Case study

Network intrusion investigation – Preparation and challenges

Andy Johnston, Jessica Reust

Information Security Professional, Baltimore, MD, USA
Digital Forensic Examiner and Investigator, Stroz Friedberg, LLC, 1150 Connecticut Avenue, NW, Suite 200, Washington, DC 20036, USA

Article info

Article history:
Received 14 July 2006
Accepted 4 August 2006

Keywords:
Intrusion investigation
Incident response
Network forensics
Digital forensic examination
Compromise of sensitive information
Forensic preparedness

Abstract

As new legislation is written mandating notification of affected parties following the compromise of confidential data, reliable investigative procedures into unauthorized access of such data assume increasing importance. The increasing costs and penalties associated with exposure of sensitive data can be mitigated through forensic preparation and the ability to employ digital forensics. A case study of the compromise of several systems containing sensitive data is outlined, with particular attention given to the procedures followed during the initial response and their impact on the subsequent digital forensic examination. Practical problems and challenges that arise in intrusion investigations are discussed, along with solutions and methodologies to address these issues. This case study illustrates both the importance of evaluating the evidence analyzed and of corroborating findings and conclusions with multiple independent sources of evidence. An initial response that incorporates forensic procedures provides a solid foundation for a successful and thorough forensic examination.

© 2006 Elsevier Ltd. All rights reserved.

1. Introduction

Investigations into network intrusions that involve the potential compromise of confidential data have taken on a new urgency with the passing of the state data breach notification laws. Organizations in both the private and public sectors are not only dealing with the embarrassment of the media coverage, but also the financial ramifications of notifying affected individuals, oftentimes in an effort to comply with these laws. This case study describes an incident that took place in 2005 that included the compromise of over 50 computers, some with confidential private data. The aim of the intrusion investigation was to determine the method of intrusion, the motivation of the intruder and whether the confidential data had been accessed and/or downloaded by the intruder(s). The investigation showed that although the confidential data on the
compromised systems could have been accessed and downloaded by the intruders, there was no evidence of such activity.

The resources of two groups were brought to bear in the collection and analysis of the evidence related to this case. The initial response to the compromise was carried out by the organization’s IT Security staff. Established incident handling procedures were followed until the severity and unusual characteristics of the compromise were discovered, after which more specialized efforts were employed to identify and secure evidence. Additional preservation and the evaluation of the evidence were performed by Digital Forensic Examiners from an independent organization.

This case study demonstrates the importance of forensic preparedness and employing digital forensics when responding to a critical incident. Practical problems that are encountered and judgments that must be made in an intrusion investigation are addressed in sidebars throughout this article. Organizations that are prepared to gather digital evidence and employ digital forensics put themselves in a better position to mitigate the increasing costs and penalties associated with exposure of sensitive data.

2. Identification of compromise

In May 2005, a server used to store sensitive information began to display a message that one of its drives was full and that it was unable to save a particular movie file. Since none of the server’s legitimate functions involved movies, the local system administration staff made a call to the IT Security Department of the victim organization. The IT Security Department quickly realized the significance of this message on a server with more than 400 gigabytes (GB) of storage that served a department with nowhere near that much data to store (especially not movies), and began to investigate. Initially, the IT Security personnel used their standard incident response toolkits and procedures. Initial response included live examinations of the server and of other systems on the same subnet to determine if they had been compromised. However, the responders noticed anomalies in the results that required additional examination.

The initial procedures included a remote port scan of the server to guide the on-site investigation. Initial responders on-site used a variety of utilities to determine the state of the live server, and then examined the output of those utilities for signs of known problems. In particular, network activity was examined for anomalous services in a listening state and for established or partially closed connections that might be inconsistent with the normal use of the server. The responders almost immediately reported two problems. First, several of the examination utilities failed to produce the expected output, or even to execute. Several families of known malware will interfere with the execution of some forensic software, but in this case the server was unable to detect the executable files themselves. Moreover, a review of the media (CDs and USB drives) containing the examination utilities revealed that the missing executables were not missing at all, but were in fact still there. The second problem appeared when the results of the remote port scan were compared with the apparent state of network connections in the server. The remote scan reported open ports that could not be detected from within the operating system of the server.

