, regarding these products' performance or reliability.

File required for Windows NT Adapter Help:

NTCARD.HLP



### Network Adapter

3Com	DEC®	Madge	Proteon	Thomas Conrad
Amplicard	Everex™	NCR	Pure Data	Toshiba®
Artisoft	HP®	NetWorth	Racal	UngermanBass®
Compaq®	IBM®	Novell®	Racore	WD(SMC)
DCA	Intel®	Olicom	SMC®	

### SCSI Adapter

Adaptec™	Future Domain	UltraStor
BusLogic	NCR	
DTP	Trantor	

### NTDETECT.COM: Startup Hardware Detector

NTDetect displays system hardware information at startup time for x86-based computers. This tool dumps the hardware information structures passed to the kernel when the system is started.

Installing NTDetect is a separate task after you install the Resource Guide utilities.

#### To install NTDetect:

- 1 At the command prompt, switch to the \RESKIT directory where the Resource Guide utilities are installed, and type **install**
- 2 Shut down and restart the computer to see the results.  
NTDetect displays information about the system component, bus/adaptor component, disk geometry, ROM blocks, keyboard and COM port and parallel port components, mouse component, and floppy component. When detection is complete, you are asked to press any key to display hardware information.
- 3 Keep pressing keys to view information. This information is similar to what is stored under HKEY\_LOCAL\_MACHINE\HARDWARE\Controllex in the Registry.  
[Example of NTDetect information for the system component](#)
- 4 After all information for detected hardware is displayed, you can press the spacebar if you want to use the LastKnownGood control set instead of the current control set to start the system.
- 5 When the system starts, you are asked to press CTRL+ALT+DEL to logon.

**Note:** Under no circumstances should you delete NTDETECT.COM. NTDETECT.COM is a required program for a Windows NT system. The version that is installed by INSTALLD.CMD is a special version that displays diagnostic information. If your system doesn't start correctly, use the Emergency Repair Disk to restore files.

#### To remove the special version of NTDetect:

 Type **install /not** at the command prompt.

Files required for NTDetect:  
INSTALLD.CMD  
NTDETECT.COM

## NTDetect Information for the System Component

```
Current Node: 00050000
Type: MaximumType
  Child: 00050045
  Parent = 00000000
  Sibling: 00000000
  ConfigurationData = 00000000
IdentifierLength = 0000011
Identifier= AT/AT COMPATIBLE
ConfigdataLength = 00000044
Version = 0000, Revision = 0000
Count = 0002
Type = Device Data
Size = 0000000C
0080 0000 00DB 000 0000 0000 002 0000 000B 0000
0001 0000
```

## OS2API.TXT: API Information

The OS2API.TXT file in the \RESKIT directory contains information for developers, describing which APIs for OS/2 are supported under the first release of Windows NT, and which are not supported.

### To use OS2API.TXT:

- Open the OS2API.TXT file using your favorite editor.

File required:

OS2API.TXT

## **PERFMTR.EXE: Performance Meter**

The Performance Meter displays text-based performance information.

### **To use the Performance Meter:**

- 1 At the command prompt, type **perfmtr**
- 2 Type a command (without pressing ENTER) for any of the following:
  - C** CPU usage
  - F** File cache usage
  - H** Header
  - I** I/O usage
  - L** LPC stats
  - P** Pool usage
  - R** Cache Manager reads and writes
  - S** Server stats
  - V** Virtual memory usage
  - X** x86 VDM (Virtual DOS Machine) usage
  - Q** Quit

File required for the Performance Meter:

PERFMTR.EXE

## PERMS.EXE: File Access Permissions per User

The Perms tool is used to display users' access permissions for a specified file or set of files. To use Perms, you need "Backup files and directories" privileges on the computer where the files are stored, and you must be logged on as a member of the Administrators group for the domain or computer where the user account is defined. Otherwise, "Access denied" errors may occur.

### To use Perms:

At the command prompt, type **perms** with the appropriate switches:

**perms** [*account*] [*path*] [*/i*] [*/s*]

Where:

*account*

Name of the user whose permissions are to be checked, in the format *domain\username* or *computer\username* or local *username*.

*path*

The name of a file or directory in any legal format, including UNC (\\). You can use the \* or ? wild cards.

*/i*

Indicates that Perms is to assume that *account* is interactively logged on to the computer where *path* resides. Without this parameter, Perms assumes the user is logged on over the network and is a member of the Network security group.

*/s*

Checks permissions on files in subdirectories.

*/?*

Displays help for the Perms command.

### Characters in Perms Listings

File required for Perms:

PERMS.EXE

## Characters in Perms Listings

The information returned by Perms uses these characters:

R	Generic Read
W	Generic Write
X	Generic Execute
D	Standard Delete
P	Change Permissions
O	Take Ownership
A	General All
None	No Access
*	The specified account is the owner of the file or directory.
#	A group of which the user is a member owns the file or directory.
?	The user's access permissions cannot be determined.

## PMON.EXE: Process Resource Monitor

PMon is a character-based tool that monitors process resource usage, tracking CPU and memory usage.

### To run PMon:

- Type **pmon** at the command prompt.

### To quit PMon:

- Press CTRL+C or choose Close from the Control menu.

Elements of the PMon format:

%CPU

CPU Time

Mem Usage

Page Faults

Commit Charge

NonP Usage

Paged Usage

Pri

ThdCnt

Image Name

File required for PMon:

PMON.EXE

## PMon format

Thd	Image		Mem	Mem	Page	Flts	Commit	Usage
%CPU	Cpu Time	Usage	Diff	Faults	Diff	Charge	NonP Page	Cnt
Name								

### %CPU

Identifies the percent of total CPU usage associated with a particular process.

### Memory Usage

Identifies the total memory used.

### Page Faults

Page Faults/sec is a count of the Page Faults in the processor. A page fault occurs when a process refers to a virtual memory page that is not in its Working Set in main memory. A Page Fault will not cause the page to be fetched from disk if that page is on the standby list, and hence already in main memory, or if it is in use by another process with whom the page is shared.

### Commit Charge

Displays the size of virtual memory (in bytes) that has been Committed (as opposed to simply reserved). Committed memory must have backing (that is, disk) storage available, or must be assured never to need disk storage (because main memory is large enough to hold it.)

### NonPaged Usage

The size of the Nonpaged Pool, which is a system memory area where space is acquired by operating system components as they accomplish their appointed tasks. Nonpaged Pool pages cannot be paged out to the paging file, but instead remain in main memory as long as they are allocated.

### Paged Usage

The amount of the Page File instance in use.

### Thread Count

Identifies the number of threads for the particular process.


### Image Name

Identifies the executable file name or process associated with a particular item.

## POSIX Utilities


These POSIX utilities are provided for use on x86-based computers.

### To use the POSIX utilities:

 At the command prompt, type the name of the command you want.

The POSIX utilities are not placed in your system's path automatically. If you choose to install these utilities, they will be installed in the \RESKIT\POSIX directory (where \RESKIT is the directory where you

choose to install the Resource Kit utilities). To run the POSIX utilities easily, put the \RESKIT\POSIX directory in your path.

 [POSIX Utility Descriptions](#)

 [POSIX Utility Notes](#)

**Caution:** FIND, MKDIR, and RMDIR are all programs shipped with Windows NT. You probably want to rename the POSIX versions of these so there is no confusion.

Files for the POSIX utilities:

AR.EXE	LS.EXE
CAT.EXE	MAKE.EXE
CC.EXE	MKDIR.EXE
CHMOD.EXE	MV.EXE
CHOWN.EXE	RM.EXE
CP.EXE	RMDIR.EXE
DEVSRV.EXE	SH.EXE
FIND.EXE	TOUCH.EXE
GREP.EXE	VI.EXE
LD.EXE	WC.EXE
LN.EXE	

## POSIX Utility Notes

When you run the POSIX utilities, any parameters consisting of filenames should be in the POSIX format; for example, if you wanted to display the file D:\TEMP\MESSAGE.TXT, in the COMMAND program you might enter:

**type d:\temp\message.txt**

whereas in the **sh** program, you would enter:

**cat //D/temp/message.txt**

Drive letters (which must be capitalized) are preceded by two forward slashes, and directories are preceded by one forward slash. Case sensitivity is also important for POSIX filenames; if you want to search all the C files in a FAT directory for the string "main", you need to enter the following:

**grep main \*.C**

If "\*.c" is entered instead, nothing would match in the FAT directory, due to the automatic uppercasing of filenames on FAT file systems.



**POSIX****Utility      Description**

---

<b>ar</b>	"ARchiver" - LIB front-end. <a href="#">More Info</a>
<b>cat</b>	"conCATenator" - TYPE substitute.
<b>cc</b>	"C Compiler" - CL front-end. <a href="#">More Info</a>
<b>chmod</b>	"CHange MODe" - ATTRIB substitute.
<b>chown</b>	"CHange OWNeR" - no equivalent.
<b>cp</b>	"CoPy" - COPY substitute.
<b>devsvr</b>	<a href="#">used with cc above. More Info</a>
<b>find</b>	<a href="#">file finder</a> - no equivalent.
<b>grep</b>	"Global Regular Expression Print" - FIND sub..
<b>ld</b>	"LoaDer" - LINK front-end. <a href="#">More Info</a>
<b>ln</b>	<a href="#">"LiNker"</a> - no equivalent.
<b>ls</b>	"LiSt" - DIR substitute.
<b>make</b>	NMAKE substitute. <a href="#">More Info</a>
<b>mkdir</b>	"MaKe DIRectory" - MD substitute.
<b>mv</b>	"MoVe" - REN substitute.
<b>rm</b>	"ReMove" - DEL substitute.
<b>rmdir</b>	"ReMove DIRectory" - RD substitute.
<b>sh</b>	<a href="#">"SHell"</a> - COMMAND substitute. <a href="#">More SH Info</a>
<b>touch</b>	<a href="#">Change dates</a> - no equivalent.
<b>vi</b>	vi ("VIsual") clone - EDIT substitute. <a href="#">More VI Info</a>
<b>wc</b>	<a href="#">"Word Count"</a> - no equivalent.

**chown**

Changes owner of files.

**devsrv**

Background process needed for **cc** to function.

**find**

Searches directory tree for files matching given criteria.

**In**

Gives multiple names to the same file.

**sh**

Large subset of Korn shell (no variable arrays).

**touch**

Changes modification date of files.

**wc**

Gives bytes, words, and lines for files.



## VI Command Information

Before using **vi**, you need to customize your environment. Some of the environment variables that you need to set are:

**PATH**=*location of the vi executable*

**TERM**=**ansi**

**TERMCAP**=*location of **termcap** equivalent*

**\_POSIX\_TERM**=**on**

■ The **PATH** variable needs to include the directory where the **vi** executable resides, to be able to run it from anywhere.

■ The **TERM** variable needs to have a value that matches an entry in the **termcap** file; by default, this entry is named "ansi".

■ The **TERMCAP** variable contains the full path of the **termcap** file. You should customize the entry in the **termcap** file according to your typical Command Prompt window size; if you usually use a window that's not 80 columns by 25 lines, change the **co** entry to match the number of columns, and change the **li** entry to match the number of lines. For example, if you normally use a 100 column by 60 line window, the **co** entry would be "co#100" and the **li** entry would be "li#60".

■ The **\_POSIX\_TERM** variable must be set to "on" to use the terminal emulation portion of the POSIX server.

## SH Command Information

Before using the shell, you may need to customize your environment. Some of the environment variables that you may want to set are:

ENV=*location of .profile equivalent*

PATH=*location of sh executable*

TZ=*appropriate timezone*

- The ENV variable contains the full path of a file that contains shell commands that you want to run initially, for example, creating aliases, customizing prompts, and so on.
- The PATH variable needs to include the directory where the **sh** resides, in order to be able to run it from anywhere.
- The TZ variable should be set according to your location on the planet, for example, for residents of Oregon it should be set to "PST8PDT", translated as "normally Pacific Standard Time, eight hours behind Coordinated Universal Time, during the summer Pacific Daylight Time".

A sample .profile file may look something like this:

```
alias a=alias
alias md=mkdir
alias rd=rmdir
alias ua=unalias
set -o vi
PS1='[!]'
```

These commands set four aliases, make available **vi**-style command line editing, and set the prompt to display the current command number inside of brackets. For a more complete view of what the shell can do, check out a good book on the subject, like Bolsky & Korn's *The KornShell Command and Programming Language*, published by Prentice Hall (ISBN 0-13-516972-0).

There is an extension to the **typeset** built-in command, a **"-p"** parameter, representing POSIX customization of an environment variable.

For example, if there was a variable named **GLOP** that contained "C:\users\default\telnet.trm", applying **typeset -p GLOP** to it would result in **GLOP** having the value **"/C/users/default/telnet.trm"**.

The converse command, **typeset +p GLOP**, would "de-POSIX-ize" the variable, leaving the original value in **GLOP**.

Note that Windows NT environment variables that reference other Windows NT environment variables, for example, **SET LOGNAME=%USERNAME%** will not be expanded inside of the shell; such variables will have the unexpanded value--for example, **LOGNAME** would contain **"%USERNAME%"**, not **"Administrator"**.

