Digital Investigation (2004) 1, 284—297

ELSEVIER

Digital Investigation

www.elsevier.com/locate/diin

Tool review — remote forensic preservation and examination tools

Eoghan Casey*, Aaron Stanley

Stroz Friedberg LLC, United States

KEYWORDS
Remote digital forensics;
Live digital forensics;
Incident response;
Electronic data discovery;
Computer forensics

Abstract Forensic tools are emerging to help digital investigators preserve evidence on live, remote systems. These tools are applying the precepts of digital forensics to incident response, enterprise policy enforcement, and electronic data discovery. This paper discusses the strengths and shortcomings of ProDiscover IR and EnCase Enterprise Edition in the context of the overall digital investigation process. In addition, a test scenario of a security breach involving a Windows rootkit is used to evaluate the capabilities of these tools. Based on this review, a comparison table is provided and several enhancements are proposed for tools used to process digital evidence on remote, live systems.

> I have spread my dreams under your feet;
> Tread softly because you tread on my dreams.
>
> (Yeats, 1899)

Introduction

Although digital investigators commonly have to obtain evidence from remote live systems, this process controverts best practice guidelines such as the "Good Practices Guide for Computer Based Electronic Evidence" (ACPO, 2003). There are certainly risks when acting on a live system: the operating system may be compromised to hide valuable evidence, and evidence may be inadvertently altered or destroyed during the investigation of the system. There are also risks in collecting evidence remotely: those under investigation may be alerted by investigators activities on the remote system, communications with the remote system may be observed or subverted, and the remote access may introduce a vulnerability that exposes sensitive data. However, data in memory are critical in some cases and it may not be feasible to gain physical access to the computer, particularly in distributed network environments and international investigations. Therefore, digital investigators need tools to obtain this evidence while minimizing the associated risks.

Until recently, when responding to incidents, digital investigators relied primarily on a loose collection of utilities to preserve evidence from

* Corresponding author.
E-mail address: ecasey@strozllc.com (E. Casey).

1742-2876/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.diin.2004.11.003

live systems. For instance, information about processes on a Windows 2000 system could be preserved using a combination of **fport** (www.foundstone.com), **handle** (www.sysinternals.com), **pmdump** (www.ntsecurity.nu/), **cryptcat** (sourceforge.net/projects/cryptcat/), and **md5sum** (www.cygwin.com). Experienced investigators painstakingly compile a virtual toolbox of trusted executables for various operating systems, to be prepared for all eventualities.

From a forensic perspective, using various utilities that are not designed with evidence preservation in mind is risky. Something as basic as altering file access times can hamper a digital investigator's ability to determine what occurred on the system. In one case, a utility crashed and created a "user.dmp" file on the subject system, destroying potentially useful digital evidence. In fact, during this tool review, the **handle** tool repeatedly crashed when run on the compromised test system, possibly due to the presence of a rootkit. Fortunately, this utility provides a warning and option between closing and debugging the program as shown in Fig. 1.

Few investigators have the time or skills to verify that each tool they download from the Internet has not been maliciously modified. Furthermore, even trusted executables in an investigator's toolkit could be undermined by altered libraries or kernel loadable rootkits, resulting in incomplete information about files, processes, network connections, and other items on the subject system. The complexity of this arrangement can make remote live examination incompatible with evidence preservation efforts.

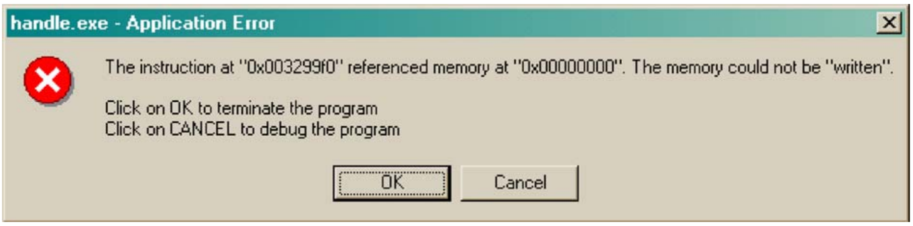
As more organizations realize the importance of properly preserving digital evidence relating to serious incidents involving their IT systems, there is an increasing demand for forensic tools that facilitate the incident response process. Two tools that integrate incident response and computer forensics are compared in this paper: ProDiscover IR 3.5 (PDIR) and EnCase Enterprise Edition 4.19a (EEE). Some features of the upcoming EEE version 5 are also described for completeness of comparison. These tools are specifically designed to preserve digital evidence from remote hosts, while minimizing the changes made on the subject system.

The tools compared in this paper have many applications beyond handling computer security breaches, including investigating fraud and intellectual property theft, and dealing with policy violations such as sexual harassment and employee misuse of IT systems. PDIR is designed to examine one system at a time and is useful for focused investigations involving a small number of computers. Conversely, EEE is designed to integrate with enterprise security architecture, providing enhanced access control and audit functions, and enabling digital investigators to process many systems on a network simultaneously. As a result, EEE is currently the tool of choice for enterprise-wide digital investigations, security audits, and electronic data discovery and subpoena compliance. In addition, EEE extracts more data from memory of remote systems than PDIR, providing investigators with details about processes and network connections that are often useful in digital investigations.

This paper provides a review of technical features for the single purpose of incident response, shortcomings for this purpose are discussed, and the paper closes with a comparison chart. This paper focuses on examining Microsoft Windows systems, but both PDIR and EEE can be used to examine Linux and Solaris hosts.

Test scenario

To test the tools in this paper, the following unauthorized access scenario was created. An intruder gained access to a Windows host named "peeker" with IP address 192.168.0.5 on the target network via VNC. The intruder then installed a rootkit and used this system as a launch pad against other systems on the organization's network. In addition to an internal hard drive with two partitions (C:\ and D:\), this host had a network share (E:\), a USB thumb drive (G:\), and a PGP disk (Z:\).


handle.exe - Application Error

The instruction at "0x003299f0" referenced memory at "0x00000000". The memory could not be "written".

Click on OK to terminate the program
Click on CANCEL to debug the program

Figure 1 Error message and prompt generated when the **handle** utility crashed on the compromised host.

Installing a servlet

To initiate the connections between the examiner machine and the subject computer, a piece of software must be loaded into the memory of the subject computer. Because this program starts a process on the subject computer that listens for outside connections, it is referred to as a "servlet". Sometimes the most challenging part of performing a network-based forensic examination or acquisition is determining how to install the servlet on the subject computer. This section summarizes the available installation methods and their ramifications.

Host-based considerations

The examiner must have Administrator-level access to the subject computer to install the servlet. Both the EEE and PDIR servlets can be run either directly from memory or as an installed service. Installing the servlet as a service keeps the application running all the time, even when the remote system is rebooted. This installation method has the added advantage of permitting connections even when the user is not logged into the remote machine. However, this method places the servlet on the remote system's hard drive and modifies the Registry. This situation can be avoided by storing the servlet somewhere other than on the subject hard drive, and loading it directly into memory. For instance, launching the servlet from removable media will avoid the need to save the executable on the subject hard drive.

The servlets can be deployed either manually while sitting at the computer or via the network if sufficient access methods are available. In a corporate environment, the installation can usually be accomplished via a logon script in the case of Windows or a remote execution in the cases of both Windows and UNIX operating systems. A logon push would require the subject to logout and log back on to their computer. If the examination is an immediate need, this method usually takes too much time.

Many corporate networks utilize some form of desktop management solution that can push the servlet on a computer without the user knowing about it. The servlet can be deployed in much the same way that system security patches are deployed in an enterprise. When such push technology is not available, examiners can use tools like **psexec** (www.sysinternals.com), Dameware (www.dameware.com), Secure Shell (SSH), or even the built-in Windows scheduler (**at**). As long as Administrator-level privileges are obtained, these programs can make the deployment simple.

