

Strider GhostBuster:
Why It's A Bad Idea For Stealth Software To Hide Files

Yi-Min Wang
Binh Vo
Roussi Roussev
Chad Verbowski
Aaron Johnson

July 24, 2004

Technical Report
MSR-TR-2004-71

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Strider *GhostBuster*: Why It's A Bad Idea For Stealth Software To Hide Files

Yi-Min Wang, Binh Vo, Roussi Roussev, Chad Verbowski, and Aaron Johnson

Microsoft Research, Redmond

July 2004

Abstract

File hiding is an advanced stealth technique that is becoming popular among system monitoring software such as RootKits, Trojans, and keyloggers. It presents a major challenge to system administrators and the anti-malware industry because detection and removal are virtually impossible if the target files are not even visible. In this paper, we present the Strider *GhostBuster* that exploits the fundamental weakness of the file-hiding behavior and turns the problem into its own solution. We have tested this diff-based tool successfully in the lab against several real-world system monitoring programs. The simplicity and effectiveness of the approach suggest that the following quote on the Internet may no longer be true: “*When you can get the dir command to lie, it's all over.*” In the post-*GhostBuster* world: “*The best way to hide is not trying to hide.*”

[February 23, 2005: note that Strider *GhostBuster* uses a “*cross-view diff*” technique, which is very different from the usual “*cross-time diff against known good*” approach. Please see the new technical report titled “*Detecting Stealth Software with Strider GhostBuster*” posted at <http://research.microsoft.com/rootkit> for a detailed discussion.]

The Strider *GhostBuster*

File-hiding [HTB03, NTI04] is an advanced stealth technique used by many popular system-monitoring software such as RootKits [RK], Trojans, and keyloggers [KL] to make executables or data files (such as recorded keystroke files) invisible. Once the monitoring program is started, it intercepts all file queries at a very low level and uses filtering to ensure that a chosen subset of files are never revealed to any file query operations made by any program running on the infected machine. This technique can defeat experienced system administrators who search the file system and Windows Registry for suspicious entries, as well as commonly used malware scanning tools that are based on known-bad file signatures.

However, file-hiding behavior has a fundamental flaw: any file to be hidden must physically exist and is visible from another clean OS uninfected with the file-hiding software. Unfortunately, this visibility alone does not mean that the hidden file is easily identifiable because there are

usually tens to hundreds of thousands of files on a file system. Fortunately, the file-hiding behavior itself provides an almost trivial way of exposing the very file that it tries to hide: if we scan the entire file system from the clean OS, and subtract from the output all those unhidden files that appear in another scan inside the infected OS, the only remaining files are the hidden files!

Figure 1 illustrates a straightforward implementation of the above technique, referred to as *ScanDiff*. It consists of three steps.

Step #1: We first boot normally into the infected OS and invoke “*dir /s /a*” to scan the entire file system. We save the output in a file named “*Infected_Scan.txt*” on a disk. The file-hiding software can arbitrarily interfere with the scanning process and/or arbitrarily modify the output file¹. (Note that the user account from which the scan is performed should be added to the ACLs of the System Volume Information folder and other folders that by default are not accessible to the user².)

Step #2: We restart the machine³ and this time boot into a clean WinPE CD⁴ [WPE] that contains a clean version of *WinDiff.exe*. We invoke “*dir /s /a*” again and save the output in the file “*Clean_Scan.txt*”. The hidden file should appear in this output because the file-hiding software was not running during the scan.

Step #3: Finally, we invoke *WinDiff.exe* to compare the two files “*Infected_Scan.txt*” and “*Clean_Scan.txt*”. Any hidden file should be revealed in the diff result⁵.

We have tested *ScanDiff* on many real-world file-hiding RootKits, Trojans, and commercial keyloggers to demonstrate its effectiveness. Figure 2 through Figure 5 display the *WinDiff* screenshots for four RootKits/Trojans: *Hacker Defender 1.0*, *Aphex - AFX Windows Rootkit 2003*, *Vanquish*, and *Msvsres.dll* (which came from the wild), respectively. Figure 6 and Figure 7 show the results for two commercial file-hiding keyloggers *ActMon* and *ProBot SE*, respectively. We also tested *ScanDiff* on four commercial software programs that are designed specifically for the purpose of hiding selected files: *Hide Files 3.3*, *Hide Folders XP*, *Advanced Hide Folders*,

¹ The software can also try to “hang” the “*dir*” scanning process so that it never finishes. But that symptom itself would indicate the existence of malware.

² Otherwise, files inside such folder will not appear in the infected scan and will appear in the final *WinDiff* result as noise.

³ “Reset” is preferred to “Shutdown” because the latter may cause additional files to be modified during the shutdown process and introduce more noise in the final *WinDiff* result.

⁴ Think the movie “*The Matrix*”: scanning inside the infected OS is like living inside the Matrix where you can only see what they want you to see; you need to get out of the Matrix in order to see the truth. WinPE CD boot is the way to get out.