## DEVSRV, MAKE, CC, AR, and LD Command Information

Before using **devsrv**, **make**, **cc**, **ar**, and/or **ld**, you will need to:

- Go to your system's MSTOOLS directory and copy LINK32.EXE to LINK.EXE.
- Customize your environment. Some of the variables that you will need to set are:

MAKEPATH=*location of default make rules*

PATH=*location of devsrv, make, cc, ar, and ld executables*

POSIX\_INC=*location(s) of default include files*

POSIX\_LIB=*location of default libraries*

SHELL=*full path of Korn shell*

**Note:** These variables need to contain POSIX-style paths, not NTFS-style paths; yet the only paths that need to be added using the System icon in Control Panel are POSIX\_INC and POSIX\_LIB. The best place to enter the other variables is in your .profile (the file pointed to by the ENV environment variable).

Also:

- The MAKEPATH variable needs to contain the directory where the default make rules files (that is, the contents of the MK-RULES.ZIP archive) are installed.

- The PATH variable needs to include the directory where the **devsrv**, **make**, **cc**, **ar**, and **ld** executables reside, in order to be able to run them from anywhere.

- The POSIX\_INC variable needs to have the location(s) of the default include files. If more than one directory is required in POSIX\_INC, then separate the directories with colons.

**Note:** Order is significant; directories listed first will be searched first.

- The POSIX\_LIB variable needs to be set to the location of the default libraries. The SHELL variable needs to point to the full path of the Korn shell executable (including any .EXE suffix).

Two separate windows are required for using **devsrv**, **make**, **cc**, **ar**, and **ld**: one for the **devsrv** program and one for everything else. The **devsrv** program needs to be started before **make**, **cc**, **ar**, or **ld** can be used, and it must also be run on the same drive where the Korn shell is located. The **devsrv** program will not produce any output until either **make**, **cc**, **ar**, or **ld** is run.

In the second window, start the Korn shell as usual (by default, "**sh**"). Go to the directory that you want to develop in, and enter **make**. If your Makefile isn't named "Makefile" or "makefile", you can enter **make -f filename** instead. This will examine your Makefile and start executing the proper commands to build your application.

If you don't have much experience in creating Makefiles, you can refer to the Microsoft NMAKE manual, which should be a fairly close match.

## PSTAT.EXE: Process and Thread Status

PSTAT is a character-based tool that lists all running processes and threads and displays their status.

### To run PStat:

☐ Type **pstat | more** at the command prompt.

The following defines the elements of the PStat output:

<i>pid</i>	<u>Process ID</u>
<i>pri</i>	<u>Priority</u>
<i>tid</i>	<u>Thread ID</u>
<i>cs</i>	<u>Context switch</u>

File required for PStat:

PSTAT.EXE

## PStat Output Format

pid: ## pri: executable name

tid: ## pri: ## cs: ## [wait: Info | running]



## PVIEWER.EXE: Process Viewer

Process Viewer is a Windows-based tool that displays everything you want to know about a running process.

### To run Process Viewer:

- Double-click the Process Viewer icon in the Resource Kit program group.  
Choose an item to view information about that element in Process Viewer:

Memory Detail

Kill Process

Computer

Process

CPU Time

Privileged

User

Process Memory Used

Priority

Thread Priority

Thread Information

File required for Process Viewer:

PVIEWER.EXE

## **Memory Detail**

Choose this button to see details about memory use for the selected process.

## **Kill Process**

Choose this button to stop the selected process.

**Caution:** Do not attempt to kill processes required for running Windows NT. Make sure you understand which program owns a process before attempting to kill it.

## **Computer**

Shows the name of the computer whose processes are currently displayed. Choose the Connect button and complete the dialog box to view processes for a remote computer.

## **Process**

Processor Time is expressed as a percentage of the elapsed time that a processor is busy executing a non-Idle thread. It can be viewed as the fraction of the time spent doing useful work. Each processor is assigned an Idle thread in the Idle process which consumes those unproductive processor cycles not used by any other threads.

## **CPU Time**

Expressed as a percentage of the elapsed time that a processor is busy executing a non-Idle thread. It can be viewed as the fraction of the time spent doing useful work. Each processor is assigned an Idle thread in the Idle process which consumes those unproductive processor cycles not used by any other threads.

## **Privileged Time**

Privileged Time is the percentage of processor time spent in Privileged Mode in non-Idle threads. The Windows NT service layer, the Executive routines, and the Windows NT Kernel execute in Privileged Mode. Device drivers for most devices other than graphics adapters and printers also execute in Privileged Mode. Unlike some early operating systems, Windows NT uses process boundaries for subsystem protection in addition to the traditional protection of User and Privileged modes. These subsystem processes provide additional protection. Therefore, some work done by Windows NT on behalf of your application may appear in other subsystem processes in addition to the Privileged Time in your process.

## **User Time**

User Time is the percentage of processor time spent in User Mode in non-Idle threads. All application code and subsystem code execute in User Mode. The graphics engine, graphics device drivers, printer device drivers, and the window manager also execute in User Mode. Code executing in User Mode cannot damage the integrity of the Windows NT Executive, Kernel, and device drivers. Unlike some early operating systems, Windows NT uses process boundaries for subsystem protection in addition to the traditional protection of User and Privileged modes. These subsystem processes provide additional protection. Therefore, some work done by Windows NT on behalf of your application may appear in other subsystem processes in addition to the Privileged Time in your

process.

### **Process Memory Used**

Working Set is the current number of bytes in the Working Set of this process. The Working Set is the set of memory pages touched recently by the threads in the process. If free memory in the computer is above a threshold, pages are left in the Working Set of a process even if they are not in use. When free memory falls below a threshold, pages are trimmed from Working Sets. If they are needed they will then be soft-faulted back into the Working Set before they leave main memory.

### **Process Priority**

The current base priority of this process. Threads within a process can raise and lower their own base priority relative to the process's base priority.

### **Threads**

Thread State is the current state of the thread. It is 0 for Initialized, 1 for Ready, 2 for Running, 3 for Standby, 4 for Terminated, 5 for Wait, 6 for Transition, 7 for Unknown. A Running thread is using a processor; a Standby thread is about to use one. A Ready thread wants to use a processor, but is waiting for a processor because none are free. A thread in Transition is waiting for a resource in order to execute, such as waiting for its execution stack to be paged in from disk. A Waiting thread has no use for the processor because it is waiting for a peripheral operation to complete or a resource to become free.

### **Thread Priority**

The current base priority of this thread. The system may raise the thread's dynamic priority above the base priority if the thread is handling user input, or lower it towards the base priority if the thread becomes compute bound.

### **Thread Information and Context Switches**

Context Switches/sec is the rate of switches from one thread to another. Thread switches can occur either inside of a single process or across processes. A thread switch may be caused either by one thread asking another for information, or by a thread being preempted by another, higher priority thread becoming ready to run.

Dynamic Priority is The current dynamic priority of this thread. The system may raise the thread's dynamic priority above the base priority if the thread is handling user input, or lower it towards the base priority if the thread becomes compute bound.



Qslice

### **QSLICE.EXE: CPU Usage by Processes**

Quick Slice shows the total CPU used by each process in the system. This tool is similar to PSTAT.EXE, but it presents the information in a graphical format.

Quick Slice displays the Process ID, Image Name, and CPU Usage for active processes.

#### **To use Quick Slice:**

- 1 Double-click the Quick Slice icon in the Resource Kit program group.
- 2 Double-click an image name in the Quick Slice window to see details about the threads of that process.

### To specify the sampling interval for Quick Slice:

- At the command prompt, type **start qslice -t $nnn$**

Where:

**-t $nnn$**

Specifies the display interval in milliseconds. (The default is 500 milliseconds.)

In the Quick Slice window:

- Red bar = kernel time
- Blue bar = user time
- For the main window, the length of the bar represents (CPU usage for a single process) / (CPU usage for all processes currently running in the system) \* 100.
- For the secondary windows, the length of the bar represents (CPU usage for an individual thread) / (CPU usage for all the threads in this process) \* 100.

File required for Quick Slice:

QSLICE.EXE



## RASUSERS.EXE: Enumerating Remote Access Users

RasUsers lets you enumerate all user accounts that have been granted permission to dial in to the network via Remote Access Service (RAS).

### To use RasUsers

At the command prompt, type **rasusers** with the appropriate switches:

**rasusers** *domainName*

**rasusers** *\\serverName*

For example:

**rasusers MYDOMAIN**

**rasusers \\rasserver1**

You can type **rasusers /?** or **rasusers /help** to get help.

File required for RasUsers:

RASUSERS.EXE

## REGBACK.EXE: Registry Backup

RegBack is a backup batch tool that backs up Registry hives to files without use of a tape drive. RegBack allows you to back up parts of the Windows NT Registry called hives, while the system is running and has the hive files open.

You must be logged on as a member of a group that has "Backup files and directories" privileges to use RegBack.

### To use RegBack:

- 1 At the command prompt, type **regback switches**  
Type **regback | more** to see help one screen at a time.
- 2 RegBack will print the command line to edit and use. You can use the cut-and-paste feature of the console to set up a manual backup quickly.  
Examine the output to make sure that RegBack did what you wanted.

### To back up all Registry files in the CONFIG directory:

At the command prompt, type:

**regback DestinationDirectory**

For example: **regback c:\monday.bku** saves all hives in the MONDAY.BKU directory. A warning appears if there are hives that must be backed up manually or if there are errors.

### To manually back up a hive to a named file:

At the command prompt, type:

**regback filename hivetype hivename**

Where:

*Hivetype*

Either **machine** or **users**. Will fail if the *hivetype* isn't **machine** or **users**.

*Hivename*

The name of an immediate subtree of HKEY\_LOCAL\_MACHINE or HKEY\_LOCAL\_USERS. Will fail if *hivename* isn't a hive root.

For example:

**regback c:\special.sav\system machine system**

**regback c:\savedir\prof users s-1-0000-0000-1234**

See Also

[RegBack ErrorLevel](#)

[Notes on RegBack](#)

[Regular Registry Backup on Multiple Computers](#)

[RegBack Example](#)

[REGREST.EXE](#), for restoring hives if you need to use the backed up version of a hive.

File required for RegBack:

REGBACK.EXE

## Regular Registry Backup on Multiple Computers

You can create a batch file that uses RegBack to regularly back up the Registry hives on multiple computers. You can schedule this batch file to run with the AT service or WinAT (the Schedule service must be running in an account that can access the network, and not in the System account).

Here's an example of such a batch file:

```
echo on
rem doback
rem Back up Registry from \CONFIG to network
rem
net use z: \\backupserv\registry
<error text>
cd z:\MyComputerName
md z:\MyComputerName\backreg
copy C:\winnt\system32\config z:\MyComputerName\backreg
regback z:\MyComputerName
```

**Note:** RegBack cannot back up files onto a target directory that already contains files with the same names. So be sure to delete all files in the target directory before you attempt a backup.

Also, remember that RegBack does not backup hives that aren't in active use. You must use the Copy, XCopy, or SCopy command to back up unused hive files, such as unused user profiles.

## Notes on RegBack

- RegBack and RegRest save and reload the entire hive, including access control lists. So it's possible to restore a hive and find that you have different access control lists than before.
- RegBack does not back up hives that aren't loaded. You can just copy these files, since they are not loaded in the Registry.
- RegBack does not automatically back up hives that don't reside in the CONFIG subdirectory (specifically, some user profiles), but it will do so manually. This avoids name conflicts.
- RegBack will stop at the first bug, except for manual hives.
- RegBack will not overwrite existing files; instead, it reports an error.
- RegBack fails if the hive files don't all fit on the target, so often it's best to use RegBack to back up hives to a hard disk directory and then use BACKUP.EXE, or use XCOPY.EXE or SCOPY.EXE to save the backed up hives on floppy disks.
- RegBack does not copy the files in the CONFIG subdirectory that are NOT currently kept open by the Registry. Use XCOPY.EXE or SCOPY.EXE to save inactive hives.
- If you have a tape drive, use the Windows NT Backup program instead. Start Backup by double-clicking its icon in the Administrative Tools program group.

## RegBack Example

```
d:\>regback d:\test
saving SECURITY to d:\test\SECURITY
saving SOFTWARE to d:\test\SOFTWARE
saving SYSTEM to d:\test\SYSTEM
saving .DEFAULT to d:\test\DEFAULT
saving SAM to d:\test\SAM

***Hive = \REGISTRY\USER\S-1-0-17-12-4-56-64744-46-1363
Stored in file \Device\Harddisk0\Partition1\PROFILES\BRYAN000
Must be backed up manually
regback <filename you choose> users S-1-0-17-12-4-56-64744-46-1363
```

```
D:\>regback d:\test\bryan000 users S-1-0-17-12-4-56-64744-46-1363
saving S-1-0-17-12-4-56-64744-46-1363 to d:\test\bryan000
```

```
d:\>regback d:\test
saving SECURITY to d:\test\SECURITY
Save failed
hivebranch='machine', hive='SECURITY', result='0x000000b7'
file='d:\test\SECURITY'
```

## ErrorLevel

The error on exit will be:

- 0 if the procedure was successful
- 1 if a hive that requires manual back up or restoration was encountered
- 2 for all other errors

## REGREST.EXE: Registry Restoration

RegRest restores Registry hive files from backup copies using the Win32 ReplaceKey function.