One word of caution — after the servlet is installed on the subject computer, it could be susceptible to alteration by a clever user. Any time a servlet is installed on a subject computer for an extended period of time, the subject has the ability to replace the servlet program with a trojanized version, or faulty version that reports false data. The authentication architecture of EEE is designed to thwart this type of tampering, using public key cryptography between the servlet and a system called the SAFE (Secure Authentication for EnCase) that must be on the network to authenticate and encrypt all communications between the examiner and subject computer. Although the PDIR servlet does not implement this type of authentication, the servlet is digitally signed with Thawte certificate to enable investigators to verify its integrity. However, with the exception of the EEE Linux servlet, both the EEE and PDIR servlets utilize libraries on a subject computer as a standard part of the operating system and data acquired from the servlet may be altered if those libraries have been tampered with.

With these risks in mind, corporate investigation departments may want to weigh the decision to install a servlet as a standard part of the corporate desktop image.

Network-based considerations

Having a network connection to the subject computer is necessary to examine and acquire data from the remote computer. That network connection, however, must be free from port restrictions and access control mechanisms that might prevent the examiner from connecting to the servlet on the subject computer. Router Access Control Lists, internal firewalls, and personal firewalls can all create barriers that prevent the examiner from connecting to the servlet. In the case of the EEE servlet, it must run on port 4445 of the subject computer. If the computer is running another program that is bound to this port, access will be impossible to obtain. The EEE examiner computer and the subject computer must be able to communicate on port 4445 with the SAFE system to authenticate the connections. While the PDIR servlet can be configured to operate on practically any port, the port must be accessible to the examiner and a personal firewall installed on the subject computer can block this access no matter which port the servlet is configured to operate on.

Review of functionality

To resolve critical incidents in an organization effectively, digital investigators must be able to detect the problem, determine the severity and extent of the damage, and preserve the associated evidence. Two approaches to detecting problems that are relevant to this discussion are: (1) detecting suspicious processes on a host, and (2) triggering actions based on IDS alerts. Both EEE and PDIR can detect a suspicious process on a remote host as detailed in the next section. EEE also can be integrated with Snort to acquire volatile data from a remote host when certain attacks are detected. This feature was not reviewed for this paper but it is worth noting that the upcoming release of EEE version 5 can be integrated with external databases, permitting enhanced IDS integration and incident response.

Memory inspection

In the test scenario, we captured information about volatile data on the target host "peeker" using PDIR and EEE. The "Find Unseen Processes" feature in PDIR detects processes that are hidden by rootkits on Microsoft Windows systems as shown here:

```
Remote system (192.168.0.5)
root.exe          [Unseen Process]
ApntEx.exe        [Seen Process]
Apoint.exe        [Seen Process]
CmLUC.exe         [Seen Process]
PGPtray.exe       [Seen Process]
PcfMgr.exe        [Seen Process]
winvnc4.exe       [Seen Process]
HKServ.exe        [Seen Process]
JogServ2.exe      [Seen Process]
vmware-authd.exe  [Seen Process]
explorer.exe      [Seen Process]
CMD.EXE           [Seen Process]
CSRSS.EXE         [Seen Process]
LSASS.EXE         [Seen Process]
mstask.exe        [Seen Process]
PGPServ.exe       [Seen Process]
regsvc.exe        [Seen Process]
SERVICES.EXE      [Seen Process]
SMSS.EXE          [Seen Process]
spoolsv.exe       [Seen Process]
svchost.exe       [Seen Process]
TASKMGR.EXE       [Seen Process]
vmnat.exe         [Seen Process]
vmnetdhcp.exe     [Seen Process]
WINLOGON.EXE      [Seen Process]
WinMgmt.exe       [Seen Process]
```

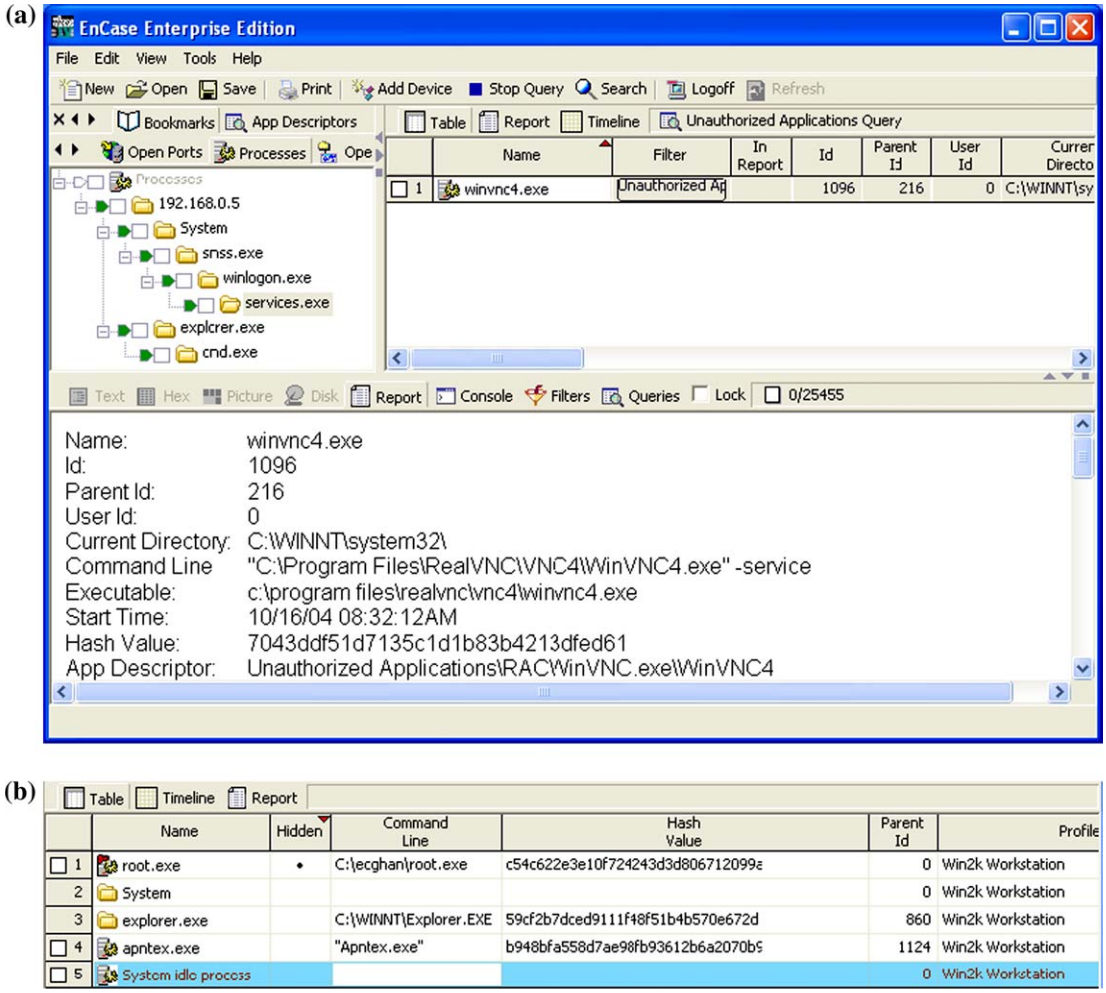
The "root.exe" process at the top of the list is flagged as unseen, indicating that a rootkit may be installed on the remote system. In some situations, the technique that PDIR uses to detect hidden processes modifies the last access times of folders on the remote system but leaves file last accessed times unchanged. PDIR does not provide the path name for the associated executable or files and ports that it has open so further examination is required to locate and identify the rootkit components.

The Snapshot module in EEE captures a list of processes and associated information such as files and ports that each process has opened. Several filters are available in the Snapshot module, including unknown and unauthorized processes. Before EEE can classify processes as unauthorized or malicious the user must create application descriptors (using MD5 hash values of files) and categorize them as unauthorized or malicious. In addition to these general categories, an organization can create profiles of approved processes for different types of systems in their enterprise to facilitate the detection of unapproved processes.