⁵ Think the movie “*The Sixth Sense*” and ask the question: “the kid can see both humans and ghosts, but how can he distinguish a ghost without any “known-bad signatures” (i.e., wounds) from a human?” The answer is he can ask his mom, who can’t see the ghosts, to take a look and take a diff of their views.

and *File & Folder Protector*. All files hidden by these programs were exposed in the *WinDiff* results.

A slight enhancement to this method can make it even more effective against keyloggers specifically: in Step #1, after the “*dir /s /a*” scan, we can type in a large number of keystrokes to try to increase the size of the keystroke log file⁶. This maximizes the chance that the log file would appear in the *WinDiff* result either because it’s hidden or because its size has changed between the two scans. Figure 8 through Figure 13 show the screenshots that expose six non-file-hiding commercial keyloggers by detecting their changed log files in response to our artificial keystrokes.

We have built a tool called the Strider *GhostBuster* that automates most of the above *ScanDiff* steps. A user puts a WinPE-based *GhostBuster* CD into her CD ROM player. Our program will auto-start, flush all disk caches to establish a solid disk image for the first scan, perform a file system scan⁷ to get “*Infected_Scan.txt*”, prompt the user to type in a large number of keystrokes and press “*Enter*”. The program then flushes all disk caches again to make sure that all updates since the first image are committed, prompts the user to press the machine’s “*Reset*” button, and instructs her to select the boot-from-CD option. Once the machine boots into WinPE, our program will auto-start, perform another file system scan to get “*Clean_Scan.txt*”, and invoke *WinDiff.exe* to display the result. The entire process currently takes 10 to 15 minutes. Figure 14 illustrates how the *GhostBuster* tool captured the CWS spyware protected (i.e., hidden) by *Hacker Defender* [CWS04].

The top portion of Figure 14 also illustrates an extension of the *ScanDiff* idea to detect Registry-hiding behavior. In the Strider Gatekeeper paper [WRV+04], we have demonstrated that monitoring and managing *Auto-Start Extensibility Points (ASEP)*, most of which reside in Windows Registry, is an effective way to detect and disable spyware programs because they strongly prefer hooking the system in a way that they can be automatically started upon every reboot without explicit user invocation. But if an ASEP scan is performed on a machine infected by Registry-hiding software, the ASEP hooks of some spyware programs may not show up. Figure 14 shows that such hidden ASEP hooks can be easily exposed by taking a diff of the infected and uninfected ASEP scans.

⁶ Any network connection and other storage media should be disconnected/turned off to ensure that the keylogger program must write the newly recorded keystrokes to a disk included in the scan. A keylogger may try to defeat this by pre-allocating a large log file so that it can accommodate newly recorded keystrokes without increasing its size. We can detect this by comparing file hash instead of just file size, but that would take much longer. We can also try to attack this method by pumping garbage keystrokes to fill up the log file.

⁷ The tool is being extended to handle files with multiple streams inside by basically treating each file as a potential folder.

Conclusions

There is a quote on the Internet [WRK] that says: “*When you can get the dir command to lie, it’s all over.*” It may be true for most of the inside-the-box solutions. But when we think “outside the box” (i.e., the infected OS), we can turn the program’s file-hiding behavior against itself and effectively turn the problem into its own solution. So the best way to hide now is not trying to hide.

We make the following four contributions in the *GhostBuster ScanDiff* work: (1) it can be used to deterministically, efficiently, and effectively detect today’s file-hiding software; (2) if widely adopted, it will help computer users gain back trustworthy file-query operations⁸, and force malware programs to give up file hiding and therefore always expose themselves to Gatekeeper ASEP scan and anti-virus-style known-bad signature-based scans; (3) it does not require known-bad signatures so there is no hassle for signature updates; and (4) the approach makes the clean assumption that *any data gathered through any apps or OS components running inside an infected OS cannot be trusted*, thereby avoiding an arms-race solution that essentially provides testing for file-hiding malware and helps eventually produce perfect malware.

In the bigger picture, *ScanDiff* is merely an optimization in our plan for a comprehensive RootKit detection solution. To detect non-file-hiding malware, we have augmented *GhostBuster* with a cross-time diff as follows. After each clean scan of the file system and the ASEPs from the *GhostBuster* CD boot, we save the output files to a floppy disk or a USB drive (that is never connected to the machine during any normal, potentially infected boot⁹). We then compare the latest scan with the previous scan to detect any newly added/updated files and ASEP hooks.

Alternatively, we can perform known-good filtering on the file and ASEP scan to help expose unknown malware. File and ASEP scans from other known-good, golden machines can provide such a known-good list. Or, we can compute a cryptographic hash of every file on the potentially infected disk and match it against the Strider Known-* Database, which currently contains millions of known-good file hashes from NIST NSRL [NSRL] and known-bad file hashes from our own malware investigation. As we continue to populate the database towards a comprehensive coverage, only a small number of files will have unknown file hashes and need to be investigated.

⁸ Malware programs cannot simply target GhostBuster and expose themselves only to GhostBuster because any file query operation may be replayed from a clean OS to allow a diff.

⁹ Or we sign each output file with an external key so that it cannot be tampered with. If a malware program deletes the file, the deletion itself is an indication of the existence of malware.