You must be logged on as a member of a group that has "Backup files and directories" privileges to use RegRest.

### To use RegRest:

- 1 At the command prompt, type **regrest switches**  
Type **regrest | more** to see help one screen at a time.
- 2 RegRest will print the command line to edit and use. You can use the cut-and-paste feature of the console to set up a manual backup quickly.  
Examine the output to make sure that RegRest did what you wanted.  
**Note:** The change takes effect only after the system is restarted.

### To restore all active Registry hive files in the CONFIG directory:

- At the command prompt, type:  
**regback newDirectory saveDirectory**

Where:

*newDirectory*

Specifies the directory for the source of the backed up hive file which will replace a hive file in the CONFIG directory. For each active Registry hive whose file is in CONFIG, attempts to replace the current file with a like-named file from the *newDirectory*.

*saveDirectory*

Specifies the directory where the old hive files in CONFIG are moved to.

For example: **regrest c:\monday.bku c:\install.sav**. A warning appears if there are hives that must be backed up manually or if there are errors.

### To manually restore a hive:

- At the command prompt, type:  
**regrest newFilename saveFilename hivetype hivename**

Where:

*newFilename*

Specifies the backup source filename. RegRest will rename this file and use it to replace the old *hivename* file.

*saveFilename*

Specifies the filename for saving the old *hivename* that is being replaced. The old hive file will be renamed and moved to this location.

*Hivetype*

Either **machine** or **users**. Will fail if the *hivetype* isn't **machine** or **users**.


*Hivename*


The name of an immediate subtree of HKEY\_LOCAL\_MACHINE or HKEY\_LOCAL\_USERS. Will fail if *hivename* isn't a hive root.


For example:

**regrest c:\special.sav\system c:\oldsys.sav machine system**

See Also

 [RegRest ErrorLevel](#)

 [Notes on RegRest](#)

 [REGBACK.EXE](#), for backing up Registry hives.

File required for RegRest:

REGREST.EXE

### **Notes on RegRest**

- ☐ RegBack and RegRest save and reload the entire hive, including access control lists. So it's possible to restore a hive and find that you have different access control lists than before.
- ☐ RegRest does not automatically restore hives that don't reside in the CONFIG subdirectory (such as some user profiles), but RegRest will do so manually. This avoids name conflicts.
- ☐ RegRest does not restore hives that aren't loaded. You can copy those files to restore them.
- ☐ All files must be on the same volume. RegRest renames rather than copies files.
- ☐ The old hive will be stored in a .SAV file, so there must be space for this file.
- ☐ RegRest will stop at the first bug, except for manually saved hives.
- ☐ If you have a tape drive, use the Windows NT Backup program instead. Start Backup by double-clicking its icon in the Administrative Tools program group.

### **REGENCY.HLP: Windows NT Registry Entries**

RegEntry provides a database of Registry entries in a help file. You can use this Help file while working in Registry Editor to find ranges, minimum-maximum values, and instructions for setting specific values in the Registry.

#### **To use RegEntry Help:**

- 1 Double-click the Registry Help icon in the Resource Kit program group.
- 2 Select a topic to see information about Registry entries for that part of the Windows NT system.

Files required for RegEntry Help:

REGENCY.HLP  
REGENCY.IND

## REGINI.EXE: Registry Change by Script

RegIni is a character-based batch file that you can use to add keys to the Windows NT Registry by specifying a Registry script. Similar functionality is available as an interactive process in the Registry Editor, but RegIni provides a quick way to add or modify drivers in the Registry.

### To run RegIni:

- At the command prompt, type **regini** with switches and file identifiers, using the following syntax:  
**regini [-h *hivefile* *hiveroot*] [*files...*]**

For example: **regini -h myuserfile system**

If you use a .INI file to specify the Registry change, the format for the text-based file is the location of the key on the first line, followed by the Registry entry value in this format:

```
\registry\<key>
    <value name> = <data type> = <value>
```

For example, a file named SRV.INI saved on a shared directory could look like this:

```
\registry\machine\system\currentcontrolset\services\lanmanserver\parameters
    DiskSpaceThreshold = REG_DWORD 0x00000000
```

The following, typed at the command prompt, would place the **DiskSpaceThreshold** parameter in the Registry or change the value that is already there:

```
regini \\popcorn\public\scottsu\srv.ini
```

Files required for RegIni:

- REGINI.EXE
- User-defined script file



## REMOTE.EXE: Remote Command Line

The Remote utility allows you to run command-line programs on remote computers. For example, when you are developing software, you can compile code using the processor and resources of a remote computer while you perform other tasks on your computer. You can also use the Remote utility to distribute the processing requirements for a particular task across several computers.

To use Remote, you must first start the Server end (using **remote /s**) on the computer where you want to run the selected program. Then, you connect to the Server end from another computer (using **remote /c**).

**Note:** Remote cannot be used to control graphical Windows-based applications.

### To start the Server end of Remote:

- At the command prompt for the Server computer, type **remote** and the appropriate switches:  
**remote /S "Cmd" uniqueID [/U [domainname\]username] [/F] [/B]**

### To start the Client end of Remote:

- At the command prompt for the Client computer, type **remote** and the appropriate switches:  
**remote /C ComputerName uniqueID [/F] [/B] [/L]**

Where:

**"Cmd"**

Specifies the text-mode executable name for the program that you want to control from another computer, enclosed in quotes with any optional parameters required.

**UniqueID**

For the **/S** switch, specifies a string to uniquely identify this session of Remote Server from other possible sessions on this computer. For the **/C** switch, specifies the *uniqueID* of the Remote Server running on *ComputerName*

**/F param**

Specifies a foreground color for the remote command prompt such as yellow, black, and so on.

**/B param**

Specifies a background color for the remote command prompt such as lblue, white, and so on, where "lblue" means light blue.

**/L param**

Specifies the number of lines to get for the Client.

**/U [domainname\]username]**

Specifies a group or user that can connect. For example **remote /s cmd Rana /U administrators** allows only members of the Administrators group to connect, and **remote /s cmd Rana /U anniep** allows only a user logged on as "anniep" to connect. If this is not specified, anyone can connect to the session.

Remote Server example

Remote Client example

### To exit Remote on the Server computer:

- At the command prompt, type **@K**

### To exit Remote on the Client computer:

- At the command prompt, type **@Q**  
This stops Remote on the Client end, but leaves the Remote running at the Server.

File required for the Remote command line:

REMOTE.EXE

## Remote Client Example

**remote /C rajx86 brillig**

This statement connects to a server session on the server named Rajx86 with the ID "Brillig" if there is a **remote /s "Cmd" brillig** started on the computer Rajx86.

## Remote Server Example

**remote /S "i386kd -v" brillig /U administrators**

This starts the Kernel Debugger with the **-v** switch, with only members of the administrators group allowed to connect.

To interact with this "Cmd" from another computer, start the Client end with this command, where "apears" is the Remote Server name:

**remote /C apears brillig**

## REPAIR.EXE

Repair is a character-based utility that can be used to create an updated Emergency Repair Disk, reflecting changes to the system configuration that occur after the initial installation.

Repair can also be used to create a new Emergency Repair Disk.

- [Introduction to Repair](#)
- [Before You Use Repair](#)
- [Using Repair](#)
- [Repair Examples](#)

Files copied and saved by Repair:

- Files from the original Emergency Repair Disk (these are the files that contain the extra information):

- USPIFS.DLL, if present
- FATBOOT.BIN, if present
- HPFSBOOT.BIN, if present
- SECURITY.\_
- SAM.\_
- The boot sector

- File created by Windows NT Setup, in the WINNT directory:

REPAIR.INF (from SETUP.LOG)

- Files backed up by Repair, each time it runs:

SYSTEM  
SOFTWARE  
DEFAULT

Plus any changed or added files in the SYSTEM32 or DRIVERS directory

File required to use Repair:

REPAIR.EXE

## Using Repair

### To run Repair:

At a command prompt, type **repair** with the appropriate options or parameters from the following list.

#### **/bx**

Specifies the boot drive. The default is C.

#### **/ppath**

Specifies the drive and path where information is stored. The default is \REPAIR on the system drive.

#### **/nname**

Specifies the subdirectory in the repair path (**/ppath**), where information is stored. The default is the local workstation's name.

#### **/dy**

Specifies the floppy drive. The default is A.

**Important:** New Emergency Repair disks must be created on the same size drives as the original.

#### **/s**

Invokes the Save Information function.

#### **/c**

Invokes the Create Emergency Repair Disks function.

#### **/x**

Causes Extra Information to be ignored. It is highly recommended that this option be used only if the original Emergency Repair Disk isn't available.

#### **/v**

Verbose mode.

#### **/q**

Quiet mode.

### To recover the system using the new Repair Disk:

- 1 Use Windows NT Setup to recover the system, as described in the *Windows NT System Guide*.
- 2 When Setup asks for the Emergency Repair Disk, insert the disk created by Repair.
- 3 If Setup asks for any Auxiliary Repair Disk, insert the appropriate disk.

See Also

[Repair Examples](#)

[If Files Are Missing or Damaged](#)

[If Registry Files Are Too Large for One Floppy Disk](#)

[Recovering Registry Files that Are Too Large for One Floppy Disk](#)

## Introduction to Repair

The Windows NT operating system Setup Program creates a disk called the Emergency Repair Disk. If a part of the Windows NT system is deleted or corrupted, the Windows NT Setup program can use information stored on the Emergency Repair Disk to restore the system to its newly installed state.

However, changes to the system configuration made after the initial setup may be lost. For instance, when a different network card is installed or a new network protocol is added, this information is not on the Emergency Repair Disk, and will therefore not be recovered by the Windows NT Setup program.

Repair creates a new Emergency Repair Disk based on the original disk created during Setup, but does not make any changes to the original disk. So you will have two levels of restoration available: to an updated configuration or, as a last resort, to the newly installed state.

Using Repair to create a new Emergency Repair Disk is a two-part process.

1. Run Repair at the command prompt with the option to save the Emergency Repair information to disk, either locally or across a network to a server.
2. When an updated Emergency Repair Disk is needed, Repair can be run again, using the option to copy the previously saved Emergency Repair information to one or more floppy disks. If the Emergency Repair information is saved to a shared network drive, a physical Emergency Repair Disk can be created when it is needed by using a different, functioning Windows NT system, even if the target computer won't start.

Repair backs up key system configuration files and then scans the Windows NT operating system files to find any files that were updated after installation. Because Repair cannot know the source of these files -- they could be from the Windows NT installation disk, from a network drive, or from an OEM disk provided by a hardware or software vendor -- and because the Repair process requires a precisely-defined location for files to recover, the files it finds are copied to a subdirectory to be used whenever Auxiliary Repair Disk(s) need to be created.

Repair also has an option to save extra Emergency Repair Information, which is read from the original Emergency Repair disk. The information includes the boot sector, some file system drivers, and backed up Security and SAM configuration databases. After the extra information is saved, you can copy it to new Emergency Repair disks that you create. Because the extra information is constant, it only needs to be saved once for any particular computer.

The extra information consists of the following files:

- Boot sector. The original Emergency Repair disk is not bootable. If you attempt to start the system with that disk, a message tells you to run Windows NT Setup to perform repairs.
- File system drivers. If Windows NT is installed over the network (using WINNT.EXT), some file system drivers are placed on the Emergency Repair disk.
- Default Security and SAM configuration databases, installed with Windows NT. Unlike other system configuration files, it isn't possible to save these as files in a running system. (Backup uses other mechanisms.) So it isn't possible to get updated versions of these files.

You should save the extra information if you have your original Emergency Repair disk. If nothing else, it provides a copy of the original security and SAM configuration databases. If you installed Windows NT over the network, you need to save extra information to provide necessary file system drivers for the Repair process.

**Note:** When Windows NT is installed over the network, the Repair process can only report on damaged system files; it cannot actually fix them. It can still restore the original system configuration files.

**Important:** You still need to make frequent and complete backups. Although an Emergency Repair Disk can be useful in restoring a system to a bootable state, it does not contain the information you need to fully recover a Windows NT system. A full and accurate set of backups is required to ensure you can completely recover a system. In particular, if you need to restore the Security or SAM database, both the original Emergency Repair Disk and the Repair disk will restore only the default configuration defined when Windows NT was installed. This information can only be backed up by a backup utility.

Also, Repair does not back up any data. This can only be recovered from a proper backup.

Besides running Repair to save new Emergency Repair information, there should be regular procedures to perform appropriate backups.

See Also

[If Files Are Missing or Damaged](#)

[If Registry Files Are Too Large for One Floppy Disk](#)

[Recovering Registry Files that Are Too Large for One Floppy Disk](#)

## **Before You Use Repair**

Repair should be used to save new Emergency Repair information whenever a change has been made to the system configuration. This includes changes to SCSI adapters or drivers, hard disks, or network adapters or protocols. It does not include changes to users, user profiles, or installed applications.