Fig. 2 shows the results of running the Snapshot module on "peeker" and applying the unauthorized processes filter. A portion of the full process tree on the remote host is visible in the upper left pane of Fig. 2(a), and the other panes contain information about the WinVNC process that has been classified as unauthorized. The bottom "Report" pane includes the path, command line, and start time of the process, and this information can also be viewed by scrolling to the right in the upper right "Table" pane. Fig. 2(b) shows the hidden "root.exe" process detected by EEE version 5. The technique used by EEE version 5 to detect hidden processes does not alter the file system of the remote system, and shows the full path of the executable "C:\eoghan\root.exe" and any associated command line options are also provided.

The EEE Snapshot also captures information about network connections and ports. For instance, Fig. 3 shows ESTABLISHED network connections on "peeker," including the intruder's VNC connection on port 5900, the EEE connection on port 4445, and what appears to be a NetBIOS session with the file server in the test scenario (192.168.0.2). The NetBIOS session indicates that another system may have been breached but further investigation is required to determine if anything sensitive or valuable on the second system was exposed.

As another example, Fig. 4 shows the "Suspicious Ports" filter being applied to an EEE Snapshot


(a)
EnCase Enterprise Edition
File Edit View Tools Help
New Open Save Print Add Device Stop Query Search Logoff Refresh
Bookmarks App Descriptors
Open Ports Processes Ope
Processes
192.168.0.5
System
snss.exe
winlogon.exe
services.exe
explcrer.exe
cnd.exe
Table Report Timeline Unauthorized Applications Query
Text Hex Picture Disk Report Console Filters Queries Lock 0/25455
(b)
Table Timeline Report

	Name	Filter	In Report	Id	Parent Id	User Id	Currer Directo
□ 1	winvnc4.exe	Unauthorized Ap		1096	216	0	C:\WINNT\sy

Name: winvnc4.exe
Id: 1096
Parent Id: 216
User Id: 0
Current Directory: C:\WINNT\system32\
Command Line "C:\Program Files\RealVNC\VNC4\WinVNC4.exe" -service
Executable: c:\program files\realvnc\vnc4\winvnc4.exe
Start Time: 10/16/04 08:32:12AM
Hash Value: 7043ddf51d7135c1d1b83b4213dfed61
App Descriptor: Unauthorized Applications\RACWinVNC.exe\WinVNC4

	Name	Hidden	Command Line	Hash Value	Parent Id	Profile
□ 1	root.exe	•	C:\ecghan\root.exe	c54c622e3e10f724243d3d806712099a	0	Win2k Workstation
2	System				0	Win2k Workstation
3	explorer.exe		C:\WINNT\Explorer.EXE	59cf2b7dced9111f48f51b4b570e672d	860	Win2k Workstation
□ 4	apntex.exe		"Apntex.exe"	b948bfa558d7ae98fb93612b6a2070b9	1124	Win2k Workstation
□ 5	System idle process				0	Win2k Workstation

Figure 2 (a) The results of running the EEE Snapshot module and applying the suspicious processes filter. (b) The "root.exe" process detected using EEE is flagged as Hidden and has a red mark on the top left of its associated icon.

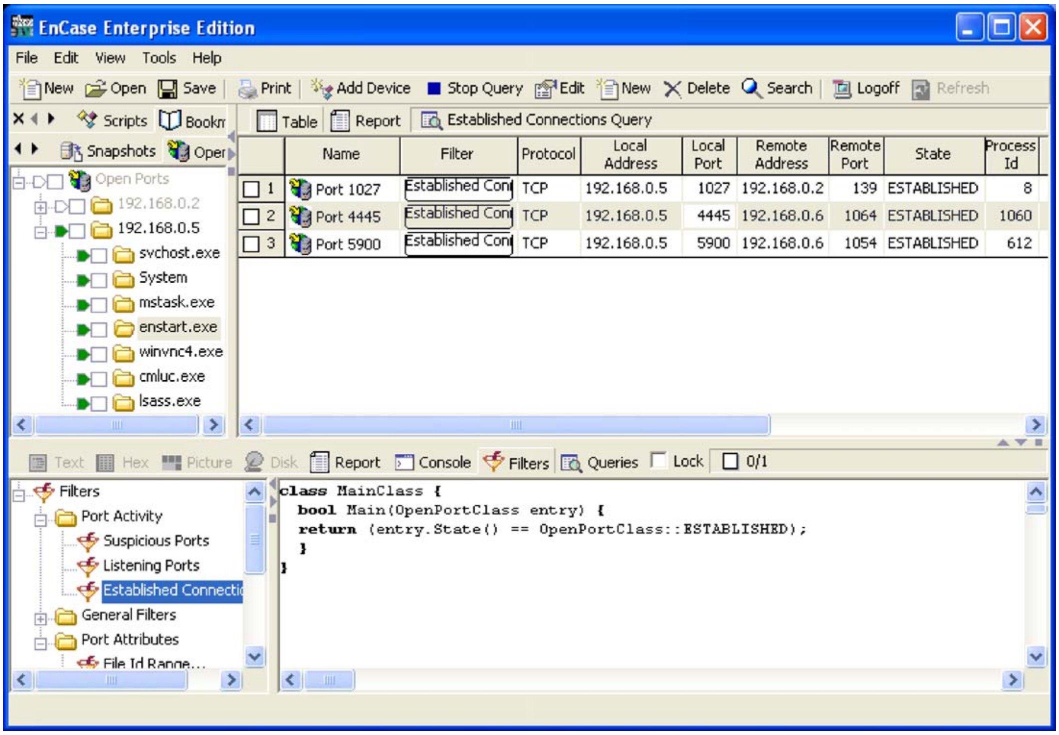
of a Linux system running the "t0rnkit" rootkit with the backdoor SSH server configured to listen on port 31337. Currently, neither EEE nor PDIR have the ability to detect hidden processes on Linux or Solaris systems.

Using these tool features, organizations can check all systems on their network periodically for signs of intrusion or misuse. The EEE Snapshot module can be configured to capture volatile data from multiple systems in one sweep whereas PDIR requires the user to query each system individually. The additional information about processes that is presented in EEE enables examiners to combine process information from multiple systems to create a timeline, find identical processes on multiple systems, or discern other patterns such as the sequence of events in an attack. The Snapshot module in EEE also preserves information about user accounts on remote hosts with associated filters. This can be useful for determining who was logged into a given host at a particular time or which machines were accessed using a given account.

Storage media examination

Once a digital investigator finds a system that deserves further attention, he/she generally wants to examine the disk contents to determine the severity of the incident. For instance, he/she may want to perform a keyword search to determine if any sensitive data were exposed (e.g., credit card numbers, intellectual property, medical information). Both PDIR and EEE can preview a remote system, providing access to physical disks and logical volumes on those disks. In addition, EnCase provides access to RAM disks such as the PGP disk shown in Fig. 5. Mounted network drives are not detected by either tool.

Both PDIR and EEE can interpret FAT, NTFS, ext2/ext3, and UFS file systems, enabling examination