The next stage in the response was booting the server from a Helix distribution CD (http://www.e-fense.com/helix/contents.php) in order to examine the filesystems using Unix-based utilities, particularly the Sleuth Kit (http://www.sleuthkit.org). Sleuth Kit utilities use timestamps of the system files to create a timeline of file activity. System compromises of almost any sort are frequently associated with the creation of malware and other files within a short span of time. File activity timelines can be examined for “bursts” of file activity leading to the closer examination of specific files. In this investigation, the timeline implicated files that had not been detectable while the server’s operating system was running.

Several of these files, upon examination, were part of the Hacker Defender project (http://hxdef.org). Hacker Defender is classed as a rootkit: a utility or group of utilities that cloaks files and processes to prevent detection by the host operating system. Hacker Defender is designed for the Microsoft Windows operating system. It can conceal files, processes, associated registry keys, network connections, and perform various other services useful in the exploitation of a Windows system. In this instance, it was configured to conceal not only itself, but also remote administration and file transfer software.

The IT Security Department was aware that the server contained sensitive information and initiated preservation measures to support a full forensic analysis. During their initial response, IT Security personnel dumped the contents of memory to an external USB drive. A memory dump can contain significant amounts of valuable evidence including IP addresses, passwords, and other data related to an intrusion. As more tools are developed to examine memory dumps, this source of evidence will become increasingly useful.

A variant of the UNIX dd utility is included on the Helix distribution under the Live Acquisition option (Fig. 1). This variant runs under Windows and recognizes drive letters and Windows pathnames as devices. It also accepts the system’s physical memory as a device. The output is saved to an external storage device supplied by the responder that may be added to the rest of the secured evidence. The memory will then be available for forensic analysis.

The Helix Acquisition utility provides a GUI that generates a dd command string that can be pasted into a command window (Fig. 2). In addition to preserving memory contents to a file, this command produces an audit log of the imaging process and a MD5 hash of the memory image.

Sidebar – incident handling procedures

The proper training of Incident Handlers is critical to ensuring that anomalies indicative of compromises are detected even when standard incident response toolkits are undermined by modern rootkits. Memory contents can contain valuable evidence of intrusions, and should be correctly preserved whenever feasible.
3. Identification and preservation of evidence

Realizing the potential severity and scope of the incident, the IT Security Department took two concurrent courses of action: locating other compromised systems (if any) and collecting and preserving evidence for possible legal action.

To search for other compromised systems, a network scan was used to identify systems with scanned profiles similar to that of the server already examined. The systems found in this way were tested for behavior that identified Hacker Defender installations like that on the compromised server (e.g., disappearing forensics utilities). In addition, software was found that could be used to identify Hacker Defender. Compromised systems were then vetted to identify those with sensitive information. Those with such information potentially available to an intruder through the compromise were then treated as potential evidence.

The collection and preservation of evidence included the preservation, as far as possible, of those machines identified as evidence in the search described above. In addition to the computers themselves, Netflow data from the organization’s network were identified as potential evidence of network traffic to and from the compromised systems. Since Netflow logs were rotated over time, resulting in the deletion of old logs to free storage for new ones, log rotation was temporarily stopped until all existing logs could be archived off-line.

A management decision was made to engage an independent external organization to perform a digital forensic examination of the incident, so the IT Security staff recorded the
steps that had been taken to identify and preserve evidence. For evidence such as the Netflow logs, where the Digital Forensic Examiners would be provided with copies of original data, IT Security staff generated MD5 hashes of the original files to be provided along with the copies. As evidence was collected, it was also reviewed by the IT Security staff to refine the profiles being used to detect other compromised systems.

The Digital Forensic Examiners took steps to preserve these and other potential sources of information. A summary of information sources that should be considered for preservation and analysis is provided in Table 1.

To identify potential sources of evidence, the Digital Forensic Examiners interviewed the IT personnel, obtained network topology and security information, and took custody of the log files and memory dumps preserved by the IT Security Department. The Examiners also took custody of four computer systems containing confidential information that may have been vulnerable to the compromise, and transported the systems to their forensic laboratory. Complete and accurate forensic images of drives in these systems were created. An initial attempt to image the RAID server using EnCase indicated that it would take several days to complete. Using the bootable Linux component of the Helix distribution, the server was acquired in a matter of hours via a network cable (Fig. 3).

4. The investigation

After reviewing the four systems, the Digital Forensic Examiners determined that only the server and one of the desktops had been compromised. In addition to examining the forensic images of each system, bootable clones of the two systems that were not compromised were created to enable a live examination. A live forensic examination was performed on these bootable clones to determine whether a rootkit or unusual processes were running. The bootable clones were also connected to a dedicated private network and examined over the network for vulnerabilities and potential backdoors.