### **To use Repair to save Emergency Repair information, you need:**

- A location to save the information to, either locally or across a network.
- The Emergency Repair Disk created by Windows NT Setup during installation, if you want to save extra information.

### **To create a new Emergency Repair Disk, you need:**

- Access to the information saved by Repair.
- One or more blank, formatted high density floppy disks for Auxiliary Repair Disk(s) and for the new Emergency Repair Disk.

## Repair Examples

### To use Repair to save emergency repair information locally:

At the command prompt, type **repair /s**

Use the **/n** option to specify another name if the computer's name is not a valid subdirectory name for the local file system. For example, if the file system is FAT, and the computer name is BILLEV\_WINNT:

**Repair /s /nSaveDir**

If the original Emergency Repair disk isn't available:

**Repair /s /x**

### To create new Emergency Repair Disks from information stored locally:

At the command prompt, type **repair /c**

If the **/n** option was used to specify a subdirectory when the information was saved, use the **/n** option now to specify that subdirectory:

**Repair /c /nSaveDir**

If the extra information wasn't saved, the command is the following:

**Repair /c**

**Note:** The **/x** parameter was omitted. Repair won't copy the extra information to the floppy if it was never saved in the first place.

### To save Emergency Repair information to a server:

1 At the command prompt, type **net use Q: \\MyServer\ERInfo** and press ENTER.

2 Type **repair /s /pq:\**

If the computer's name is not a valid subdirectory name for the server's file system, or if another name is assigned by the LAN administrator, use the **/n** option to specify another name. For example:

**net use Q: \\MyServer\ERInfo**

**Repair /s /nDirToUse /pQ:\**

### To create new Emergency Repair Disks from information on a server:

1 At the command prompt, type **net use Q: \\MyServer\ERInfo** and press ENTER.

2 Type **repair /c /pq:\**

If the **/n** option was used to specify a subdirectory when the information was saved, use the **/n** option now to specify that subdirectory. For example:

**net use Q: \\MyServer\ERInfo**

**Repair /c /nDirToUse /pQ:\**

### To create new Emergency Repair Disks for another computer:

1 At the command prompt, type **net use Q: \\MyServer\ERInfo** and press ENTER.

2 Type **repair /c /nDirToUse /pQ:\**

In this case, the computer is always another computer.



## **If Files Are Missing or Damaged**

When Repair runs, it scans the files listed on the original Emergency Repair Disk's list of files. If a file from the original installation is missing, a message is displayed, to let the operator know about the situation. The Emergency Repair information is retained from the original Emergency Repair Disk.

If a file has changed (as detected by a checksum check), and it is a Win32 executable (that is, .EXE, .DLL, and so on), it will be checked for internal consistency. Any file that fails this test is corrupt, and the operator will be alerted. The Emergency Repair information is retained from the original Emergency Repair Disk.

## **If Registry Files Are Too Large for One Floppy Disk**

The original Emergency Repair Disk created by Windows NT Setup consists of only one floppy disk. In most cases, the new Emergency Repair Disk created by Repair will also be only one floppy. However, because Registry files can grow without limit, the files that fit onto the single original disk may no longer fit on one disk.

If this happens, Repair will copy one or more of the Registry files to separate floppies, which will be called Auxiliary Hive 1, Auxiliary Hive 2, and so on. If this happens, it is extremely important to note which Registry files are copied onto which floppies (this is displayed on the screen as Repair runs.) This information is required if the Registry files are ever to be restored.

The Registry files are always copied in the same order. If one will not fit into the space remaining on the floppy, Repair asks for a new floppy, and the copies continue. A Registry file is never split across two floppies (which means that if a file is bigger than a floppy, it can't be backed up.) The Registry files are copied in the following order:

- 1 SECURITY
- 2 SAM
- 3 DEFAULT
- 4 SYSTEM
- 5 SOFTWARE

SECURITY and SAM can't overflow, because they are from the original Emergency Repair Disk, where they already fit.

■ Recovering Registry Files that Are Too Large for One Floppy Disk

## Recovering Registry Files that Are Too Large for One Floppy Disk

To recover Registry files when one or more overflowed onto an Auxiliary Hive Disk, you must know which Registry files are on which floppies.

- ☒ The SECURITY and SAM hives will always be on the updated Emergency Repair Disk.
- ☒ The DEFAULT, SYSTEM, and SOFTWARE hives may be on one or more Auxiliary Hive disks.

When Setup is used to repair a system, three operations can be selected:

- ☒ Verify Windows NT system files
- ☒ Verify boot files on your C: drive
- ☒ Inspect Registry files

When you select the Inspect Registry Files option, you can select five configuration files:

- ☐ SYSTEM (System Configuration)
- ☐ SECURITY (Security Policy)
- ☐ SAM (User Accounts Database)
- ☐ DEFAULT (Default User Profile)
- ☐ SOFTWARE (Software Information)

Because the original Emergency Repair Disk has these files on a single floppy, Setup expects a single file for the repair process. So, to recover a Registry file from an Auxiliary Hive disk, that disk must be placed into the disk drive before the verify-and-repair operation is started. Also, only Registry files stored together on a single disk can be verified and repaired in a single operation. (Of course, you can run the repair process several times.)

For example, if the SOFTWARE and SYSTEM Registry files overflowed onto the Auxiliary Hive 1 disk, the DEFAULT USER, SECURITY and SAM Registry files could be verified and repaired in one operation, and the SOFTWARE and SYSTEM Registry files could be verified and repaired in another. To repair both the default USER and the SYSTEM Registry files requires two separate repair operations.

## SCOPY.EXE: File Copy with Security

The Security Copy tool can be used to copy files and directories from NTFS partitions with their security intact. You can use this utility to back up files and directories from NTFS partitions without using a tape drive.

**Note:** If you are copying from directories that are on FAT or HPFS partitions, use **xcopy** instead.

### To use Security Copy:

At the command prompt, type **scopy** with the appropriate switches:

**scopy** *source* [*destination*] [/o] [/a] [s]

Where:

*source*

Specifies the source for files to be copied.

*destination*

Specifies where to copy files to.

/o

Copies owner security information.

/a

Copies security auditing information.

s

Copies all files in subdirectories.

You can type **scopy /?** at the command prompt to get help.

**Note:** To copy your own files, you do not require any special user privileges. But to use the /o or /a switches, or to copy other users' files that you don't ordinarily have access to, you must be logged on as a member of the Administrators group on both the computer where you are copying the files from and on the computer you are copying the files to.

[More about SCopy and User Privileges](#)

File required for Security Copy:

SCOPY.EXE

## SCopy and User Privileges

SCopy tries to use the following user privileges in these cases:

- "Backup files and directories" privilege allows you to copy files when ordinarily your access is restricted at the source.
- "Restore files and directories" privilege is needed to use the **/o** switch to copy files that aren't your own.
- "Manage auditing and the security log" privilege is needed to use the **/a** switch.

## SLEEP.EXE: Batch File Wait

The Sleep utility waits for a specified amount of time. Sleep is useful in batch files and may be more convenient to use than the **at** command in certain cases.

### To use Sleep:

- Add a line in a batch file in the format **sleep time**  
Where *time* is the number of seconds to pause.

For example, **sleep 3600** will pause for an hour before running the next command in a batch file; **sleep 10** waits 10 seconds.

### Sleep Batch File Example

File required for Sleep:

SLEEP.EXE

## Sleep Batch File Example

This example could be used in a logon script:

```
@echo off
echo * Message of the Day - 7/22/93 *
echo.
echo Don't forget the company picnic!
echo Buses leave at 4pm.
sleep 60
```

## SMBTRACE.EXE: Server Message Block Tracer

SmbTrace is a character-based network diagnostic tool that traces Server Message Blocks (SMBs) sent and received by the server and redirector. It can be used to diagnose some network traffic problems. SmbTrace captures SMB requests from the server (or redirector) and dumps them on the screen.

You must be logged on as a member of the Administrators group to use SmbTrace.

### To run SmbTrace:

At the command prompt, type **smbtrace** with the appropriate switches:

**smbtrace [/rdr/srv] [/slow/fast] [/ver:N] [/data:N] [/buf:N] [/max:N]**

Where:

#### **/rdr**

Capture SMBs from the redirector.

#### **/srv**

Capture SMBs from the server (the default).

#### **/slow**

Capture all SMBs, slowing down the redirector or server if necessary to prevent loss of SMBs. Care should be taken while using this mode. Press CTRL+S in this mode to stop processing in the redirector or server.

#### **/fast**

Capture all SMBs without slowing down the redirector or server. Some SMBs may be lost in this mode if the capture buffer fills. SmbTrace will indicate how many SMBs were lost if this occurs.

#### **/verbosity:N**

Set verbosity level to *N*, where the range is 1 - 5 and the default is 2 (low).

#### **/data:N**

Dump *N* bytes of raw data of each SMB. The default is 0.

#### **/maxSMB:N**

Allow a maximum size of *N*. The default is 4096.

#### **/bufferize:N**

Buffer up to *N* kilobytes. The default is 1 MB.

#### **/number:N**

Buffer up to *N* different SMBs. The default is 128.

### To turn off SmbTrace:



Type **smbtrace /stop** at the command prompt.  
Or press CTRL+C in the window where SmbTrace is running.

File required for SmbTrace:

SMBTRACE.EXE

## SNMPUTIL.EXE: SNMP Browser

The SNMP browser is a utility that lets you get SNMP information from an SNMP host on your network.

### To use the SNMP utility:

- At the command prompt, type **snmputil** with the appropriate switches:

**snmputil [get|getnext|walk] agent community oid [oid]**

or

**snmputil trap**

Where:

#### **get**

Used to get the current value of the specified *oid*(s).

#### **getnext**

Used to get the current value of the item in the MIB that follows the item whose *oid* is specified.

#### **walk**

Used to step through the MIB and retrieve the values of all items in the branch of the MIB specified by *oid*.

#### *agent*

Specifies the computer to query. This can be either an IP address or a hostname if the computer is specified in the hosts file.

#### *community*

Specifies a community name, which is used to group computers together into management groups. You can see a list of community names in the SNMP Service Configuration dialog box if you choose the Network icon in Control Panel. Or see the value of **ValidCommunities** in the SNMP\Parameters subkey in the Registry. By default, a community called "public" is created when you install SNMP under Windows NT.

#### *oid*

The ASN.1 name of the variable being queried, of the form .N.N.N.N (that is, a string of numbers separated by periods or, alternately, a string of names separated by periods). *oid* is an abbreviation for "Object Identifier."

#### **trap**

Tells SNMPUTIL to listen for trap PDUs.

For example:

**snmputil get jb486 public .iso.org.dod.internet.mgmt.mib-2.system.sysDescr.0**

This would return a text description of the type of computer hardware and software being used on the computer named jb486.

Notice that the *oid* strings can get quite long. There is one built-in short cut. For items in the .iso.org.dod.internet.mgmt.mib-2 branch of the Internet MIB, the request can be shortened to:

**snmputil get jballard486 public system.sysDescr.0**

In this case, the *oid* does not start with a period (".").

**Note:** The SNMP service does not need to be running on the computer that is executing **snmputil**, but the proper files must be copied to that computer. In particular, MGMTAPI.DLL and MIB.BIN must be present, and TCP/IP must be installed on the computer.



File required for the SNMP utility:

SNMPUTIL.EXE

(SNMP service must be installed)



Server  
Manager

## **SRVMGR.EXE: Server Manager**

Server Manager is a tool you can use to manage domains and computers. With Server Manager you can:

- Select a domain, workgroup, or computer that you want to administer.
- Manage a computer. For a selected computer you can view a list of connected users, view shared and open resources, manage directory replication, manage the list of administrative alert recipients, manage services and shared directories, and send messages to connected users.
- Manage a domain. When administering a domain you can promote a server to become the domain controller, synchronize servers with the domain controller, and add computers to and remove computers from the domain.

Some of the capabilities offered by Server Manager are also offered by the Services and Server options in the Control Panel of every Windows NT computer. However, Server Manager can manage both local and remote computers, while these Control Panel options only affect the local computer.

### **To use Server Manager:**

- 1 Double-click the Server Manager icon in the Resource Kit program group.
- 2 To get Help while using Server Manager, press F1.

In most cases, when Server Manager is first started it displays your logon domain. The Server Manager title bar shows the domain name, and the body of the Server Manager window lists the computers of that domain. A computer can be selected from this list, and can then be managed using commands from the Computer menu.

**Note:** If you are already running Windows NT Advanced Server, this version is the same as the Server Manager application in the Administrative Tools group in Program Manager.

Files required for Server Manager:

SRVMGR.EXE  
SRVMGR.HLP



Topdesk

## **TOPDESK.EXE: Multiple Desktops**

TopDesk provides a powerful way to switch between and organize applications while conserving screen space.

You can use the keyboard and mouse to work with TopDesk. In general, use the left mouse button to move objects, and use the right mouse button to display popup menus to choose actions.