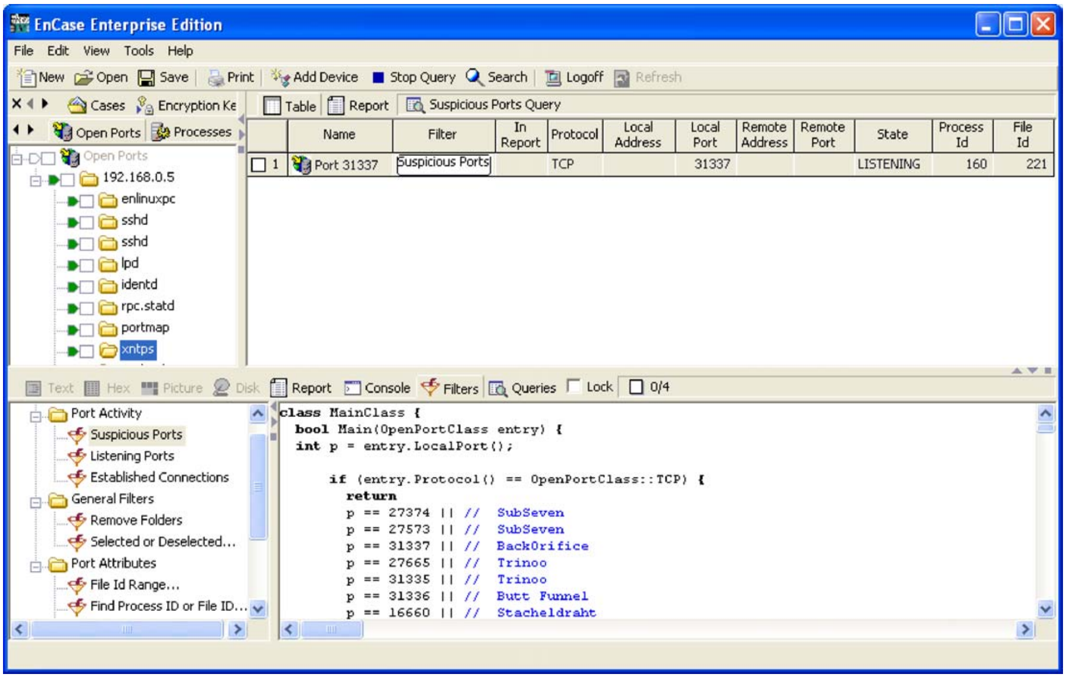
	Name	Filter	Protocol	Local Address	Local Port	Remote Address	Remote Port	State	Process Id
□ 1	Port 1027	Established Conn	TCP	192.168.0.5	1027	192.168.0.2	139	ESTABLISHED	8
□ 2	Port 4445	Established Conn	TCP	192.168.0.5	4445	192.168.0.6	1064	ESTABLISHED	1060
□ 3	Port 5900	Established Conn	TCP	192.168.0.5	5900	192.168.0.6	1054	ESTABLISHED	612

```
class MainClass {
  bool Main(OpenPortClass entry) {
  return (entry.State() == OpenPortClass::ESTABLISHED);
  }
}
```

Figure 3 EEE Snapshot of volatile data from "peeker" filtered to show established network connections.

of active and deleted files, as well as slack and unallocated space as shown in Fig. 6.

Both tools can perform keyword searches or MD5 hash comparisons on a remote system. A hash comparison can be used to exclude known good files from a search or to detect known bad files on the disk. For instance, the AFX rootkit which includes the hidden "root.exe" process detected

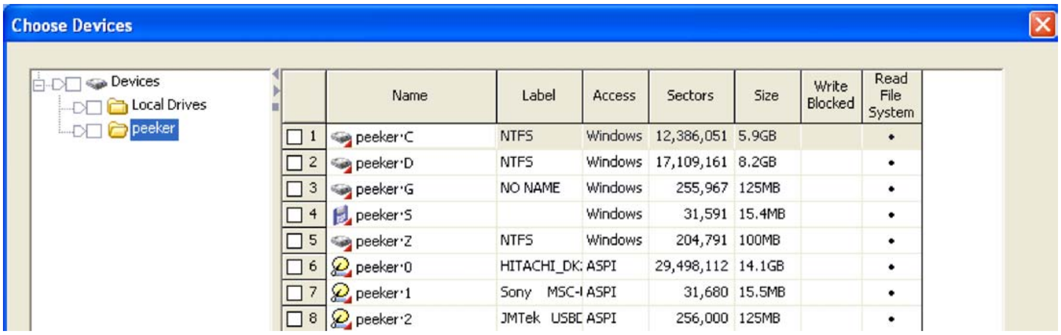

EnCase Enterprise Edition
File Edit View Tools Help
New Open Save Print Add Device Stop Query Search Logoff Refresh
Cases Encryption Ke
Open Ports Processes
Table Report Suspicious Ports Query
Open Ports
192.168.0.5
enlinuxpc
sshd
sshd
lpd
identd
rpc.statd
portmap
xntps
Text Hex Picture Disk Report Console Filters Queries Lock 0/4
Port Activity
Suspicious Ports
Listening Ports
Established Connections
General Filters
Remove Folders
Selected or Deselected...
Port Attributes
File Id Range...
Find Process ID or File ID...

	Name	Filter	In Report	Protocol	Local Address	Local Port	Remote Address	Remote Port	State	Process Id	File Id
□ 1	Port 31337	Suspicious Ports		TCP		31337			LISTENING	160	221

```
class MainClass {
  bool Main(OpenPortClass entry) {
  int p = entry.LocalPort();

      if (entry.Protocol() == OpenPortClass::TCP) {
        return
        p == 27374 || //  SubSeven
        p == 27573 || //  SubSeven
        p == 31337 || //  BackOrifice
        p == 27665 || //  Trinoo
        p == 31335 || //  Trinoo
        p == 31336 || //  Butt Funnel
        p == 16660 || //  Stacheldraht
```

Figure 4 EEE Snapshot of volatile data from a compromised Linux host filtered to show malicious ports.



	Name	Label	Access	Sectors	Size	Write Blocked	Read File System
□ 1	peeker·C	NTFS	Windows	12,386,051	5.9GB		•
□ 2	peeker·D	NTFS	Windows	17,109,161	8.2GB		•
□ 3	peeker·G	NO NAME	Windows	255,967	125MB		•
□ 4	peeker·S		Windows	31,591	15.4MB		•
□ 5	peeker·Z	NTFS	Windows	204,791	100MB		•
□ 6	peeker·0	HITACHI_DK:	ASPI	29,498,112	14.1GB		•
□ 7	peeker·1	Sony MSC-I	ASPI	31,680	15.5MB		•
□ 8	peeker·2	JMTek USBC	ASPI	256,000	125MB		•

Figure 5 Acquiring a PGP disk (Z:\) on a remote system using EnCase.

earlier, was identified on "peeker" using both PDIR and EEE by searching for the MD5 values of files in this rootkit. The hidden directory named "eoghan" was visible in both PDIR and EEE because they process the raw file tables without relying on the remote operating system. The "Find Suspect Files" feature in PDIR found the AFX rootkit files based on MD5 values specified in an input Hash Set, and included the following results in the report:

Evidence of interest:

```
Total Evidence Items of Interest: 2

\\192.168.0.5\PhysicalDrive0, Hard Disk C:
   List of Files:

\\192.168.0.5\PhysicalDrive0\C:\eoghan\hook.
dll
MD5 Checksum: B0F9E41EF7C0B5D1EFA8FAC854C4DAFE
Created: 10/07/2004 21:05 Modified: 10/16/
2004 11:50 Last Accessed: 10/16/2004 20:56

\\192.168.0.5\PhysicalDrive0\C:\eoghan\root.
exe
MD5 Checksum: C54C622E3E10F724243D3D806712099A
Created: 09/22/2004 03:11 Modified: 09/22/
2004 03:11 Last Accessed: 10/16/2004 17:59
   \\192.168.0.5\PhysicalDrive0, Hard Disk
   C:: Evidence of Interest: 2
```

The same approach of using Hash Sets was implemented in EEE to detect the rootkit files as shown in Fig. 7.