References:

- [CWS04] CWS spyware protected by Hacker Defender RootKit, Spyware Weekly Newsletter, June 2004. <http://www.spywareinfo.com/newsletter/archives/0604/8.php>.
- [HTB03] "How to become unseen on Windows NT," <http://rootkit.host.sk/knowhow/hidingen.txt>.
- [IKS03] "Guilty Plea in Kinko's Keystroke Caper," July 2003, <http://www.securityfocus.com/news/6447>.
- [KL] Keyloggers, <http://skrasavi.ds.uiuc.edu/Others/keyloggers.htm>.
- [NSRL] National Software Reference Library, <http://www.nsl.nist.gov/>.
- [NTI04] "NTIllusion: A portable Win32 userland rootkit," Phrack Magazine, July 2004. http://www.phrack.org/phrack/62/p62-0x0c_Win32_Portable_Userland_Rootkit.txt
- [RK] RootKits, <http://www.rootkit.com/>.
- [WPE] Microsoft Windows Preinstallation Environment (Windows PE), <http://www.microsoft.com/licensing/programs/sa/support/winpe.msp>.
- [WRK] Windows Rootkits, <http://www.umn.edu/oit/security/WindowsRootkits.pdf>.
- [WRK03] "Windows Root Kits a Stealthy Threat," <http://www.securityfocus.com/news/2879>.
- [WRV+04] Yi-Min Wang, Roussi Roussev, Chad Verbowski, and Aaron Johnson, "Gatekeeper: Monitoring Auto-Start Extensibility Points (ASEPs) for Spyware Management," under revision, accepted for publication by *Usenix LISA*, Nov. 2004. Available at <http://research.microsoft.com/sm/strider/>.

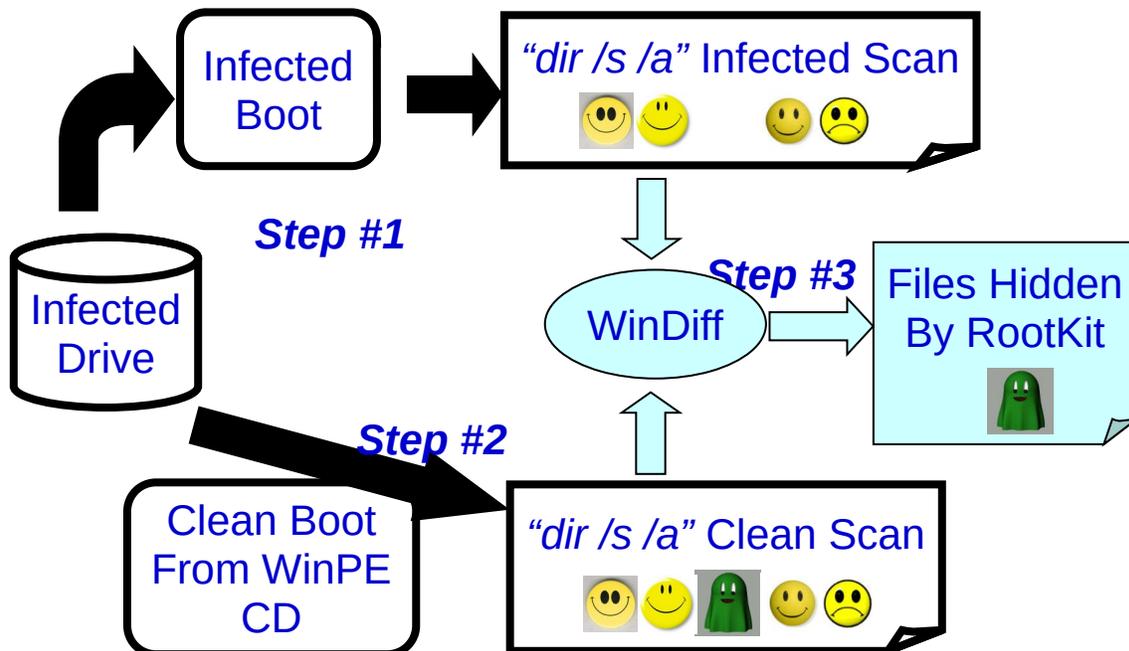


Figure 1. The *ScanDiff* approach to exposing file-hiding software.

Line	Op	Date	Time	File Name	Size
11	<!	06/15/2004	11:43 AM	0 Strider XPBoot Scan.txt	
[11]	!>	06/15/2004	11:47 AM	0 Strider PEBoot Scan.txt	
	!>	06/15/2004	11:43 AM	909,129 Strider XPBoot Scan.txt	
15	<!			4 File(s)	305,352 bytes
[15]	!>			5 File(s)	1,214,481 bytes
	!>	06/14/2004	04:34 PM	1,665 hxdef.czweb[1].htm	
1111	<!			26 File(s)	116,041 bytes
[1111]	!>			27 File(s)	117,706 bytes
	!>	06/14/2004	04:35 PM	1,460 hxdef100[1].zip	
1200	<!			26 File(s)	773,435 bytes
[1200]	!>			27 File(s)	774,895 bytes
	!>	06/14/2004	11:25 AM	455 hxdef100.lnk	
1247	<!			21 File(s)	11,396 bytes
[1247]	!>			22 File(s)	11,851 bytes
	!>	06/15/2004	11:27 AM	<DIR>	hxdef100
	!>			Directory of C:\tools\hxdef100	
	!>				
	!>	06/15/2004	11:27 AM	<DIR>	-
	!>	06/15/2004	11:27 AM	<DIR>	--
	!>	12/31/2003	12:15 PM	70,144 hxdef100.exe	
	!>	12/31/2003	12:17 PM	3,872 hxdef100.ini	
	!>	06/14/2004	02:45 PM	3,342 hxdefdrv.sys	
	!>			3 File(s)	77,358 bytes
	!>				
	!>	06/15/2004	11:27 AM	8,710 HXDEF100.EXE-3688A7E9.pf	
	!>	06/15/2004	11:27 AM	5,024 HXDEFá\$.EXE-2229B4D2.pf	