4.1. The rootkit

An analysis of the compromised systems revealed that the intruder(s) gained access to the desktop and server, and placed several files on the system (listed in Table 2).

These files were placed on the desktop on January 29, 2005, at 08:37 AM and on the server at 09:41 AM. This was approximately three months prior to the IT Security Department’s discovery of a potential intrusion.

Research was conducted to determine the function of the files placed on the systems by the intruder(s). The Hacker Defender rootkit (http://www.hxdef.org) enabled the intruder to hide files, folders, processes, services, ports, and registry keys from a user on the compromised system. Specifically, the rootkit on both the systems was configured to conceal the files placed on the system by the intruder, including its own renamed component files (winsvc.* and win.sys). The rootkit also concealed ports 3210 and 4489, and any folder named “.tmp.”

The remote administration (radmin) server gave the intruder full remote control of a computer, including the ability to view the screen and control the mouse. This was confirmed by performing a functional analysis of the software in an isolated test environment. The file transfer functionality of the radmin server is illustrated in Fig. 4. The documentation for this software states that the maximum file size that can be transferred is 2 GB. This limitation of the software eliminated it as an avenue of extrusion for the files containing sensitive data because they were larger than 2 GB.

The Serv-U FTP server was configured to listen for connections on port 3210, one of the ports hidden by the Hacker Defender rootkit. This server enabled anyone with the appropriate username and password to transfer files to and from the compromised system. This FTP server was configured to allow access to any folder on any volume on the system (from C: to Z:). To confirm that such broad access was possible, a functional reconstruction was performed using a bootable clone of the compromised desktop.

The name “cannon” was found in the Serv-U FTP server configuration file, suggesting that this is the nickname of the intruder. Other compromised systems examined by the IT Security Department were running an Internet Relay Chat (IRC) “bounce” server that connected to an IRC channel named “#cannon.”

Sidebar – planning the investigation

The Digital Forensic Examiners discussed the investigation’s scope and parameters with the client to ensure that they had a clear understanding of the client’s questions, concerns and priorities. A proper understanding of an investigation’s scope allows for the construction of targeted questions that need to be addressed. This in turn facilitates a comprehensive, methodical approach to the structure of the investigation. Finally, the investigators and the client can establish the products to be delivered to the client at the investigation’s close. All of these issues should be addressed and resolved at the inception of any investigation.

Table 1 – Summary of information sources for preservation and analysis

<table>
<thead>
<tr>
<th>Information source</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer media</td>
<td>Servers, personal computers, removable media</td>
</tr>
<tr>
<td>Memory dumps</td>
<td></td>
</tr>
<tr>
<td>Server application logs</td>
<td>HTTP, FTP, SMTP, OWA</td>
</tr>
<tr>
<td>Authentication logs</td>
<td>Host Logon WTMP/NT</td>
</tr>
<tr>
<td>Intrusion Detection Systems</td>
<td>Logs of Traffic Capture/Alerts</td>
</tr>
<tr>
<td>Network Management Systems</td>
<td>SNMP snapshots</td>
</tr>
<tr>
<td>Centralized Anti-virus Alerts</td>
<td>Syslog and NetFlow logs</td>
</tr>
<tr>
<td>Firewall &amp; router configuration and logs</td>
<td></td>
</tr>
</tbody>
</table>
4.2. Method of intrusion

Investigating the method of access requires review and analysis of the network-level logs, and forensic examination of the compromised computers themselves. These two evidence sources can provide independent corroboration and verification of identified evidence.

4.2.1. NetFlow log analysis

Many modern routers have the capability to record packet header information in the form of NetFlow logs, but this is not enabled by default. These logs contain information such as the source and destination IP address, IP type, source and destination port number, number of packets, and start and end transmittal time (Romig, 2004). A suite of tools, named ‘flow-tools,’ (http://www.splintered.net/sw/flow-tools) can be used to record, filter, print and analyze the logs generated by Netflow, i.e. the Netflow logs. Organizations are advised to capture this information at their Internet border and internal routers since it provides an overview of activities on the network without the overhead of capturing full packets.

As noted above, NetFlow logs were preserved in this case; these logs contained information about traffic between the victim organization’s network and the Internet. Unfortunately, the logs do not give a complete view of the traffic because a large number of the NetFlow datagrams were not captured. Approximately 45% of NetFlow datagrams were not captured on January 29, 2005. Such losses can be caused by network congestion or resource exhaustion on the NetFlow collector.