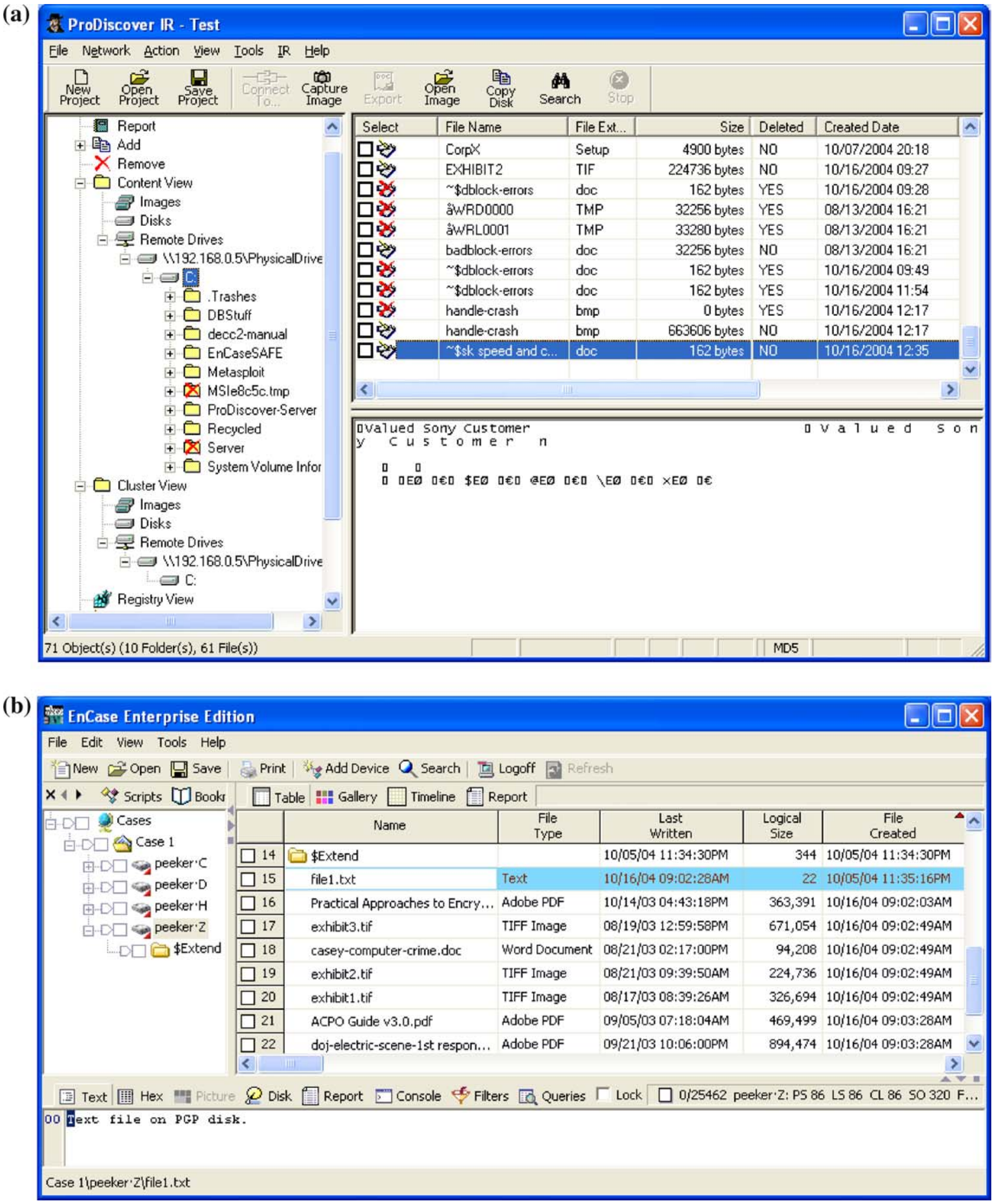
An added feature in PDIR called "Find Unseen Files" will flag files that are hidden by rootkits on Windows system.

Once a file of interest is located on the remote system, PDIR and EEE can copy the contents to the investigator's system. Logical files can be copied from the remote host onto the examination system even if a file is open and in use on the remote system. If a file is modified while a remote live system is being examined, EEE and PDIR can only copy the altered data after the remote operating system commits the changes to disk. Because of file system caching, file modifications may not be written to disk for extended periods, causing PDIR and EEE to copy the original data associated with files that are being edited.

When an incident involves multiple systems, it can be useful to combine information from all systems in a single view to create a timeline, find identical files on multiple systems, or discern other patterns. As an example, Fig. 8 shows files from three remote hosts combined and sorted by their creation time. In this scenario, at 10:30 PM the AFX rootkit was uploaded on to 192.168.0.2, at 10:35 PM two files were copied from 192.168.0.2 to 192.168.0.5, at 10:37 PM the IIS Web server on 192.168.0.8 was compromised from 192.168.0.5, and at 11:02 PM "psexec.exe" and "nc.exe" were uploaded on to the Web server.

This type of correlation cannot be performed using PDIR because it only connects to one remote host at a time. In addition, PDIR does not have the ability to combine file listings from multiple images, so each one must be examined independently.

If digital investigators decide that it is necessary to preserve the entire contents of a hard drive or partition of a remote host, PDIR and EEE can acquire a forensic image over the network. Because things can change during a sector by sector copy of an active hard drive, the MD5 hash value of the drive prior to the acquisition may differ from the MD5 value of the forensic image. Therefore, the concept of a forensic image of live systems differs from that of a dead system. A "live" forensic image can only be verified against itself whereas a "dead" image can be verified against the original media. When processing a live system, it is useful to know if large amounts of data are being added or destroyed. Neither PDIR



Select	File Name	File Ext...	Size	Deleted	Created Date
□	CorpX	Setup	4900 bytes	NO	10/07/2004 20:18
□	EXHIBIT2	TIF	224736 bytes	NO	10/16/2004 09:27
□	~$dblock-errors	doc	162 bytes	YES	10/16/2004 09:28
□	åWRD0000	TMP	32256 bytes	YES	08/13/2004 16:21
□	åWRL0001	TMP	33280 bytes	YES	08/13/2004 16:21
□	badblock-errors	doc	32256 bytes	NO	08/13/2004 16:21
□	~$dblock-errors	doc	162 bytes	YES	10/16/2004 09:49
□	~$dblock-errors	doc	162 bytes	YES	10/16/2004 11:54
□	handle-crash	bmp	0 bytes	YES	10/16/2004 12:17
□	handle-crash	bmp	663606 bytes	NO	10/16/2004 12:17
□	~$sk speed and c...	doc	162 bytes	NO	10/16/2004 12:35

```
▯Valued Sony Customer                             ▯ V a l u e d   S o n
y   C u s t o m e r   n

   ▯   ▯
   ▯ ▯E∅ ▯€▯ $E∅ ▯€▯ @E∅ ▯€▯ \E∅ ▯€▯ ×E∅ ▯€
```

71 Object(s) (10 Folder(s), 61 File(s))

	Name	File Type	Last Written	Logical Size	File Created
□ 14	$Extend		10/05/04 11:34:30PM	344	10/05/04 11:34:30PM
□ 15	file1.txt	Text	10/16/04 09:02:28AM	22	10/05/04 11:35:16PM
□ 16	Practical Approaches to Encry...	Adobe PDF	10/14/03 04:43:18PM	363,391	10/16/04 09:02:03AM
□ 17	exhibit3.tif	TIFF Image	08/19/03 12:59:58PM	671,054	10/16/04 09:02:49AM
□ 18	casey-computer-crime.doc	Word Document	08/21/03 02:17:00PM	94,208	10/16/04 09:02:49AM
□ 19	exhibit2.tif	TIFF Image	08/21/03 09:39:50AM	224,736	10/16/04 09:02:49AM
□ 20	exhibit1.tif	TIFF Image	08/17/03 08:39:26AM	326,694	10/16/04 09:02:49AM
□ 21	ACPO Guide v3.0.pdf	Adobe PDF	09/05/03 07:18:04AM	469,499	10/16/04 09:03:28AM
□ 22	doj-electric-scene-1st respon...	Adobe PDF	09/21/03 10:06:00PM	894,474	10/16/04 09:03:28AM

□ 0/25462 peeker·Z: PS 86 LS 86 CL 86 SO 320 F...

```
00 Text file on PGP disk.
```