### **To use TopDesk:**

- 1 Double-click the TopDesk icon in the Resource Kit program group.
- 2 Press F1 for online Help.

Files required for TopDesk:

TOPDESK.EXE  
TOPDESK.HLP  
TOPHOOK.DLL

## **TROUBLE.HLP: Troubleshooting Flowcharts**

Troubleshooting Help leads you through a series of questions to aid you in solving problems with hardware and software setup in Windows NT.

### **To use Troubleshooting Help:**

- 1 Double-click the Troubleshooting Flowcharts icon in the Resource Kit program group.
- 2 Select a topic to begin troubleshooting for that part of the Windows NT system.

File required for Troubleshooting Help:

TROUBLE.HLP  
TROUBLE.IND



Uconvert

## **UCONVERT.EXE: Unicode Converter**

The Unicode Converter can be used to convert a file to Unicode from an ANSI, OEM, or other type of code page. You can also use the commands on the Conversion menu to specify conversion for or to other types of code pages, to specify various conversion options, and other options.

### **To use the Unicode Converter:**

- 1 Double-click the Unicode Converter icon in the Resource Kit program group.
- 2 From the File menu, choose Open Source File, and then specify the file you want to convert.
- 3 From the Conversion menu, choose Convert now.
- 4 In the Save As dialog box, specify a filename for the converted file.

Uconvert also allows you to install additional conversion tables without using REGEDT32.EXE. To do this, choose the Install New Conversion Tables command from the Conversion menu, and choose the Add button in the Conversion Tables dialog box so that you can select additional .NLS files to add to the conversion list. The .NLS files provided with Windows NT are stored in the *SystemRoot*\SYSTEM32 directory.

File required for Unicode Converter:

UCONVERT.EXE



## UPEDIT.EXE: User Profile Editor

User Profile Editor is a Windows NT administrative tool for creating customized user profiles for these cases:

- To define what's displayed when the logon prompt appears on a Windows NT computer.
- To set up profiles for users who haven't logged onto the computer yet.

### To use User Profile Editor:

- 1 Double-click the User Profile Editor icon in the Resource Kit program group.
- 2 To get Help while using User Profile Editor, press F1.

**Note:** If you are already running Windows NT Advanced Server, this version is the same as the User Profile Editor application in the Administrative Tools group in Program Manager.

Files required for User Profile Editor:

UPEDIT.EXE  
UPEDIT.HLP

### To define the screen display for the logon prompt

- 1 Log on as a member of the Administrators group. Configure the colors, screen saver, and wallpaper that you want. This can include custom wallpaper such as the company's logo, and can also include custom text for the logon prompt. (See WinLogon in REGENTRY.HLP.)
- 2 Choose the Save As System Default command from the File menu in User Profile Editor.

### To set up profiles new users

- 1 Log on as a member of the Administrators group. Configure all the options that you want to define for the profile.
- 2 Choose the Save As User Default command from the File menu in User Profile Editor.

### User Profile

A user profile is a definition of the Windows NT configuration for a specific user. By default, each Windows NT workstation maintains a profile for each user who has logged on to the workstation. Each user profile contains information about a particular user's Windows NT configuration. Much of this information is about things the user can set, such as color schemes, screen savers, and mouse and keyboard layout. Other information is about things that can be set only by a Windows NT administrator, such as access to common program groups or network printers.



### USRMGR.EXE: User Manager for Domains

User Manager for Domains is a tool you can use for remote management of security for domains, servers, and workstations. With User Manager for Domains you can:

- Select the domain or computer to be administered.
- Create and manage user accounts.
- Create and manage groups.
- Manage the security policies.

### To use User Manager for Domains:

- 1 Double-click the User Manager icon in the Resource Kit program group.
- 2 To get Help while using User Manager for Domains, press F1.

In most cases, when User Manager for Domains first starts, it displays your logon domain. The title bar shows the domain name, and the body of the User Manager for Domains window displays two lists. The upper list contains user accounts; the lower list contains groups. One or more user accounts, or one group, can be selected and then managed using commands from the User menu.

**Note:** If you are already running Windows NT Advanced Server, this version is the same as the User Manager for Domains application in the Administrative Tools group in Program Manager.

Files required for User Manager for Domains:

USRMGR.EXE

USRMGR.HLP





## WINAT.EXE: Command Scheduler

The Windows NT Command Scheduler can be used to schedule commands on a local or remote computer to occur once or regularly in the future. The workstation service must be started to use this application.

The current AT commands for the local computer are displayed by default when Command Scheduler is started. The list of AT commands is automatically refreshed, so the display is always current. AT commands are displayed in a format similar to that of command-line AT command.

**Note:** See Windows NT Help for a full description of definitions for the AT command.

### To use the Windows NT Command Scheduler:


- 1 Double-click the Scheduler icon in the Resource Kit program group.

Or at the command prompt, type **winat** *\\computername*

- 2 Press F1 for help while you are working with Command Scheduler.

**Important:** If you want the command you are executing to work across the network, then the account the Schedule service is using must be able to work across the network. The System Account cannot access the network. Also, if you want a command to access NTFS protected files, the account the Schedule service uses must have access to those files.

### To view the commands at a remote computer:

 From the File menu, choose the Select Computer command, and complete the dialog box in the usual way.

### To add a new command:

- 1 Choose the Add button to display the Add Command dialog box.
- 2 Type the command name in the Command box. Commands are limited to 128 characters.
- 3 Choose the Today, Tomorrow, Every, or Next button.
- 4 From the Days and Time list boxes, select the days and time when the command is to run, and then choose OK.

### To change a command:

- 1 Choose the Change button to display the Change Command dialog box.
- 2 Modify the command in the same way as you add a new command.

### To remove a command:

 Select the command you want to remove, and choose the Remove button.



WinDiff

## **WINDIFF.EXE: File and Directory Comparison**

WinDiff shows the differences between specified files or directories of ASCII text files. This is particularly useful for program source code.

The display either shows a summary of the comparison status of a list of files (outline mode) or a detailed line-by-line comparison of one of the files (expanded mode).

You can run WinDiff like any other Windows-based application and use the menu commands to choose files to compare and perform other actions. Or you can run WinDiff from the command prompt with switches.

 Running WinDiff with Command-Line Switches

 Running WinDiff with Menu Choices

Files required for WinDiff:

WINDIFF.EXE

GUTILS.DLL

## Running WinDiff with Command-Line Switches

To run WinDiff from the command-line:

- At the command prompt, type **windiff** with the appropriate switches:  
**windiff** [*paths*] [*saveoption*]

Where:

*paths*

Specifies pathnames for the comparison as a relative or absolute path, network or local path, or file or directory, according to the following:

*path* to compare what's at *path* with what's in the current directory

*path1 path2* compares what's at *path1* with what's at *path2*

*saveoption*

**-s** *slrd* *savefile*

where *slrd* is any combination of these four letters to write the names of files that are:

- s** the same in both paths
- l** only in the left hand path
- r** only in the right hand path
- d** in both paths, but the two files are different

Colours

RED background = LEFT FILE


YELLOW background = RIGHT FILE

blue text = moved line

black text = everything else

## Running WinDiff with Menu Choices

### To run WinDiff using the menu:




-  Double-click the WinDiff icon in the Resource Kit program group.

### To compare two files with WinDiff:

- 1 From the File menu, choose Compare Files.
- 2 In the Select First File dialog box, specify a filename for the first file in the comparison. Then select the other file to compare in the Select Second File dialog box.

### To compare directories with WinDiff:

- 1 From the File menu, choose Compare Directories.
- 2 In the Select Directories dialog box, specify two directory names and, if desired, check the Include Subdirectories check box.

-  Menu Commands
-  Outline Mode View
-  Expanded Mode View

## **WinDiff Menu Commands**

File Menu

Edit Menu

View Menu

Expand Menu

Options Menu

## **WinDiff File Menu**

### **Compare Files...**

Displays a dialog box that allows you to select files to compare. It displays the standard Open File dialog box twice, once for the each file.

### **Compare directories**

Displays a dialog box that allows you to select two directories to compare. UNC names are allowed. The names can be absolute paths or relative to the current directory. The Include Subdirectories check box determines whether the comparison will include the trees, starting from the given paths, or only the directories named.

### **Abort**

Dimmed unless an operation is in progress, then allows that operation to be terminated before completion.

### **Save File List**

Allows you to save the outline mode data in a file. During the file comparison, a checksum is calculated for each file (which is useful if the network is not working). These can be recorded together with the file names, if required.

### **Copy Files...**

Displays a dialog box for writing the listed files to a disk. WinDiff can copy any combination of the four categories (identical, and so on). If greater selectivity is required, then save the file list (without checksums), and edit the file into a .BAT file to do the copying.

For a remote compare, this will compress the files before sending them across the network.

### **Print**

Prints the file list or the expanded comparison.

## WinDiff Edit Menu

**Edit Left File**

**Edit Right File**

**Edit Composite File**

These commands allow a text editor to be invoked from WinDiff. You can edit either the left file, the right file or the merged file.

### **Set Editor**

Allows you to select which editor is invoked. If you use the Slick editor, you should set this to:

**s %p -#%l**

**%p** represents the path to edit (that is, the filename, which may be a name created by WinDiff) **.%l** represents the line number. The default is **notepad %p**.

## **Checksums in WinDiff**

For each file compared a checksum is calculated. This is a 4-byte number derived from some combination of the values of all the bytes in the file in such a way that it is extremely unlikely that two files that are different will have the same checksum, while ensuring that files that are identical always have the same sum.

(The odds of a two different files giving the same sum is about 1 in 4 billion, which is to say that you could compare 1 file a second for the next 100 years and only have about a 50/50 chance of seeing it happen).

## **WinDiff Expand Menu**

### **Left File Only**

Shows only lines from left file (but colored to highlight changed lines).

### **Right File Only**

Shows only lines from right file (but colored to highlight changed lines).

### **Both Files** (default)

Shows a merge of both files. All the lines in the left file are shown in the order in which they occur in that file; likewise for the right file. Lines that are only in the left file are shown in red. Lines that are only in the right file are shown in yellow.

### **Left Line Numbers** (default)

Line numbers are shown, based on the left file.

### **Right Line Numbers**

Line numbers are shown, based on the right file.

### **No Line Numbers**

Line numbers are turned off.

## **WinDiff View Menu**

### **Outline**

Sets outline mode, showing lists of files.

### **Expand**

Sets expanded mode, showing a comparison of selected files.

### **Picture**

Shows picture as well as (turning the picture off saves space on the screen).

### **Previous Change**

Skips to previous point of difference in the file.

### **Next change**

Skips to next point of difference in the file.



## WinDiff Options Menu

### Ignore Blanks

Blanks are ignored in the expanded mode, so that lines that differ only in white space are shown as identical.

The next four options control which files (if any) are displayed in outline mode:

### Show Identical Files

Includes files that are identical in each path.

### Show Left-Only Files

Includes files that occur only in the left hand path.

### Show Right-Only Files

Includes files that occur only in the right hand path.

### Show Different Files

Includes files that occur in both paths, but that are not the same.

## WinDiff Outline Mode Display

In outline mode, the display shows one file per line with the files listed in alphabetical order--except files contained in subdirectories appear after all files in higher directories. The paths shown are all relative to the root path given when the comparison is invoked.

For example, this command:

**windiff c:\temp d:\progs\new**

might show:

.\myfile.txt	identical
.\hisfile.txt	files differ
.\newest\myfile.txt	only in C:\TEMP
.\newest\x.x	only in D:\PROGS\NEW
.\oldest\file	identical

In this case, the full paths would be:

.\myfile.txt	c:\temp\myfile.txt and d:\progs\new\myfile.txt
.\hisfile.txt	c:\temp\hisfile.txt and d:\progs\new\hisfile.txt
.\newest\myfile.txt	c:\temp\newest\myfile.txt
.\newest\x.x	d:\progs\new\newest\x.x
.\oldest\file	c:\temp\oldest\file and d:\progs\new\oldest\file

The status of a file can be one of the following:

### Identical

The files are identical, including white space (see below).

### Files Differ or Different Sizes

The two files differ, but possibly only in white space (see below).

### Only In *path*

The file only exists in one directory or the other same size and so on.

### Files Differ? (Left Unreadable)

The file in the left path existed, but could not be read. There are many possible reasons. Some of the common ones are Network Error and Access Denied.

### Files Differ? (Right Unreadable)

The file in the right path existed, but could not be read. The file in the left path was successfully read.

Unreadable files are treated as being different (highlighted in red, and so on).

The Options menu allows for files in any of the four categories to be either displayed or suppressed (that is, any of the 16 combinations). The categories are:

- ☐ Identical
- ☐ Differ (including unreadable)
- ☐ Left only
- ☐ Right only

**Note:** A common source of puzzlement is a blank display caused by having only files in categories that are turned off from the Options menu. For example, all the files are identical, but identical files are not being displayed.

## WinDiff Expanded Mode Display

The left of the screen shows a pictorial view of the comparison. Blocks of colour represent sections of the files. The left column of blocks represents the left hand file, the right hand column represents the right hand file.