Figure 2. RootKit Hacker Defender 1.0 exposed by GhostBuster: all “hxdef*” files.

Line	Op	Date	Time	File Name	Size
11	<!	06/16/2004	01:35 PM	0 Strider XPBoot Scan.txt	
[11]	!>	06/16/2004	01:39 PM	0 Strider PEBoot Scan.txt	
	!>	06/16/2004	01:36 PM	875,486 Strider XPBoot Scan.txt	
15	<!			4 File(s)	305,352 bytes
[15]	!>			5 File(s)	1,180,838 bytes
	!>	06/16/2004	01:17 PM	6,450 AFXROOTKIT.EXE-1C35E349.pf	
11014	<!			100 File(s)	3,130,806 bytes
[1101]	!>			101 File(s)	3,137,256 bytes
	!>	06/16/2004	01:14 PM	190,464 AFXRootKit.exe	
	!>	06/16/2004	01:29 PM	88,576 explorer.dll	
	!>	06/16/2004	01:29 PM	82,944 iexplore.dll	
14287	<!			1745 File(s)	269,646,656 bytes
[1428]	!>			1748 File(s)	270,008,640 bytes
17922	<!			13463 File(s)	2,606,115,926 bytes
[1792]	!>			13468 File(s)	2,607,359,846 bytes
17923	<!			1782 Dir(s)	14,162,399,232 bytes free
[1792]	!>			1782 Dir(s)	14,161,522,688 bytes free

Figure 3. RootKit Apex - AFX Windows Rootkit 2003 exposed by GhostBuster: “AFX*” files, explorer.dll, and iexplore.dll.

Line	Op	Date	Time	File	Size
[11]	<!	06/16/2004	02:33 PM	0 Strider_XPBoot_Scan.txt	
11	!>	06/16/2004	02:37 PM	0 Strider_PEBoot_Scan.txt	
12	!>	06/16/2004	02:33 PM	871,596 Strider_XPBoot_Scan.txt	
14	!>	06/16/2004	02:12 PM	117 vanquish.log	
	<!			4 File(s)	305,352 bytes
17	!>			6 File(s)	1,177,065 bytes
	<!			6 File(s)	2,467 bytes
1162	!>	06/16/2004	01:51 PM	347 vanquish-0.2.0.zip.lnk	
1163	!>	06/16/2004	01:51 PM	253 Vanquish.lnk	
1164	!>			8 File(s)	3,067 bytes
6797	!>	02/23/2004	05:29 PM	49,152 vanquish.dll	
6798	!>	02/23/2004	05:29 PM	24,576 vanquish.exe	
[6815]	<!			84 File(s)	5,767,174 bytes
6815	!>			86 File(s)	5,840,902 bytes
10920	!>	06/16/2004	02:12 PM	4,648 VANQUISH.EXE-298C1D5C.pf	
	<!			99 File(s)	2,995,956 bytes
10948	!>			100 File(s)	3,000,604 bytes

Figure 4. RootKit Vanquish exposed by GhostBuster: all “vanquish*” files.

Line	Op	Date	Time	File	Size
10	!>	06/30/2004	10:36 AM	0 Clean_Scan.txt	
	<!	06/30/2004	10:32 AM	0 Infected_Scan.txt	
[10]	!>	06/30/2004	10:33 AM	872,662 Infected_Scan.txt	
15	<!			4 File(s)	305,352 bytes
[15]	!>			5 File(s)	1,178,014 bytes
990	<!	06/30/2004	10:21 AM	<DIR> Temp	
[990]	!>	06/30/2004	10:23 AM	<DIR> Temp	
2318	<!	06/30/2004	10:21 AM	<DIR> RP22	
[2318]	!>	06/30/2004	10:28 AM	<DIR> RP22	
12975	<!	05/11/2004	10:46 AM	<DIR> ias	
[1297]	!>	05/12/2004	07:16 AM	<DIR> ias	
	!>	01/23/27136	12:25 PM	18,944 msvsres.dll	
13900	<!	05/11/2004	10:44 AM	<DIR> spool	
[1390]	!>	05/12/2004	07:05 AM	<DIR> spool	
14236	<!			1745 File(s)	269,647,323 bytes
[1423]	!>			1746 File(s)	269,666,267 bytes
16484	<!	05/12/2004	07:08 AM	<DIR> Trace	
[1648]	!>	05/12/2004	07:09 AM	<DIR> Trace	
17871	<!			13404 File(s)	2,625,493,033 bytes
[1787]	!>			13406 File(s)	2,626,384,639 bytes
17872	<!			1794 Dir(s)	13,709,971,456 bytes free
[1787]	!>			1794 Dir(s)	13,709,090,816 bytes free

Figure 5. Trojan Msvsres.dll exposed by GhostBuster.