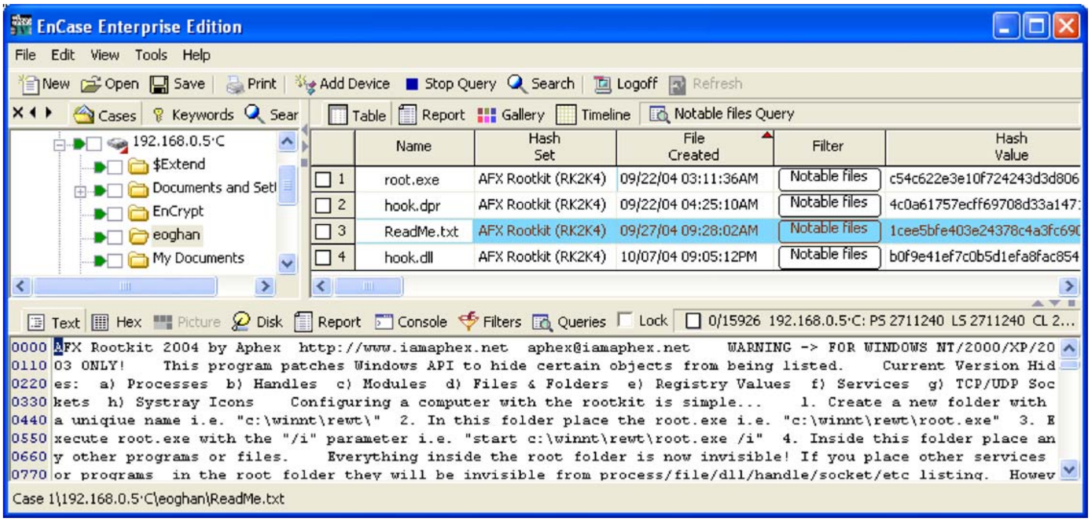
Case 1\peeker·Z\file1.txt

Figure 6 (a) Preview of remote file system on USB thumb drive using ProDiscover IR. (b) Preview of remote file system on PGPdisk using EnCase.

nor EEE give clear information about what is changing on the remote system while it is being acquired or examined.

In remote preview and image capture modes, both PDIR and EEE did not alter the metadata of files on the subject system, and preserved the metadata when copying the file to the client computer. To test this, a USB flash drive was created with various files having different create, modified, and accessed times. The flash drive was then imaged to be restored after each test run. PDIR was used to load the flash drive remotely and open and copy some of the documents on it. The metadata were not changed on the flash drive and remained consistent on the client system as well. This process was repeated with EnCase Enterprise Edition with the same results.

Both PDIR and EEE have other features for examining storage media that are beyond the scope of this review. For instance, the EEE


EnCase Enterprise Edition
File Edit View Tools Help
New Open Save Print Add Device Stop Query Search Logoff Refresh
Cases Keywords Sear
Table Report Gallery Timeline Notable files Query
192.168.0.5·C
$Extend
Documents and Setl
EnCrypt
eoghan
My Documents
Text Hex Picture Disk Report Console Filters Queries Lock

	Name	Hash Set	File Created	Filter	Hash Value
□ 1	root.exe	AFX Rootkit (RK2K4)	09/22/04 03:11:36AM	Notable files	c54c622e3e10f724243d3d806
□ 2	hook.dpr	AFX Rootkit (RK2K4)	09/22/04 04:25:10AM	Notable files	4c0a61757ecff69708d33a147
□ 3	ReadMe.txt	AFX Rootkit (RK2K4)	09/27/04 09:28:02AM	Notable files	1cee5bfe403e24378c4a3fc69
□ 4	hook.dll	AFX Rootkit (RK2K4)	10/07/04 09:05:12PM	Notable files	b0f9e41ef7c0b5d1efa8fac854

□ 0/15926 192.168.0.5·C: PS 2711240 LS 2711240 CL 2...

```
0000 AFX Rootkit 2004 by Aphex  http://www.iamaphex.net  aphex@iamaphex.net    WARNING -> FOR WINDOWS NT/2000/XP/20
0110 03 ONLY!    This program patches Windows API to hide certain objects from being listed.    Current Version Hid
0220 es:  a) Processes  b) Handles  c) Modules  d) Files & Folders  e) Registry Values  f) Services  g) TCP/UDP Soc
0330 kets  h) Systray Icons    Configuring a computer with the rootkit is simple...    1. Create a new folder with
0440 a uniqiue name i.e. "c:\winnt\rewt\"  2. In this folder place the root.exe i.e. "c:\winnt\rewt\root.exe"  3. E
0550 xecute root.exe with the "/i" parameter i.e. "start c:\winnt\rewt\root.exe /i"  4. Inside this folder place an
0660 y other programs or files.   Everything inside the root folder is now invisible! If you place other services
0770 or programs  in the root folder they will be invisible from process/file/dll/handle/socket/etc listing.  Howev
```

Case 1\192.168.0.5·C\eoghan\ReadMe.txt

Figure 7 EEE using Hash Sets to detect AFX rootkit.

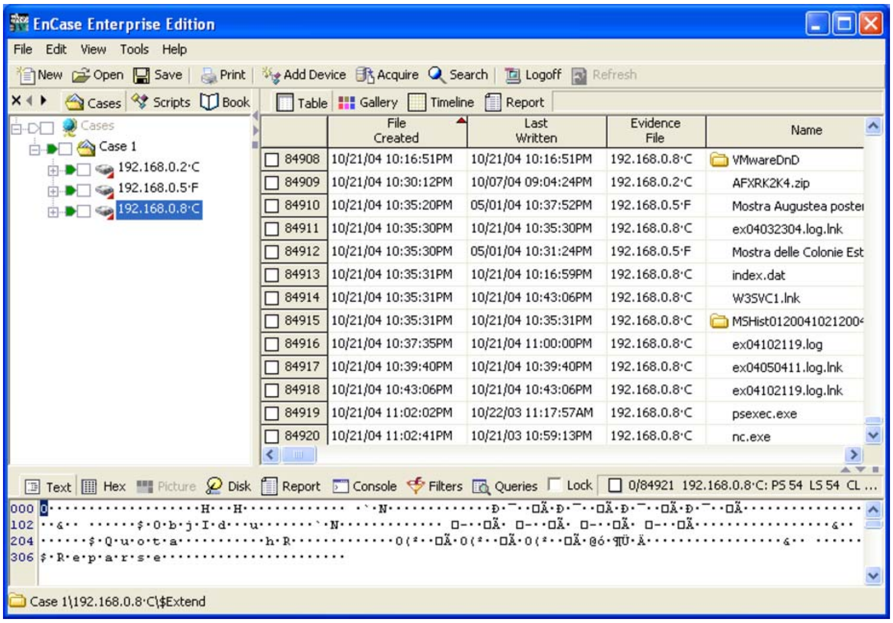
decryption module provides various methods for extracting data encrypted using the NTFS Encrypted File System (EFS), and PDIR can interpret metadata within EXIF files.

Investigative considerations

There are several important considerations when dealing with digital evidence on remote live systems: the original evidence should be accessed in read-only mode, the remote operating system should not be trusted, forensic tools should not overload the remote system, and evidence and the remote system should be protected from unauthorized access. This section reviews EEE and PDIR in the context of these issues.

Treading softly — investigative considerations

Generally, both EEE and PDIR do not alter data on the remote system when performing operations on the subject system via servlet. Although the EEE and PDIR servlets occupy memory on the remote host, they appear to be designed not to use swap space, thus minimizing the changes made to potential digital evidence on the remote system.


EnCase Enterprise Edition
File Edit View Tools Help
New Open Save Print Add Device Acquire Search Logoff Refresh
Cases Scripts Book
Table Gallery Timeline Report
Cases
Case 1
192.168.0.2·C
192.168.0.5·F
192.168.0.8·C
Text Hex Picture Disk Report Console Filters Queries Lock