Where blocks of lines match in the two files, these are shown as white blocks joined by a line connecting the block in each file. Lines that do not match are represented by a red block in the left file or a yellow block in the right file (with no connecting line).

The picture can be turned off by using the View menu.

Lines of text are displayed using the same colours. That is, lines that occur only in the left file are shown with a red background; lines that occur only in the right file are shown with a yellow background. Lines that are moved are shown with blue text against the appropriate background.

The Expand menu allows you to choose whether to look at the left file, the right file, or (the default) a merge of the two files. This merged file contains all the lines that are in either file in the order in which they occur in that file.

Lines that are common to the two files are normally shown only once. (Moved lines are shown twice: once with a red background showing where the line was in the left file ,and once with a yellow background showing where the line is in the right file.)

Lines are numbered. The numbering can be turned off to save screen space or can be shown based on the left file or shown based on the right file by using the Expand menu.



## WinVTP **WINVTP.EXE: Windows NT Communications**

WinVTP is a Windows NT communications tool that allows you to access the Net2Com service from a computer running Microsoft Windows NT in lieu of using a terminal program. WinVTP creates a connection via NetBIOS to communicate with the remote Net2Com service and emulates the common functions of an ANSI/VT100 terminal.

### **To create a connection from the command prompt:**

- At the command prompt, type **winvtp** plus the service name. For example, type **winvtp LIONHEART** to use the Net2Com service.

WinVTP.exe will automatically try to connect you to the Net2Com service (**LIONHEART** is the default NetBIOS connection name for the Net2Com service).

**Note:** The service name that you type has to be the same case as the Net2Com service was setup with.

### **To create a connection after WinVTP is running:**

- Choose Connect... from the File menu. Then enter the service name (**LIONHEART** if you are using the Net2Com service), and choose the Connect button to try connecting to the service.

If you succeed in connecting to the service, the WinVTP title bar shows the connected service name. Once you're connected, you're talking directly to the modem.

### **To dial a number on a Hayes-compatible modem:**

- Type **ATDT** <phone number> and press ENTER.

The "ATDT" string can be either upper or lower case. The <phone number> can consist of the digits 0...9, '-' (hyphen), and ',' and spaces. Use a comma to insert a pause. For example, if you need to enter a 9 before the phone number to reach outside numbers, you should enter a comma after the 9.

### **To hang up a modem connection on a Hayes-compatible modem:**

- Quickly enter **+++**

### **To finish using the service:**

- Either choose Hangup from the File menu if you want to connect to other services, or choose Exit from the File menu to quit WinVTP.

For more information about communicating with the modem, see your modem manufacturer's manual.

For more information about using WinVTP commands:

- WinVTP File Menu
- WinVTP Options Menu

File required for WinVTP:

WINVTP.EXE

## **WinVTP File Menu**

### **Connect**

Displays a dialog box so you can enter the service you want to connect to. To connect to the Net2Com service provided on the Resource Guide disks, enter **LIONHEART**.

WinVTP keeps track of the four most recently connected services. These service names appear at the bottom of the File menu.

### **Hangup**

Ends the connection so you can connect to other services.

### **Exit**

Quits WinVTP. If you select Exit while a connection to a service is still active, WinVTP will disconnect you from the service automatically.

## **WinVTP Options Menu**

### **25 Lines, 43 Lines, 50 Lines, and Custom Lines**

Use these commands to specify the size of the WinVTP display window. The Custom Lines command displays a dialog box for specifying the display to be 16 - 99 lines high.

### **Fonts**

Displays a Font selection dialog box. WinVTP can only use fixed-width fonts, so it asks the system to display only fixed-width fonts in the dialog box. Some fonts that the system believes to be fixed-width aren't really, such as the Courier New TrueType font. If you select Courier New or another fixed-width TrueType font, you may see display problems.

### **Local Echo**

Specifies whether local echoing is on or off. Local Echo will display all your keyboard input since the modem may not "echo" your input. After you connect to the service and before you connect to another computer via the modem, you'll probably want to turn on local echoing. A check mark appears beside the command name when local echoing is on.

### **No Connection Lost dialog**

Specifies that WinVTP won't display a message if the connection is lost. If the connection is broken, the WinVTP title bar will be updated to show whether you're connected to a service, regardless of whether or not you selected the No Connection Lost Dialog command.

### **No Connect Retry dialog**

If WinVTP cannot establish a connection to a service on the first try, it will display the Connect Auto Retry dialog box, and will attempt to connect to the service every 5 seconds until it succeeds or the user chooses the Abort button or closes the dialog box. If you don't want this behavior, you can hold down the SHIFT key after you select a service to connect to.

You can also select the No Connect Retry Dialog command from the Options menu.

### **Text Colour**

Allows you to specify the text color for WinVTP's display.

### **Background Colour**

Allows you to specify the background color for WinVTP's display.



## MacFile **SFMATALK.SYS: AppleTalk Protocol and Services for Macintosh Administrator Tools**

You can install the AppleTalk Protocol and Services for Macintosh (SFM) Administrator Tools included on the Resource Guide disks. Microsoft Windows NT Services for Macintosh® is a thoroughly integrated component of Windows NT, making it possible for PC and Apple® Macintosh workstations to share files and printers.


**Note:** For Windows NT Advanced Server, you can choose to install Services for Macintosh in the Network Settings dialog box, and then install the service from your Windows NT Advanced Server product disks.


Installing AppleTalk Protocol or SFM Administrator Tools is a separate task after you install the Resource Kit tools.

### **To install AppleTalk Protocol or SFM Administrator Tools on a Windows NT computer:**

- 1 Choose the Network icon in Control Panel.
- 2 In the Network Settings dialog box, choose the Add Software button.
- 3 In the Add Network Software dialog box, select AppleTalk Protocol or Services for Macintosh Administrator Tools from the list, and then choose the Continue button.
- 4 Follow the directions on screen. In the message box that asks you specify the disks for installation, you must specify the drive and complete path name for the Resource Kit utilities. That can be A: or B: for floppy disks. For CD-ROM, type *driveletter:\i386* or *driveletter:\MIPS*, depending on your type of computer. For a network resource, type a drive letter and path (such as z:\RESKIT) for an existing connection, or the complete UNC pathname (such as \\SHARE\UTILITY\RESKIT).

After you install SFM Administrator Tools, a MacFile menu appears in the File Manager and Server Manager menu bars.

 Stopping and Removing SFM

 SFM System Requirements

Files required for Services for Macintosh Administrator Tools:

Installation information:

OEMNSVKT.INF

AppleTalk Utilities:

SFMATCFG.DLL

SFMATMSG.DLL

SFMWSHAT.DLL

AppleTalk driver:

SFMATALK.SYS

SFM Utility:

SFMUTIL.DLL

SFM Administrator tools:

SFMAPI.DLL

SFMMGR.HLP

SFMMGR.CPL

## **Stopping and Removing SFM Administrator Tools**

After you have set up SFM Administrator Tools on a server, you can remove it at any time. For example, you might want to move your SFM Administrator Tools installation to another server. When you remove SFM Administrator Tools, all SFM Administrator Tools files are deleted from the servers hard disk, except some program files that are in use. (The files in use will be reworked when the system is rebooted.)

Before you remove SFM Administrator Tools, it is recommended that you stop the services using the Devices control in the Control Panel.

If the services are running, removing will delete only the registry entries.

### **To stop the services:**

- 1 Choose the Devices icon in Control Panel.
- 2 From the Device list in the Devices dialog box, select AppleTalk Protocol, and then choose the Stop button.
- 3 In the Stopping dialog box, choose the OK button to stop the AppleTalk Protocol and the Print and File Servers for Macintosh.

### **To remove AppleTalk Protocol or SFM Administrator Tools:**

- 1 Choose the Network icon in the Control Panel.
- 2 In the Installed Network Software box of the Network Settings dialog box, select AppleTalk Protocol or SFM Administrator Tools (whichever one you want to remove).
- 3 Choose the Remove button.

The AppleTalk Protocol or SFM Administrator Tools will be deleted as specified. This will also make all volumes unavailable to the Macintosh users, but it will not delete them; in other words, the volumes will revert to directories. If you remove SFM Administrator Tools and later decide to set it up again, you must use the Resource Guide disk and installation program to copy the required SFM Administrator Tools files to the server. Removing SFM Administrator Tools deletes the distribution files (except Macintosh-accessible volumes) instead of disabling them.



## **SFM Administrator Tools System Requirements**

To set up SFM Administrator Tools, you need a PC running Windows NT; to make full use of it, you need Macintosh workstations. Requirements for each of these follow.

Also, SFM Administrator Tools supports version 5.2 (or later) of the LaserWriter printer driver, and AppleTalk File Protocol v. 2.0 and 2.1.

### **Requirements for the Windows NT Computer:**

There are no additional requirements for the Windows NT computer where you install SFM Administrator Tools. However, the Windows NT server that you use must have an NTFS partition for directories (Macintosh-accessible volumes) that can be used by Macintosh workstations. The local computer where you are using SFM Administrator Tools does not require an NTFS volume, because you can focus on another computer (the SFM server) and administer that computer.

### **Requirements for Macintosh Workstations:**

All Macintosh computers that can use AppleShare® (the Apple networking software for the Macintosh) can use Services for Macintosh. These include all Macintoshes except the Macintosh XL and Macintosh 128K. To use Services for Macintosh, the Macintosh must have version 6.0.7 or later (including System 7™ or higher) of the Macintosh operating system.

SFM Administrator Tools supports LocalTalk®, Ethernet, Token Ring, and FDDI (Fiber Distribution Data Interface). Ethernet and token ring are commonly used when integrating Macintoshes into PC networks.

## WINNTP.EXE: Computer Profile Setup

Computer Profile Setup allows you to easily install either Windows NT or Windows NT Advanced Server on multiple x86-based computers.

**Note:** Computer Profile Setup works only for English-language versions of Windows NT.

The utility programs that make up Computer Profile Setup are used to make a copy (a "Computer Profile") of an installed Windows NT system. Then, for any number of computers that have an identical configuration, this Computer Profile is loaded and reinstalled using Windows NT Profile Setup.

### To use Computer Profile Setup:

- Follow the steps in the following topics:
- [Getting Started with Computer Profile Setup](#)
- [Using Computer Profile Setup](#)
- [Limitations for Computer Profile Setup](#)

See Also

Chapter 3, "Customizing Windows NT Setup," in the *Windows NT Resource Guide*, for details about Computer Profile Setup.

Files required for Computer Profile Setup:

- ~PROFILE.INF
- FATBOOT.BIN
- GETBS.EXE (optional)
- LOADACCT.EXE
- PROFILE.INI
- RESTFILE.EXE
- RESTKEYS.EXE
- UPLODPRF.EXE
- UPLODPS2.BAT (optional)
- WINNTP.EXE

## Getting Started with Computer Profile Setup

The following describes the prerequisites for Computer Profile Setup.

### Source (Model) Computer Requirements:

The Source computer is the prototype where the system is installed and configured that will serve as the Computer Profile. Before running any CPS utilities on this computer, you must first use Windows NT Setup to install Windows NT or Windows NT Advanced Server. Then load any additional software you want to propagate with this Computer Profile.

- Only Common program groups are propagated with the Computer Profile, so be sure that any installed software uses Common program groups. Otherwise, these program groups will not appear on the Target computers.

- Local user accounts are not part of the profile; however, local group accounts and domain and local group permissions are part of the profile.

After the Source computer has been configured as you want it, then load the CPS utilities onto the Source computer.

**Note:** All software to be propagated must reside on the same volume (logical disk) as the Windows NT system directory, which must also be the boot drive (typically C:\WINNT).

### Target Computer(s) Requirements:

The Target computer must have access to the Computer Profile directory (either by way of the network or a removable drive) and must have MS-DOS 5.0 or 6.0 installed.

The Target computers should be identical to the hardware configuration on the Source computer, but some exceptions may not cause problems:

- The disk on the Target computer may be larger, but cannot require a different driver, because the required driver may not be installed or configured properly.
- Memory (RAM) may be larger on the Target computer.

Components that are certain to cause problems if different from the Source computer:

- Different network card on the Target computer. Hardware Autodetect is not run during Profile Setup and the necessary driver file will not be available.
- Smaller disk or RAM on the Target computer.
- Any hardware (such as a video monitor) that requires a different driver or configuration from the Source computer.

See Also

- [Input \(.INI\) File for Computer Profile Setup](#)
- [Computer Profile Setup Processing Overview](#)
- [Security Dump File Formats for Computer Profile Setup](#)

## Using Computer Profile Setup

Setting up multiple computers with Computer Profile Setup requires three activities:

- Setting Up the Source Computer
- Running UPLDPRE (Upload Profile)
- Setting Up the Target Computer

## Setting Up the Source Computer

### To set up the source computer:

- 1 Configure the Source computer the way you want it, including all the directories and files on the Windows NT system drive that are to be part of the Computer Profile.  
Windows NT must be installed on the boot drive. Other applications and files to be propagated must be on the same volume as Windows NT.
- 2 Create a new directory or clean an existing directory where the Computer Profile is to be stored, either on a network share point or a removable disk
- 3 Decide which file system to install on the Source computer, either keeping the existing FAT file system or converting the volume to NTFS after files are installed.  
It's easiest to install and set up the profile on the Target computer on an NTFS volume (for example, using Autoconvert during Setup). Then you can go to either FAT or NTFS using Computer Profile Setup.  
Going from FAT on the Source computer to NTFS on the Target computer may not provide the correct file protection (because the defaults will be used). You need to set permissions on the Source computer for these to be propagated; therefore, use NTFS on the Source computer.
- 4 Copy the CPS utilities to the Source computer (or run them from a floppy disk).
- 5 Run UPLDPREF.