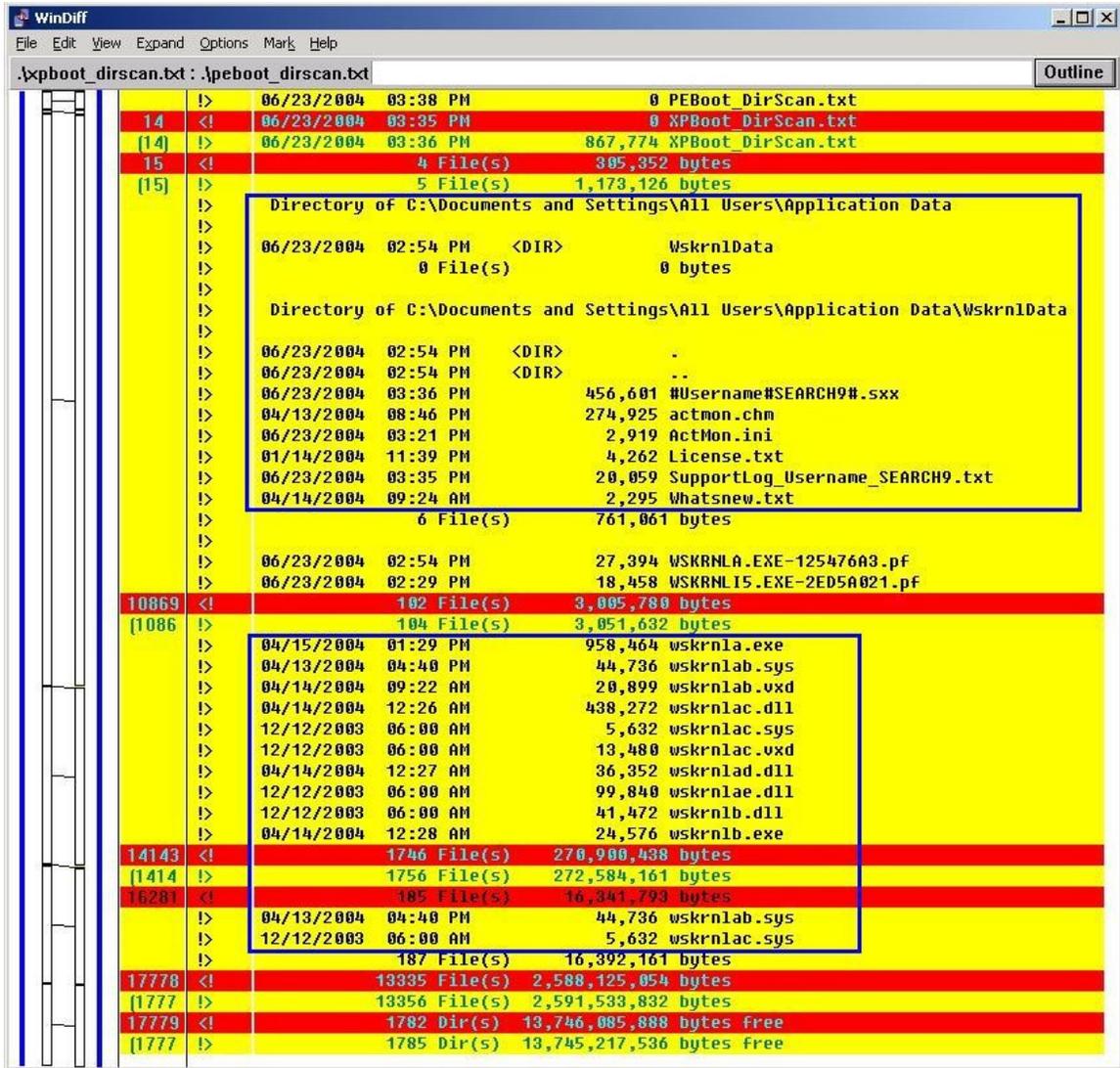


Figure 6. Commercial keylogger ActMon exposed by GhostBuster: all “ActMon*” and “wskrnl*” files.