	File Created	Last Written	Evidence File	Name
□ 84908	10/21/04 10:16:51PM	10/21/04 10:16:51PM	192.168.0.8·C	VMwareDnD
□ 84909	10/21/04 10:30:12PM	10/07/04 09:04:24PM	192.168.0.2·C	AFXRK2K4.zip
□ 84910	10/21/04 10:35:20PM	05/01/04 10:37:52PM	192.168.0.5·F	Mostra Augustea poster
□ 84911	10/21/04 10:35:30PM	10/21/04 10:35:30PM	192.168.0.8·C	ex04032304.log.lnk
□ 84912	10/21/04 10:35:30PM	05/01/04 10:31:24PM	192.168.0.5·F	Mostra delle Colonie Est
□ 84913	10/21/04 10:35:31PM	10/21/04 10:16:59PM	192.168.0.8·C	index.dat
□ 84914	10/21/04 10:35:31PM	10/21/04 10:43:06PM	192.168.0.8·C	W3SVC1.lnk
□ 84915	10/21/04 10:35:31PM	10/21/04 10:35:31PM	192.168.0.8·C	MSHist0120041021200
□ 84916	10/21/04 10:37:35PM	10/21/04 11:00:00PM	192.168.0.8·C	ex04102119.log
□ 84917	10/21/04 10:39:40PM	10/21/04 10:39:40PM	192.168.0.8·C	ex04050411.log.lnk
□ 84918	10/21/04 10:43:06PM	10/21/04 10:43:06PM	192.168.0.8·C	ex04102119.log.lnk
□ 84919	10/21/04 11:02:02PM	10/22/03 11:17:57AM	192.168.0.8·C	psexec.exe
□ 84920	10/21/04 11:02:41PM	10/21/03 10:59:13PM	192.168.0.8·C	nc.exe

□ 0/84921 192.168.0.8·C: PS 54 LS 54 CL ...

```
000 0··················H···H············  ·`·N············Đ·¯··□Ă·Đ·¯··□Ă·Đ·¯··□Ă·Đ·¯··□Ă················
102 ··&··  ······$·O·b·j·I·d···u·······`·N············  □–··□Ă·  □–··□Ă·  □–··□Ă·  □–··□Ă·················&··
204 ······$·Q·u·o·t·a···········h·R·············O(²··□Ă·O(²··□Ă·O(²··□Ă·@ó·¶Ü·Ä·················&··  ······
306 $·R·e·p·a·r·s·e·························
```

Case 1\192.168.0.8·C\$Extend

Figure 8 Combining data from multiple remote systems in a single view using EEE.

However, as mentioned earlier, PDIR changes last accessed date|time stamps of folders when performing the "Find Unseen Processes," and "Find Unseen Files" operations. Also keep in mind that when the EEE and PDIR servlets are installed on a system it will make changes to the Registry and disk. Therefore, some organizations install servlets on important systems as part of normal operations, enabling them to monitor these systems periodically and respond to critical incidents more effectively.

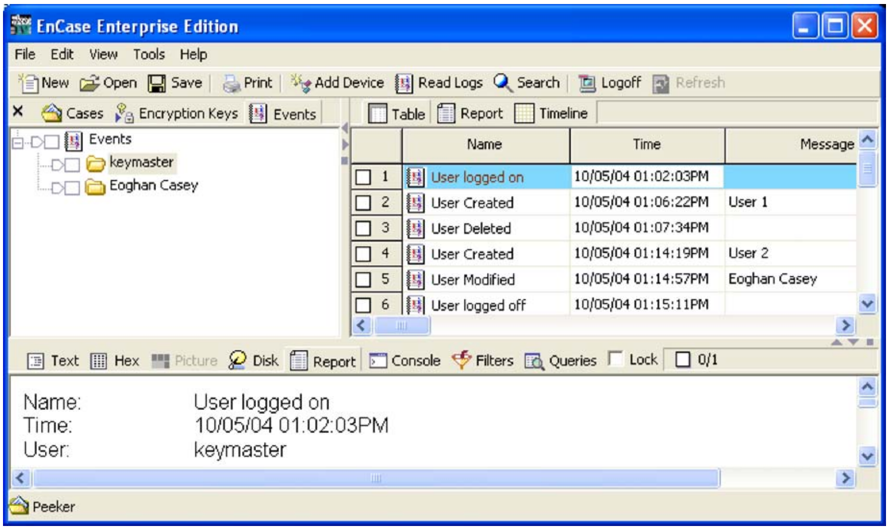
Recall from the introduction that even trusted executables in an investigator's toolkit can be undermined by altered libraries or kernel loadable rootkits, resulting in incomplete information about files, processes, network connections, and other items on the subject system. The EEE and PDIR servlets on Windows, Linux, and Solaris require certain libraries on the subject system. To mitigate the associated risks, it is advisable to bring trusted copies of these libraries to the subject system along with the servlet.

Security

When responding to an incident, digital investigators should assume that the network they are using cannot be trusted. A user or intruder may have full control of the system under investigation, and may be monitoring system activities and network traffic to determine if he/she is being investigated. A malicious offender may even take evasive action to disrupt investigators. Therefore, all communications should be encrypted, and all tools used for remote evidence processing should not introduce vulnerabilities on subject systems.

The PDIR online documentation recommends that the servlet only be run when needed to mitigate the security risks of leaving it running for extended periods of time. PDIR has optional encryption and password protection that are not enabled by default. When the servlet is installed on the remote host as a service, it can be configured with a password, and encryption can be enabled only after a connection is established. Although encryption cannot be enabled prior to authenticating with the PDIR servlet, the password is not transmitted in plain text and therefore is not plainly visible in network traffic. However, because the password is provided as an argument to the PDIR servlet, it can be obtained by users who can view the Registry or process details of the subject system. There does not appear to be a limit to the number of password attempts that the servlet will accept, and anyone with PDIR can attempt to guess the password of the servlet. However, PDIR uses Global Unique Identifiers to restrict a servlet to one client per session and to prevent tampering with the network communications.

EEE uses a dedicated system called the SAFE to manage security. The SAFE protocol uses a combination of public, private, and session keys to ensure that all connections to remote servlets are authorized and encrypted. Attempts to connect to a servlet in another EEE implementation were unsuccessful. In addition, separate user accounts can be created for different roles, and the SAFE enforces access permissions and


EnCase Enterprise Edition
File Edit View Tools Help
New Open Save Print Add Device Read Logs Search Logoff Refresh
Cases Encryption Keys Events
Table Report Timeline
Events
keymaster
Eoghan Casey
Text Hex Picture Disk Report Console Filters Queries Lock 0/1
Peeker

	Name	Time	Message
□ 1	User logged on	10/05/04 01:02:03PM	
□ 2	User Created	10/05/04 01:06:22PM	User 1
□ 3	User Deleted	10/05/04 01:07:34PM	
□ 4	User Created	10/05/04 01:14:19PM	User 2
□ 5	User Modified	10/05/04 01:14:57PM	Eoghan Casey
□ 6	User logged off	10/05/04 01:15:11PM	

Name: User logged on
Time: 10/05/04 01:02:03PM
User: keymaster

Figure 9 Screenshot of audit trail.

maintains an audit trail for these accounts (see Fig. 9). Although this configuration makes the installation of EEE more complicated, it results in a more secure environment. In addition to restricting who has remote access to certain hosts, there is a high degree of control over what each account can do on each system. This control is advantageous in an enterprise that separates roles. For instance, some organizations do not permit incident responders to view the file system of any hosts, only acquire the evidence and pass the images to forensic examiners.

Although the clients of both EEE and PDIR accept connections from the network, attempts to crash them by connecting to their listening port were unsuccessful. However, the EEE client and servlet disconnect unauthorized connections immediately whereas the PDIR client and servlet do not, potentially making the latter more susceptible to denial of service attacks.

Performance

The impact that the remote examination process has on performance of the subject system and network may be a concern for some users. If the servlet utilizes all available resources on the subject system, this may be unacceptable from a business continuity standpoint if it renders a critical server unusable. This situation may also be undesirable from an investigative standpoint because it could alert users on the system. Similarly, if the remote forensic tool makes the network sluggish, this may be an unacceptable hindrance to the organization or investigation.

Both EEE and PDIR exhibited different network behaviors during the pre-acquisition and acquisition phases. While previewing a remote computer, PDIR was measured to use an average of 340 kb/s of network bandwidth. This led to a time measurement of 4 min to open a 2 MB bitmap file stored on a USB drive. Because EEE employs a process whereby it reads the disk configuration and file structure of the subject system prior to allowing the examiner to preview the data, the same file opened almost immediately during the EEE test, but EEE did take a few minutes to read the device at first. When reading the device initially, the EEE client was measured to use approximately 5.2 MB/s of network bandwidth. After the initial reading in, browsing through the subject system had no noticeable impact on network usage. Because EEE relies on the SAFE, which stores the encryption and decryption keys for the examiner and subject system to authenticate with each other, there is constant communication between the examiner, subject computer, and SAFE machine. In testing, the SAFE's network utilization never exceeded approximately 50 kb/s.