-  Running UPLDPREF (Upload Profile)
-  Setting Up the Target Computer

## Running UPLDPRF

UPLDPRF (Upload Profile) reads the Windows NT Registry, user account, and security information of the Source computer and generates the necessary information and Registry files for reinstallation, and then copies all the files to the Computer Profile directory.

### To run UPLDPRF:

- At the command prompt on the Source computer, type **uplodprf** with the appropriate switches:  
**uplodprf [switches] -s:driveletter -i:filename [-a | \dirs]**

Where:

#### **-q**

Suppress information messages. If you want to check progress messages, you can save them in a log file using MS-DOS redirection, then search the log file for "Error" to see if there were any problems in copying files. For example: **uplodprf -s\\mysys\profile -i:profile.INI -a > profile.log**.

#### **-b**

Copies only the boot sector to the specified file.

#### **-b-**

Does not copy the boot sector. (All other processing is performed.)

#### **-m**

Rescan the share directory and update the WINNT.INF file.

#### **-f:{f | n}**

Overrides the default file system to be installed on the Target computer. **:f** will install a FAT file system on the Target computer, and **:n** will install the NTFS file system. The default is to install the same file system as is found on the Source computer.

#### **-u**

Generates the user account definition file specified in the .INF file and nothing else.

#### **-u-**

Does not generate a user account definition file.

#### **-n**

Dumps the access control lists of the files found in the directory list (for NTFS volumes only).

#### **-n-**

Does not dump access control lists in the profile.

#### **-r**

Dumps the access control lists of the profiled Registry keys.

#### **-r-**

Does not dump the Registry key access control lists.

#### **-s:driveletter**

Specifies the location of the Computer Profile directory, which will contain the information derived from the Source Computer to be used by the Target computers for downloading and installation. This directory must be emptied before you run UPLDPRF.

#### **-i:filename**

Computer Setup information file.

#### **\dirs**

Additional directories from the Source computer's *SystemRoot* directory to be included in the profile.

#### **-a**

Saves the entire volume containing the Windows NT system. Using this switch only copies files in subdirectories of the root directory and not files in the root directory itself. Because root directory files may be special and not always desirable to copy to other computers, the files found in the root directory that are to be part of the profile must be manually added to the **[SystemFilesToSubstitute]** section of PROFILE.INI. An example of such a file is NTLDR. If you are not planning to use the **-a** switch and want to specify the directories to profile on the command line, then the default version of PROFILE.INI should work fine.

For example:

**uplodprf -s:z:\ -i:profile.ini \winword \excel \tools**

This command uses the PROFILE.INI file to do the following:

- Process the Source computer's system.
- Load the Computer Profile to the Z drive.
- Copy the Windows NT system directory tree to the Z drive for subsequent reinstallation.
- Copy the \WINWORD, \EXCEL and \TOOLS directory trees to the profile on the Z drive.

If there are any problems encountered during this conversion process, an error message appears.

Each subkey under SYSTEM and SOFTWARE in the Registry must have Full Control permissions or at least the following permissions for the Administrators group on the local computer:

Query Value, Set Value, Create Subkey, Enumerate Subkeys, Notify, Delete, Read Control

**Note:** If you have an exceptionally large configuration, the following message might appear when you run UPLDPRF: "(1003) Cannot complete this function. WINNTP.EXE input file will be too large to process. Try to create a profile with fewer directories."

The message occurs if the DOSNET.INF that UPLDPRF creates would be larger than 64K. If the resulting DOSNET.INF is too large for WINNTP to process, you must re-run UPLDPRF and select fewer directories to include in the profile. You can create another directory tree that contains the remaining files to be installed, and then use XCOPY to copy these files to the new installation after Windows NT installation is complete.

- Setting Up the Source Computer
- Setting Up the Target Computer

## Setting Up the Target Computer

On the Target computer, the WINNTP program must be run .

### To install the Computer Profile files on a Target computer:

- At the command prompt on the Target computer, type **winnntp** with the correct switches:

**winnntp [/D:sysroot] [/s:sourcepath] [/T:tempdrive] [/I:infile] [/X | /F] [/C] /R**

Where:

**/D[:]*sysroot***

Removes the Windows NT system files from the installation in *SysRoot* (the root directory for the Windows NT system files, which is usually C:\WINNT).

**/S:*sourcepath***

Specifies the source location of the Windows NT files for Computer Profile Setup, which contains the information derived from the Source computer. This must be a full path name of the form *x:\[path]* or *\server\share\[path]*.

**/T[:]*tempdrive***

Specifies a drive to contain temporary setup files. If this switch is not specified, Setup attempts to locate a drive for you.

**/I[:]*infile***

Specifies the filename (with no path) of the Setup information file. The default is DOSNET.INF, which is created during the upload process by UPLODPRF.

**/C**

Specifies to skip the free-space check on the Setup boot floppy you provide.

**/F**

Specifies not to verify files as they are copied to the Setup boot floppy.

**/X**

Specifies not to create the Setup boot floppy.

**/R**

Indicates that a Computer Profile directory is being downloaded (rather than the Windows NT compact disc). This switch is required when you install a profile. This overrides the **/C**, **/F**, and **/X** switches, since profile installations do not require a floppy disk.

For example:

**winnntp /s:z:\ /r**

WINNTP copies all files from the Computer Profile directory and installs them into the appropriate directories on the Target computer, then asks the user to restart the computer, after which the graphical portion of Windows NT Setup runs to complete the installation.

- Setting Up the Source Computer
- Running UPLODPRF (Upload Profile)



## Limitations for Computer Profile Setup

- All Computer Profile files to be included in the Computer Profile must be on the same drive as the Windows NT system directories.
- Only Common program groups are propagated with the Computer Profile.
- The CPS version of WINNTP must be used to install the Computer Profile on the Target computer.
- If the network cards cannot be autodetected, then the user will be prompted for settings during Setup on the Target computer.
- Windows NT must be installed on the boot drive of the Source computer (usually the C drive) before profiling, and will be installed on the boot drive when the profile is installed on the Target computer.
- UPLDPRF is unable to read the boot sector on IBM® PS/2® computers. An MS-DOS-based program, GETBS.EXE, is provided for this as a work-around. See UPLDPS2.BAT on the Resource Guide disks for an example.
- Before profiling, filenames that exceed the 8.3 MS-DOS format must be renamed. That is, a filename that is longer than 8 characters or a filename extension longer than 3 characters cannot be profiled. If you use the **-q** switch with UPLDPRF, a message will appear to warn you of any filenames encountered that are too long.

## Input (.INI) File for Computer Profile Setup

The input file used by UPLDPRF.EXE to create computer profiles is created manually by using or modifying the PROFILE.INI file supplied with the Resource Kit. The filename is entered as a parameter on the command line of the **uplodprf** program. The format must be that of a Microsoft Windows .INI file, where sections are indicated by [ ] characters, and keys and their corresponding values are separated by an equal sign (=).

In the description of the entries below, the values of the keys may be modified unless specifically prohibited or warned against. In any event, changing or adding value and keys, should be done only with a clear understanding (perhaps through experimentation) of the impact of the change.

### [HiveUpdateInfo]

Global constants for the program.

#### **RootHiveKey**

The key in the Registry where the **HiveRootName** (described below) will be created. The **HiveRootName** will be created as a subkey to this key. This should not be modified.

#### **HiveRootName**

The name of the subkey that will be created under the **RootHiveKey** key. The SOFTWARE and SYSTEM hives are copied under this key for manipulation into profile hives.

#### **INFUpdateFile**

The name of the file where the network information required by Setup will be written. This file will be appended to the **ProfileInfTemplateFile** file before being copied to the profile.

#### **ProfileInfTemplateFile**

The filename of the template file to serve as the basic .INF used by the Windows-based portion of Windows NT Setup. The update file (**INFUpdateFile**) is appended to this and copied to the file named by **ProfileInfPathName**.

#### **ProfileInfPathName**

This entry must not be modified--the value should be PROFILE.INI. It specifies the destination file to be created by appending the **INFUpdateFile** file to the **ProfileInfTemplateFile** file. This is the file that the Windows-based part of Setup will load when it is run on the profile Target computer.

#### **BOOT\_INI**

The file and path name of the BOOT.INI to use as a master. This file will be copied to the **TempDir** directory. The copy will be modified to have Timeout set to 0 before being copied to the profile.

#### **TempDir**

The directory where the temporary files are created. This directory will be created if it doesn't exist, and when **uplodprf** is completed, this directory and all files in it will be deleted.

#### **WinntInfFileName**

The name of the .INF file created by **uplodprf** and used by **winnttp** to download the files to the Target computer. The default is DOSNET.INF. This corresponds to the /i: switch for WINNTP.EXE

#### **UserAccountFile**

This entry should not be changed. It specifies the name of the file that will receive the dump of the Source computer's user account information.

#### **RegistryACLDumpFile**

This entry should not be changed. It specifies the name of the file to contain the dump of the Registry key access control lists (ACLs)

#### **NTFSFilesAcIDumpFile**

This entry should not be changed. It specifies the name of the file to contain the dump of file ACLs for files of an NTFS volume. This file is not created if the Source computer has a FAT system volume.

**Note:** This entry must not be changed.

### **[RegistryAcIDumpKeys]**

The Registry keys listed in this section, followed by an equals sign, indicate the Registry keys that will have their security information dumped to the **RegistryACLDumpFile**.

### **[ServiceBillboardTable]**

This section maps the billboard (the popup messages that appear during the Windows-based part of Setup) to the various network services installed during Setup. This mapping is defined in the ~PROFILE.INF file and should not be modified by the user.

### **[SystemDirsToIgnore]**

Directory paths (and all their subdirectories) listed in this section will not be copied to the profile.

### **[SystemDirsToCreate]**

Directory paths specified in this section will be created on the profile but not populated with any files. (They may have files added by using the next section, though.)

### **[SystemFilesToSubstitute]**

Entries in this section are used to replace or override files found in the system directories of the Source computer. The sequence of file copying is:

- 1 Copy all system files (that is, \WINNT) and subdirectories except those in the **[SystemDirsToIgnore]** directories.
- 2 Create empty directories from **[SystemDirsToCreate]**.
- 3 Copy files described in **[SystemFilesToSubstitute]**.

The format of these entries is to have the destination file and path on the left side of the equals (that is, the "key") and the source file from the Source computer on the right side (the "value").

### **[NTFSSystemFilesToSubstitute]**

Entries in this section behave identically to those in the above section, but are only copied if the Source computer's system drive is formatted to use the NTFS file system.

### **[FATSSystemFilesToSubstitute]**

Entries in this section behave identically to those in the above section, but are only copied if the Source computer's system drive is formatted to use the FAT file system.

### **[NetworkHiveInfo]**

Information used to extract and remove network configuration.

#### **FindInstalledServicesAt**

The root key whose subkeys will be enumerated and searched for entries that match the criteria below. The key name does not need to include reference to the fact that the "working" Registry keys are relocated. For example, if the keys that indicate network services are found under the SOFTWARE\Microsoft key, in a normal system, then that is the value that should be entered in this key, NOT SOFTWARE\Microsoft\HiveRootName\SOFTWARE\Microsoft. The program will modify the working version (that is, the copy), NOT the real version.

#### **InstalledServiceKey**

The subkey of the enumerated keys under the key above that, if present, indicate this is a network service.

#### **ServiceTypeKey**

The subkey of the enumerated subkeys for the **FindInstalledServicesAt** key that contains the value indicating the Type of network service (for example, Driver, Transport, Service, etc.) This is used to group the services in the correct arrangement for Setup (for example, to ensure that devices are installed before transports).

### **ServiceTypeKeyName**

The value name under the above key that indicates the type of network service.

### **FindServicesToRemoveAt**

The key whose keys will be enumerated and searched to build a list of Network Keys (services) that will be removed from the keys listed in **RemoveServiceFrom\_n**.

### **RemoveServiceKey**

The subkey of the enumerated subkeys of the key listed in **FindServicesToRemoveAt**, which if present indicates that this key is that of a network service that should be removed from the Registry for Setup.

### **RemoveServiceFrom\_1**

### **RemoveServiceFrom\_2**

### **RemoveServiceFrom\_3 ....**

The list of keys in the Registry that contain one or more of the networks services found under the **RemoveServiceKey** key. The dehydrator program builds a list of keys to network services using the **FindServicesToRemoveAt** key and then deletes all subkeys that match entries in that list found under the **RemoveServiceFrom\_x** keys.