Line	Op	Date	Time	Size	Name
	!>	07/01/2004	03:08 PM	0	Clean_DirScan.txt
10	<!>	07/01/2004	03:05 PM	0	Infected_DirScan.txt
(10)	!>	07/01/2004	03:05 PM	873,844	Infected_DirScan.txt
15	<!>			4 File(s)	305,952 bytes
	!>			5 File(s)	1,179,196 bytes
991	<!>	07/01/2004	03:00 PM	<DIR>	Temp
(991)	!>	07/01/2004	03:02 PM	<DIR>	Temp
	!>	08/29/2002	03:41 AM	262,144	olafol198.dll
	!>	08/29/2002	03:41 AM	131,584	ugjpub47.exe
14232	<!>			1745 File(s)	269,647,360 bytes
(1423)	!>			1747 File(s)	270,041,088 bytes
	!>	08/29/2002	03:41 AM	6,672	ayiqes73.sys
	!>	08/29/2002	03:41 AM	15,554	evevam71.sys
16370	<!>			185 File(s)	16,341,793 bytes
(1637)	!>			187 File(s)	16,364,019 bytes
17905	<!>			13427 File(s)	2,613,014,966 bytes
(1790)	!>			13432 File(s)	2,614,304,764 bytes
17906	<!>			1797 Dir(s)	13,723,025,408 bytes free
(1790)	!>			1797 Dir(s)	13,722,148,864 bytes free

Figure 7. Commercial keylogger *ProBot SE* exposed by *GhostBuster*: the four dll/exe/sys files with random file names.

Line	Op	Date	Time	Size	Name
	!>	07/07/2004	12:37 PM	0	Clean_DirScan.txt
[12]	<!>	07/07/2004	12:31 PM	0	Infected_DirScan.txt
12	!>	07/07/2004	12:32 PM	948,730	Infected_DirScan.txt
[20]	<!>	05/12/2004	07:30 AM	<DIR>	System Volume Information
20	!>	07/07/2004	12:35 PM	<DIR>	System Volume Information
[24]	<!>			10 File(s)	805,893,126 bytes
24	!>			11 File(s)	806,841,856 bytes
[1647]	<!>	07/07/2004	12:31 PM	1,024	ntuser.dat.LOG
1647	!>	07/07/2004	12:32 PM	20,480	ntuser.dat.LOG
[1654]	<!>			3 File(s)	787,636 bytes
1654	!>			3 File(s)	807,000 bytes
[2667]	<!>	07/07/2004	12:27 PM	83	ik.dat
2667	!>	07/07/2004	12:32 PM	399	ik.dat
[2676]	<!>			11 File(s)	593,574 bytes
2676	!>			11 File(s)	593,890 bytes
[3308]	<!>	05/12/2004	07:30 AM	<DIR>	.
3308	!>	07/07/2004	12:35 PM	<DIR>	.
[3309]	<!>	05/12/2004	07:30 AM	<DIR>	..
3309	!>	07/07/2004	12:35 PM	<DIR>	..

Figure 8. Commercial keylogger *Invisible Keylogger* exposed by *GhostBuster*: keystroke log file "ik.dat" is exposed due to size increase.

Line #	Op	Date/Time	Size	File Name
	!>	07/06/2004 12:19 PM	0	Clean_DirScan.txt
10	<!	07/06/2004 12:15 PM	0	Infected_DirScan.txt
[10]	!>	07/06/2004 12:15 PM	873,199	Infected_DirScan.txt
15	<!		4 File(s)	305,352 bytes
	!>		5 File(s)	1,178,551 bytes
1042	<!	07/06/2004 12:02 PM	<DIR>	W2SE0.TMP
[1042]	!>	07/06/2004 12:03 PM	<DIR>	W2SE0.TMP
2374	<!	07/06/2004 12:10 PM	<DIR>	RP21
[2374]	!>	07/06/2004 12:13 PM	<DIR>	RP21
6748	<!	07/06/2004 12:14 PM	439	iks.dat
[6748]	!>	07/06/2004 12:16 PM	2,444	iks.dat
6837	<!		85 File(s)	5,768,800 bytes
[6837]	!>		85 File(s)	5,770,808 bytes
12984	<!	05/11/2004 10:46 AM	<DIR>	ias
[1298]	!>	05/12/2004 07:16 AM	<DIR>	ias
13909	<!	05/11/2004 10:44 AM	<DIR>	spool
[1390]	!>	05/12/2004 07:05 AM	<DIR>	spool
16494	<!	05/12/2004 07:08 AM	<DIR>	Trace
[1649]	!>	05/12/2004 07:09 AM	<DIR>	Trace
17881	<!		13419 File(s)	2,618,116,733 bytes
[1788]	!>		13420 File(s)	2,618,991,937 bytes
17882	<!		1785 Dir(s)	13,718,495,232 bytes free
[1788]	!>		1785 Dir(s)	13,717,610,496 bytes free

Figure 9. Commercial keylogger *Invisible KeyLogger Stealth (IKS)* exposed by *GhostBuster*: the keystroke file "iks.dat" is exposed due to size change. (This is the keylogger involved in the Kinko's case [IKS03].)