To filter the NetFlow logs with flow-tools in an attempt to identify the possible method of access, it was necessary to determine the IP addresses of the compromised desktop and server. The server was assigned a static IP address, which

---

Sidebar – functional reconstruction

Functional reconstructions of potential avenues of extrusion are necessary to ascertain whether the information in question could have been accessed and taken using those methods.

---

Fig. 3 – Grab interface on Helix distribution.
was easily determined. However, the desktop was configured to obtain an IP address using the Dynamic Host Configuration Protocol (DHCP).

It was not possible to determine the IP address that had been assigned to the desktop in January 2005. The most recent IP address assigned to this system was determined from the registry. Based on the Serv-U FTP welcome message, a different IP address was ascertained as having been assigned to the desktop at one time. A search of domain controller authentication logs from January 2005 for the account in use on the desktop was not successful in determining the IP address. The Digital Forensic Examiners analyzed the Internet Explorer history for the desktop around January 29, 2005, in the hopes of being able to identify an unusual website visit. The date and time of the visit to the website IP address on the desktop was not successful in determining the IP address.

The date and time of the visit to the website IP address on the desktop was not successful in determining the IP address. This analysis was not successful. However, this may have been due to the loss of NetFlow datagrams as discussed above. Because the desktop’s IP address changed may have been due to the loss of NetFlow datagrams, the hopes of being able to identify an unusual website visit were unsuccessful. The analysis of the NetFlow logs relating to the intrusion and other activity by the intruders on the desktop was limited.

The NetFlow logs for the morning of January 29, 2005 were examined. The logs showed that a scan of port 135 had been performed. Discussions with the IT Security Department revealed that all traffic from port 135 was blocked, however, so this was not the method utilized by the intruder(s). Further review of the NetFlow logs did not identify a method of access, although traffic appearing to be associated with the download of the rootkit files onto the server was identified. No NetFlow logs had been retained that corresponded to any IP address around the time the desktop computer was compromised, possibly because it was compromised from within the victim network. The victim organization did not monitor or log the traffic activity within the victim network, and therefore no information was available about the intruder(s)’ network activities within the organization. It was not possible to determine where the rootkit files were downloaded from without the IP address of the computer and without any way of determining the activity of the intruder(s) within the internal network.

4.2.2. Forensic examination of computer media

Forensic examination of the desktop and server revealed strong evidence that the intruder(s) had gained access to both the desktop and server through guessing weak passwords. For both systems, the passwords for two of the Windows user accounts (IUSR_WIN and SUPPORT) were changed on January 29, 2005. On the desktop, these two accounts were changed at 08:37:31 AM, 11 s after the first of the rootkit files were downloaded, strongly suggesting that this password was chosen by the intruder(s). On the server the passwords of the IUSR_Win and SUPPORT accounts were changed at 09:41:40 AM, 8 s after the first of the rootkit files were downloaded.

### Sidebar – incomplete logs

Incomplete logging can be detrimental to the investigation, sometimes even more so than no logging, as it takes up analysis time with limited or no results. Furthermore, without the IP address of the compromised system at the time of intrusion, analysis of network-level logs is limited. This does not, however, mean that incomplete logs cannot be valuable in an investigation. If analysis of incomplete logs is undertaken, it is important to realize the additional time it will likely take, and the limitations it places on the conclusions drawn.