### **[RemoveHiveKeys]**

Lists the subkeys that should be removed from a key.

The format for this section is:

- The "Key" is a root key in the Registry containing the subkeys to be deleted.
- The "Value", followed by a comma-separated list of subkeys to be removed or an "\*" if all subkeys are to be removed.

The processing of this section does not delete the root key ("Key").

### **[RemoveHiveValues]**

Lists the values that should be removed from a key.

Similar to the section above, in this section the "Key" is a Registry key containing values to be removed. The "Value" is a comma-separated list of the values to be deleted from this key.

### **[UpdateHiveValues]**

Lists values of Registry keys to be added or modified.

This section provides a list of values in the Registry that need to be added or changed. The format of the "Key" in this section is:

- The full key path
- A colon (":" )
- The value to be modified.

The "Value" is the new value to be written to the Registry. The value can be of several Registry data types:

- REG\_SZ: a single unquoted string (with no commas) will be stored in the Registry as a REG\_SZ data type.
- REG\_MULTI\_SZ: a comma-separated list will be stored as a MULTI\_SZ data type. Each entry between commas being a string. The commas are not stored.
- REG\_DWORD: a value beginning with the characters "0x" will be treated as hex value, converted to a DWORD and stored in the Registry as such.
- REG\_EXPAND\_SZ: a string containing the "%" will be stored as an REG\_EXPAND\_SZ type.



### **[NTFSHiveValues]**

These values are processed in the same fashion as those described in the **[UpdateHiveValues]** section. They are only applied if the Target computer's file system is NTFS.

### **[FATHiveValues]**

These values are processed in the same fashion as those described in the **[UpdateHiveValues]** section. They are only applied if the Target computer's file system is FAT.

See Also

-  [Computer Profile Setup Processing Overview](#)
-  [Security Dump File Formats for Computer Profile Setup](#)

## Computer Profile Setup Processing Overview

This section provides an overview of the processing performed by UPLDPRF. This amounts to the dehydration of the "installed" Registry into hives ready for use again by Computer Profile Setup. Sample entries from an .INI file illustrate the process in this discussion.

### Initialization

The dehydration process starts by copying the current, running version of the Registry (or at least the SYSTEM and SOFTWARE keys), to a subkey for further processing without running the risk of corrupting the working Registry. This is done by saving the SYSTEM and SOFTWARE keys to hive files, creating the working subkey and then loading the hive files as subkeys to this key. This is illustrated below using the excerpt from the .INI file shown here.

.INI FILE:

```
[HiveUpdateInfo]
RootHiveKey       = SOFTWARE\Microsoft
HiveRootName      = HiveUpdateRoot
```

Registry:

#### Registry Before Initialization

```
MACHINE
  DEFAULT
  SECURITY
  SOFTWARE
  SYSTEM
  USER
```

#### Registry After Initialization

```
MACHINE
  DEFAULT
  SECURITY
  SOFTWARE
    Microsoft
      HiveUpdateRoot
        SOFTWARE
        SYSTEM
  SYSTEM
  USER
```

The next step in the process is to distill the information necessary for rebuilding the Registry when the system is reinstalled. This consists mostly of the network services that will be reinstalled by Setup. The task of the dehydrator is to determine what network services are installed and will need to be reinstalled when Setup is run on the Target computer.

There are two steps to this:

- Identify the services to be reinstalled by Setup.
- Identify the keys to be removed from Registry that reference these services.

To accomplish this, information from the **[NetworkHiveInfo]** section of the .INI file is used. The **FindInstalledServicesAt** key indicates the key in the Registry whose subkeys may be network services that should be reinstalled. Because not all subkeys are likely to be network services, the **InstalledServiceKey** is used to identify those that are. The example below illustrates this relationship.

.INI FILE:

```
[NetworkHiveInfo]
FindInstalledServicesAt = SOFTWARE\Microsoft
InstalledServiceKey     = CurrentVersion\NetRules
ServiceTypeKey         = CurrentVersion
ServiceTypeKeyName      = SoftwareType
```

Registry:

```
MACHINE
  SOFTWARE
    Microsoft
      HiveUpdateRoot
        SOFTWARE
          Microsoft
```

```

Service_A
    CurrentVersion
Service_B
    CurrentVersion
        :SoftwareType = Device
    NetRules
Service_C
    CurrentVersion
        :SoftwareType = Service
    NetRules

```

In the example above, the subkeys of SOFTWARE\Microsoft (that is, the one under **HiveUpdateRoot**) will be enumerated and examined. Service\_A would not be selected as a network service to be reinstalled since it does not have a CurrentVersion\NetRules key under it. Service\_B and Service\_C would be saved since they contain the subkey that identifies them as a network service according to the rules listed in the .INI file shown above. This list is formatted and written to the file specified by the **INFUpdateFile** key in the .INI file.

Once the network services have been listed for reinstallation, their entries need be removed from the Registry. This is accomplished using a similar approach to that shown above. The Root Key and search criteria are identified by the following keys in the .INI file.

.INI FILE:

```

[NetworkHiveInfo]
FindServicesToRemoveAt = SYSTEM\ControlSet001\Services
RemoveServiceKey       = Linkage

```

In the above example, all subkeys of SYSTEM\ControlSet001\Services that have a Linkage subkey will be entered into a list used to identify all the keys that correspond to network services that need to be removed. The keys containing the network subkeys to be removed are then listed in the .INI file as shown below.

.INI FILE:

```

[NetworkHiveInfo]
RemoveServiceFrom_1 = SOFTWARE\Microsoft
RemoveServiceFrom_2 = SYSTEM\ControlSet001\Services
RemoveServiceFrom_3 = SYSTEM\ControlSet001\Services\EventLog\System
:

```

In this example, any subkey found under the keys listed above will be deleted from the Registry if it matches an entry in the list derived above. So all network services entries found under the keys below will be removed:

```

SOFTWARE
    Microsoft
        <net keys>
SYSTEM
    ControlSet001
        Services
            <net keys>
        EventLog
            System
                <net keys>

```

The remaining modifications are strictly from the information contained in the .INI file. The first step is to remove all unwanted subkeys (and all subkeys to them).

.INI FILE

```

[RemoveHiveKeys]
SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkCards = *
SOFTWARE\Microsoft\Windows NT\CurrentVersion\ProfileList = *

```

```
SOFTWARE\Secure = *
SYSTEM\ControlSet001\Control\NetworkProvider\Order = *
SYSTEM = ControlSet002, ControlSet003, ControlSet004, ControlSet005
```

The asterisk value "\*" indicates that all subkeys are to be removed. The Registry key identified as the "Key" in the .INI file excerpt above is not removed. If only certain subkeys are to be removed; then they are identified in a comma-separated list as shown in the last line of the .INI file excerpt above. Note that the entire key path must be specified.

Next, remove all unwanted values from subkeys as described in the .INI file.

The format for this command is similar to the above -- the only exception being that the .INI file "Value" is a value under the subkey listed in the .INI file "Key" as opposed to a subkey. If more than one value under a subkey is to be deleted, then they can be listed in a comma-separated list (no quotes). Note that the entire key path must be specified, as shown above.

#### .INI FILE:

```
[RemoveHiveValues]
SOFTWARE\Microsoft\Windows NT\CurrentVersion =
    CurrentVersion, CurrentBuild, CurrentType, SystemRoot, SourcePath
SOFTWARE\Microsoft\Windows NT\CurrentVersion\WinLogon =
    DebugServerCommand, CacheValid, CacheTrustedDomains,
    CacheTrustedPrimaryDomain, CacheLastController, CacheLastUpdate
SYSTEM\ControlSet001\Control\TimeZoneInformation =
    StandardName, StandardBias, StandardStart,
    DaylightName, DaylightBias, DaylightStart, ActiveTimeBias
SYSTEM\ControlSet001\Services\EventLog\Application =
    Sources, Network Control Panel, Replicator
SYSTEM\ControlSet001\Services\EventLog\Security = Sources
SYSTEM\ControlSet001\Services\EventLog\System = Sources
```

Finally, any new entries or entries that need to be modified are entered into the Registry using the section of the .INI file listed below.

The format of the .INI file key is slightly different for this section than it is for the previous sections. For this section:

- The .INI file "Key" contains both the full key path as well as the value to be added or modified.
- The .INI file "Value" contains the new value for that Registry entry.

The format of the .INI file "Value" determines how that value will be interpreted and stored in the Registry. The four types of data that can be stored using the method are:

- REG\_SZ: a single unquoted string (with no commas) will be stored in the Registry as a REG\_SZ data type.
- REG\_MULTI\_SZ: a comma-separated list will be stored as a MULTI\_SZ data type, each entry between commas being a string. The commas are not stored.
- REG\_DWORD: a value beginning with the characters "0x" will be treated as hex value, converted to a DWORD and stored in the Registry as such.
- REG\_EXPAND\_SZ: a string containing the "%" will be stored as an REG\_EXPAND\_SZ type.

The .INI file excerpt below shows one of each data type:

#### .INI FILE:

```
[UpdateHiveValues]
SYSTEM\ControlSet001\Control\ProductOptions:ProductType = WinNt
SYSTEM\ControlSet001\Control\Sample:MultiSZ = string1, string2
SYSTEM\ControlSet001\Services\EventLog:Start = 0x00000004
SYSTEM\ControlSet001\Control\Session Manager\Environment:Path =
    \%SystemRoot%\system32
```

- Input (.INI) File for Computer Profile Setup
- Security Dump File Formats for Computer Profile Setup





## Security Dump File Formats for Computer Profile Setup

The Source computer's security information is dumped to three files that are named in the **[HiveUpdateInfo]** section of the .INI file. These files are (referencing their name key in the .INI file):

User Accounts	= <b>UserAccountFile</b>	= USERACCT.INF
Registry Keys	= <b>RegistryAcIDumpFile</b>	= REGISTRY.ACL
NTFS files	= <b>NTFSFilesAcIDumpFile</b>	= NTFSFILE.ACL

### User Account Dump File Format

The user account dump file is a hierarchical dump of the accounts in the Source computer's SAM database. The hierarchy of objects is shown here:

- Computer Name (SAM database)
  - Domains
    - Groups
      - user accounts
  - Alias
    - user accounts

These objects are decomposed into a text format in the user account dump file.

#### [General]

Contains information that applies to all other sections

##### **SourceComputerName =**

The name of the computer that the file was generated from. This is used to identify and differentiate "local" computer information from network, or domain computer information.

#### [Domains]

Lists the domains found in this database.

##### **Domain\_x =**

Each domain is enumerated starting with x= 0 to x=n where there are n+1 domains in the database.

For each domain, there's a section with the domain name as the section name and listed under that section are all the groups and aliases of that section in an enumerated fashion:

##### **Group\_x =**

:

:

##### **Alias\_y =**

:

:

For each group and alias under a domain section, they, too, have their own section containing the attribute and account information. The format of the section name is:

[Group: domain\group] (for groups)

[Alias: domain\alias] (for alias)

Listed under each section header is:

##### **AdminComment = <comment string>**

The comment or description of the group or alias.

##### **Attributes = <digit>**

The value of the attribute byte for the group.

##### **Member\_x = domain\user [Security ID-dump]**

The enumerated list of members in that group after each member entry is the domain and user

account name and the Security ID (in dumped format) for that domain.



### **Registry Key Dump Format**

### **NTFS File Dump Format**

The formats of these two files are similar. The main difference is that the section names in the Registry key dump file are the names of Registry keys, while in the NTFS File Dump file, the section names are the pathnames of the files. Following the Section header is the value of the Access Control List (ACL) header that is used in the regeneration of the ACL for that item.

Under each section header (that is, key name or file name) is a list of each of the Access Control Entries (ACE) in the object's ACL. Each line in the file constitutes one ACE in the ACL. The format of each ACE entry is to have the domain\account-name string followed by the dumped Security ID on the left of the equals, and on the right, the contents of the ACE header (H:) and the ACE Access mask (A:) dumped in hexadecimal format.

See Also

-  [Input \(.INI\) File for Computer Profile Setup](#)
-  [Computer Profile Setup Processing Overview](#)

## Copyright Information

Information in this online document is subject to change without notice.

Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Microsoft Corporation.

© 1985-1993 Microsoft Corporation. All rights reserved.

Microsoft is a registered trademark and Windows NT is a trademark of Microsoft Corporation.

Adaptec is a trademark of Adaptec Inc. Apple, AppleShare, LocalTalk, Macintosh are registered trademarks and System 7 is a trademark of Apple Computer, Inc. Compaq is a registered trademark of Compaq Computer Corporation. DEC is a registered trademark of Digital Equipment Corporation. Everex is a trademark of Everex Systems, Inc. HP is a registered trademark of Hewlett-Packard Company. IBM and PS/2 are registered trademarks of International Business Machines Corporation. Intel is a registered trademark of Intel Corporation. Novell is a registered trademark of Novell, Inc. SMC is a registered trademark of Standard Microsystems Corporation. Toshiba is a registered trademark of Kabushiki Kaisha Toshiba. UngermanBass is a registered trademark of Ungermann-Bass, Inc.