Line #	Op	Date/Time	Size	File Name
	!>	07/08/2004 08:07 AM	0	Clean_DirScan.txt
11	<!	07/08/2004 08:02 AM	0	Infected_DirScan.txt
[11]	!>	07/08/2004 08:02 AM	950,595	Infected_DirScan.txt
19	<!	07/07/2004 08:38 AM	<DIR>	System Volume Information
[19]	!>	07/08/2004 08:06 AM	<DIR>	System Volume Information
23	<!		10 File(s)	805,893,126 bytes
[23]	!>		11 File(s)	806,843,721 bytes
3299	<!	07/07/2004 08:38 AM	<DIR>	.
[3299]	!>	07/08/2004 08:06 AM	<DIR>	.
3300	<!	07/07/2004 08:38 AM	<DIR>	..
[3300]	!>	07/08/2004 08:06 AM	<DIR>	..
15493	<!	07/08/2004 07:58 AM	<DIR>	v2act
[1549]	!>	07/08/2004 08:01 AM	<DIR>	v2act
19031	<!	08/29/2002 03:41 AM	233,464	2D8056F912AF49109FC6224C6DB4B9B1F4716C5C.tpr
[1903]	!>	08/29/2002 03:41 AM	288,624	2D8056F912AF49109FC6224C6DB4B9B1F4716C5C.tpr
19040	<!	08/29/2002 03:41 AM	5,839	BDC3CCE5166FF6FAC51712174DB5931DA9D1D8C7.tpr
[1904]	!>	08/29/2002 03:41 AM	6,927	BDC3CCE5166FF6FAC51712174DB5931DA9D1D8C7.tpr
19045	<!		16 File(s)	2,062,288 bytes
[1904]	!>		16 File(s)	2,118,536 bytes
19561	<!		14071 File(s)	3,461,411,701 bytes
[1956]	!>		14072 File(s)	3,462,418,544 bytes
19562	<!		2349 Dir(s)	13,717,962,752 bytes free
[1956]	!>		2349 Dir(s)	13,716,877,312 bytes free

Figure 10. Commercial keylogger *Spector Pro 5.0* exposed by *GhostBuster*: the "*.tpr" log files are exposed due to size increase.

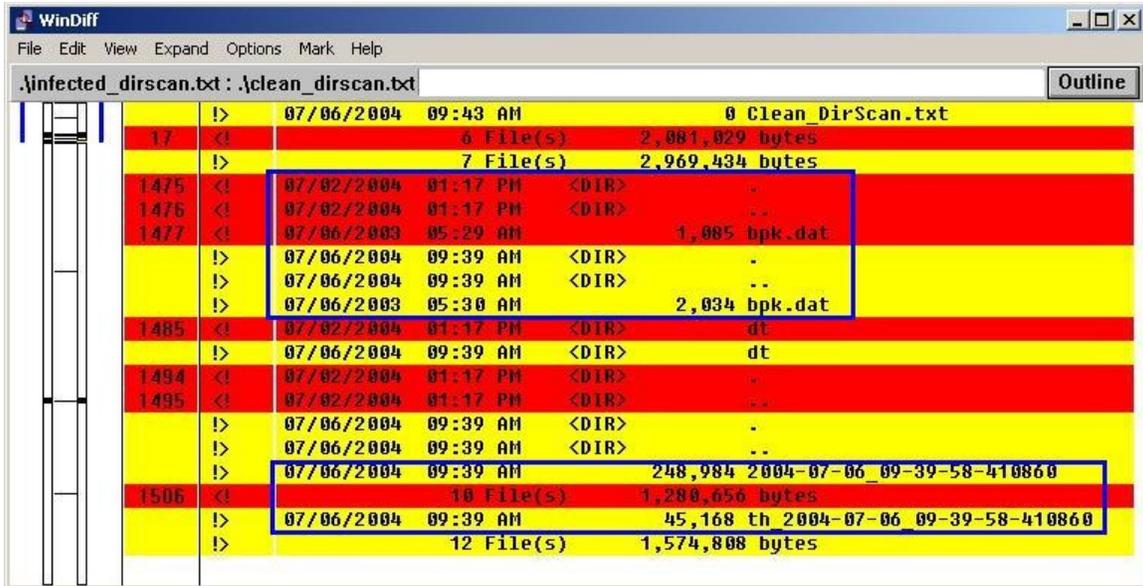


Figure 11. Commercial keylogger *Perfect Keylogger* exposed by *GhostBuster*: the "bpk.dat" log file exposed due to size change, the other two new "*2004*" log files also exposed.