PDIR, in acquisition mode, was measured to use a maximum of 5.5 MB/s of network bandwidth during an acquisition. Acquisitions using EEE took longer than PDIR when the initial time spent reading the device is taken into account. During the acquisition, however, the network usage dropped to 3.5 MB/s.

To test whether an intruder who was controlling a subject system via VNC would have any idea that the system was being profiled, monitored, or acquired, a VNC connection was maintained and monitored at points during the bandwidth testing. With VNC connected, the overall network usage of the subject system fluctuated between 3.5 MB/s and 4.8 MB/s. The only noticeable slow down from the perspective of the intruder would be during times when the examiner was acquiring or reading data from the hard drive of the subject system. If the intruder is also running a disk-intensive process, the competition for the hard drive resources will limit the speed at which the intruder can read or write. This was exhibited by using Eraser to delete a 30 MB file by performing a single-pass overwrite. Without any other disk activity, the erase procedure took 0.81 s. With EEE performing an acquisition of the drive, that time increased to 1.08 s, a 33% increase.

Keyword searching, hashing and other processes such as "Find Unseen Files" cause the CPU load on remote system to increase for as long as the search takes. PDIR has the option to search content (logical) or clusters (physical). Cluster searches of remote hosts did not have a significant impact on the performance of the remote system but content searches were more CPU intensive, possibly because file system structure must be taken into account when performing a logical search. EEE performs both logical and physical searches simultaneously but can be configured to just search logical files. Overall, EEE completed keyword searches more quickly than PDIR but placed more CPU load on the remote system. "EEE also obtains information about processes more efficiently because it extracts data directly from memory of the remote system, whereas PDIR parses the remote file system and compares the results with information available through the remote operating system, placing more load on the remote system." Neither tool has a rate limiting feature to enable the user to control how much load is placed on the remote system.

Arguably, a remote forensic servlet should not be visible to a user or intruder on the subject system. The EEE and PDIR servlets do not use sophisticated concealment techniques and an alert offender may detect their presence. To evaluate the overall stealthiness of the servlet applications, the Windows task manager and the **netstat** command were employed to monitor process and network port activity during the acquisitions. The EEE servlet can be run from the command line on the subject system, or can be installed as a service that runs at the System level. Running the service at the System level makes it more difficult for users of the subject system to terminate the process.

The PDIR servlet can be run stand-alone without installation and has a menu option to switch to "Stealth Mode". This stealth mode simply hides the application window that the PDIR servlet initially opens, using the same process name and port. When installed as a service, the PDIR servlet runs at the Administrator level on the subject system. This allows a user with administrative privileges on the system to end the process manually, thus terminating the examiner's connection. In addition, PDIR opens up a large number of ports on the subject system, most of them that sit in TIME_WAIT status, which could alert an intruder or user on the remote system.

Limitations

This section summarizes the main limitations of PDIR and EEE. Both tools require Administrator-level privileges on the subject computer, which may prevent their use in some investigations. Firewalls can also block communication between client and servlets, preventing investigators from utilizing these tools in some situations.

Investigators cannot access certain information from remote systems using EEE and PDIR. For instance, neither tool can acquire process memory or view data on mounted network shares, limiting the amount of information that can be obtained from the system remotely. Neither tool gives an indication of how much data are changing on the remote system while they are being examined or acquired. Although the EEE Snapshot module gives information about which files are open, this information is not available when viewing the associated file system, making it more difficult to determine if any documents are being edited. In addition, PDIR does not detect mounted RAM disks (e.g., PGP disks).

Although both tools make an effort to minimize changes to the original evidence, running a process in memory or installing a service necessarily alters the system. These minor changes are a necessary tradeoff when examining a live system but every effort must be employed to prevent such changes. Because of the Windows API behavior on some systems, the PDIR "Find Hidden Process" and "Find Hidden Files" functions may change last accessed times of folders. Although these operations do not alter file accessed times, the modification of folder accessed times may be detrimental in some investigations. It is also important not to trust the remote operating system. With the exception of the EEE Linux servlet, the servlets in both tools rely on libraries on the remote operating system, which could be altered to hide data or provide false information.

While PDIR and EEE are designed to detect hidden information on computers, they do not employ advanced concealment techniques. Servlets are not hidden from users on the system being examined and can be killed by users in some circumstances. An added limitation that applies to all remote forensic examination tools is that offenders may modify DNS or change their IP address, causing investigators to connect to the wrong system.

Summary

Both ProDiscover IR and EnCase Enterprise Edition are bridging the gap between computer forensics and incident response. Although both tools emphasize preservation of digital evidence, each tool has a different design philosophy and technical implementation. PDIR is designed for examining a small number of systems involved in an incident whereas EEE is designed to integrate with enterprise security architecture and examine a large number of systems simultaneously. PDIR only presents data that are verifiably complete and accurate to a high degree of confidence whereas EEE provides the most information possible, including information provided by the subject system (e.g., process and network connection details). There are exceptions such as PDIR changing folder last access times under some conditions but this is distinctly labeled as an incident response operation as opposed to a forensic function. Similarly, EEE has an add-on remediation module to remove malicious programs but this capability is clearly distinguished as an incident response function.

Table 1 summarizes aspects of PDIR and EEE discussed in this review.

Digital investigators will continue to encounter situations that are not suitable for these integrated

Table 1

Feature	EnCase Enterprise Edition	ProDiscover IR
Integrated graphical user interface	✓	✓
Servlet		
Ability to run servlet in memory from read-only media	✓	✓
Ability to install servlet as a service	✓	✓
Configure servlet to listen on alternate port	x	✓
Hide servlet from users of the remote system	x	x
Authentication between client and servlet	PKI	Password (optional)
Encrypted communication between client and servlet	128-bit AES	128-bit AES / 256-bit Twofish
Detect malicious alterations to servlet	Client only connects to servlets approved by PKI	Manually verify digital signature
Servlet does not rely on local libraries of remote system	Linux servlet is statically compiled	x
Storage Media and File Systems		
FAT/NTFS	✓	✓
EXT2/EXT3	✓	✓
UFS	✓	✓
View information about file systems on remote computers without altering file system metadata	✓	✓
Identify malicious or known files on remote system using MD5 hash values	✓	✓
Copy individual files or folders while preserving metadata	✓	✓
Indicate in file system view which files are open for editing	x	x
Capture forensic image of remote computers	✓	✓
Memory Inspection		
View information about processes on remote computers	✓	✓
Obtain process details without altering remote file system	✓	x
Reveal hidden processes on remote system	Windows only	Windows only
Provide executable paths for processes on remote computers	✓	x
List open files on remote computers	✓	x
Acquire memory of processes on remote system	x	x
Provide information about network connections on remote systems	✓	x
Miscellaneous		
View mounted RAM disks (e.g., PGP Disk)	✓	x
View mounted network shares	x	x
Combine and correlate data from multiple remote systems	✓	x
Integration with Snort intrusion detection system	✓	x
User options to control the load placed on remote system during preview, acquisition, search, or other operations	x	x

remote forensic tools. For instance, when investigators do not have Administrator privileges, or when dealing with Macintosh systems it may not be possible to install a servlet on the computer. Therefore, it is necessary to remain conversant with the various utilities that exist for examining systems as described in the beginning of this paper.

Acknowledgements

EnCase Enterprise Edition: Brian Karney, Rich Cummings, and Michael J. Staggs, Guidance Software (http://www.encase.com).

ProDiscover IR: Christopher Brown and Ted Augustine, Technology Pathways (http://www.techpathways.com).

Bibliography

Association of Chief Police Officers. Good practices guide for computer based electronic evidence. National High Tech Crime Unit: United Kingdom; 2003.<http://www.nhtcu.org/ACPOGuidev3.0.pdf>.

Yeats WB. He wishes for the cloths of heaven. In: The wind among the reeds; 1899.