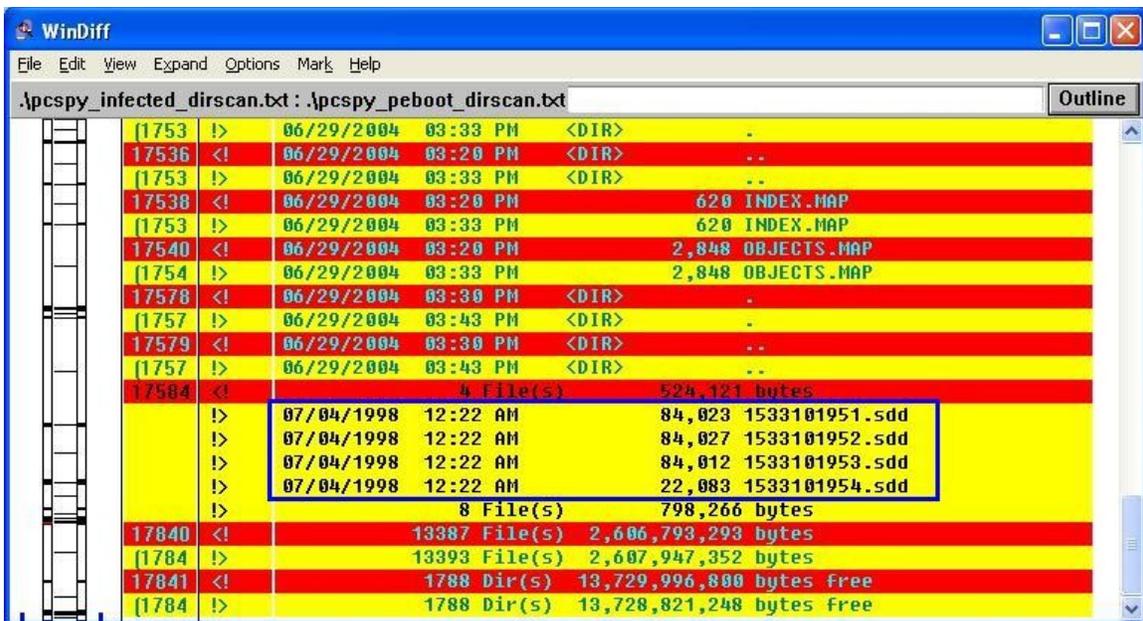


Figure 12. Commercial keylogger *PC Spy* exposed by *GhostBuster*: the new "*.sdd" screenshot log files exposed.



Figure 13. Commercial keylogger Win-Spy exposed by GhostBuster: the “Keylog.txt” log file exposed due to size increase.

```

<entry>
  <Name>{E3EEBBE8-9CAB-4C76-B26A-747E25EBB4C6}</Name>
  <DisplayName>COOLWE~1.DLL</DisplayName>
  <ImageName>COOLWE~1.DLL</ImageName>
  <Type>1</Type>
  <DataLen>2</DataLen>
  <Data></Data>
  <Context>path = HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\ToolBar\
value {E3EEBBE8-9CAB-4C76-B26A-747E25EBB4C6} =
CLSID\{E3EEBBE8-9CAB-4C76-B26A-747E25EBB4C6}\InprocServer32\ = C:\PROGRA~1\COOLWE~1.INF\COOLWE~1.DLL</Context>
</entry>
  <Name>HackerDefender100</Name>
  <DisplayName>HackerDefender100</DisplayName>
  <ImageName>C:\hxdeF100\hxdeF100.exe</ImageName>
  <Type>10000000</Type>
  <DataLen>0</DataLen>
  <Data></Data>
  <Context/>
</entry>
<entry>
  <Name>HackerDefenderDru100</Name>
  <DisplayName>HackerDefenderDru100</DisplayName>
  <ImageName>??\C:\hxdeF100\hxdeFdru.sys</ImageName>
  <Type>10000000</Type>
  <DataLen>0</DataLen>
  <Data></Data>
  <Context/>
</entry>
<entry>
</entry>
<totalFiles>15009</totalFiles>
<totalFiles>15019</totalFiles>
<totalFolders>1190</totalFolders>
<totalFolders>1192</totalFolders>
<readFolders>1190</readFolders>
<readFolders>1192</readFolders>
<file> <fname>c:\ep-in.xml</fname> <md5>00000000000000000000000000000000</md5> <size>59780</siz
<file> <fname>c:\filescan-in.xml</fname> <md5>00000000000000000000000000000000</md5> <size>2609
<file> <fname>c:\filescan-in.xml</fname> <md5>00000000000000000000000000000000</md5> <size>2610
<file> <fname>c:\filescan-out.xml</fname> <md5>00000000000000000000000000000000</md5> <size>261
<file> <fname>c:\Documents and Settings\Username\ntuser.dat.LOG</fname> <md5>000000000000000000000000
<file> <fname>c:\Documents and Settings\Username\ntuser.dat.LOG</fname> <md5>000000000000000000000000
<file> <fname>c:\Documents and Settings\Username\Recent\hxdeF100.ini.lnk</fname> <md5>0000000000000000
<file> <fname>c:\Documents and Settings\Username\Recent\hxdeF100.lnk</fname> <md5>0000000000000000
<file> <fname>c:\hxdeF100\hxdeF100.exe</fname> <md5>00000000000000000000000000000000</md5> <siz
<file> <fname>c:\hxdeF100\hxdeF100.ini</fname> <md5>00000000000000000000000000000000</md5> <siz
<file> <fname>c:\hxdeF100\hxdeFdru.sys</fname> <md5>00000000000000000000000000000000</md5> <siz
<file> <fname>c:\Program Files\CoolWebSearch.info\coolwebsearch-info.dll</fname> <md5>0000000000000000
<file> <fname>c:\Program Files\CoolWebSearch.info\readme.rtf</fname> <md5>000000000000000000000000
<file> <fname>c:\Program Files\CoolWebSearch.info\readme.txt</fname> <md5>000000000000000000000000

```

Figure 14. Hacker Defender-Protected CWS Spyware Exposed by the GhostBuster automated tool: the top portion shows the ASEP diff ; the bottom portion shows the file scan diff.