### Table 2 – Files placed on compromised systems by the intruder(s)

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Winsvc.exe</td>
<td>Hacker Defender executable</td>
</tr>
<tr>
<td>Winsvci.ini</td>
<td>Hacker Defender configuration file</td>
</tr>
<tr>
<td>Win.sys</td>
<td>Hacker Defender driver</td>
</tr>
<tr>
<td>Msdtsr32.exe</td>
<td>Serv-U FTP server executables</td>
</tr>
<tr>
<td>Tzoib.dll</td>
<td>Component of Serv-U FTP server</td>
</tr>
<tr>
<td>AvMeter32a.dll</td>
<td>Configuration file of Serv-U FTP server</td>
</tr>
<tr>
<td>ServUStartUpLog.txt</td>
<td>Serv-U FTP server startup log</td>
</tr>
<tr>
<td>WELCOME.MSG</td>
<td>Serv-U FTP server welcome message</td>
</tr>
<tr>
<td>R_server.exe</td>
<td>Remote administration server (<a href="http://www.radmin.com">www.radmin.com</a>)</td>
</tr>
<tr>
<td>AdmDll.dll</td>
<td>Component of r_server</td>
</tr>
<tr>
<td>AVI32.DLL</td>
<td>IRC-related file</td>
</tr>
<tr>
<td>FPORT.EXE</td>
<td>Utility for viewing listening ports and the associated process</td>
</tr>
<tr>
<td>PSLIST.EXE</td>
<td>Utility for listing processes</td>
</tr>
<tr>
<td>PSDKILL.EXE</td>
<td>Utility for terminating processes</td>
</tr>
<tr>
<td>UPTIME.EXE</td>
<td>Utility for determining how long a computer had been running</td>
</tr>
<tr>
<td>RESTART.BAT</td>
<td>Batch file for restarting a process named &quot;ntserv&quot; referenced in winsv.ini but was not found on the which is system</td>
</tr>
<tr>
<td>DTREG.EXE</td>
<td>Program for batch registry modification</td>
</tr>
<tr>
<td>MPEGC.DLL</td>
<td>IRC bot configuration file</td>
</tr>
</tbody>
</table>

### Table 3 – Dates of incorrect password logon attempt for the compromised desktop

<table>
<thead>
<tr>
<th>Account</th>
<th>Last incorrect password logon attempt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrator</td>
<td>01/29/2005 09:26:33</td>
</tr>
<tr>
<td>IUSR_WIN</td>
<td>01/29/2005 09:26:33</td>
</tr>
<tr>
<td>SMSCISvcAcct&amp;</td>
<td>01/29/2005 09:26:34</td>
</tr>
<tr>
<td>SUPPORT</td>
<td>01/29/2005 09:26:34</td>
</tr>
</tbody>
</table>

### Table 4 – Date of incorrect password logon attempt for the compromised server

<table>
<thead>
<tr>
<th>Account</th>
<th>Last incorrect password logon attempt</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBADMIN</td>
<td>01/29/2005 09:35:00</td>
</tr>
</tbody>
</table>
downloaded, adding weight to the hypothesis that the intruder gained access via a weak password that they subsequently changed to prevent others from gaining unauthorized access.

It is also possible that these systems were being scanned for weak passwords from an internal system as a number of the accounts on the desktop and server had a ‘Last Incorrect Password Logon Attempt’ on January 29, 2005 as detailed in Tables 3 and 4.

The Security Event Logs for both the desktop and server were empty, limiting any further analysis or determinations that could be made regarding these logon attempts, such as their origin. In addition, as noted above, there were no network-level logs of intra-domain connections on the network, so no information was available to determine if this scenario is in fact what happened or from which computer the attacks were initially launched.

Other evidence suggested that the compromised desktop was being used to break into other systems. A search of unallocated space on the desktop for FTP “GET” and “PUT” commands revealed significant amounts of IRC and exploit related activities.

For instance, the following items, all indicative of IRC and exploit related activities, were found in unallocated space:

- `ftp -i GET dcom.exe`
- `mircdir\shiver.exe cygwin1`
- `HTTP P/1.0 IIS buff flow eeye.com`
- `GET/scripts/./%255c.%255cwinn cmd.exe?c+ □ ÝÔ y' f p<'y' Tups uuhuh.myetang.co -finger -ftp`  
- `\ACHTUNG ON\RUN PASSWORD REGISTRY`
- `vmload.vxd Qbik WinGate GDP\] Qbik InetService`
- `OPENPORT OWS\CUR BACKDOOR`
- `DataDrain UserID Password ParseList ProxyPort`
- `DALLY PORT IP_TO TARGET ROOT PACKET_F_EDUP SOCKET TARGET PATCHED`
- `MAILSNARF DSNIFFÔ « URLSNARF TCP_QUEUE « WEBSPY TCP_QUEUE « DSNIFF PASS ICQ`

None of the programs related to these exploits were found on the compromised desktop.

In addition, on January 31, 2005, several days after the server was compromised, seven patches fixing vulnerabilities that allow remote code execution and elevation of privilege were applied via Windows Update. The installation of the patches on January 31, 2005 suggests that the server was vulnerable to the named exploits at the time of the intrusion.

The same is not true for the desktop – the same seven patches were applied before the compromise. Specifically, the seven patches that were applied on the server on January 31, 2005 were applied to this system via Windows Update on January 17, 2005. Having the patches applied indicates that the system was not vulnerable to the named exploits at the time of the intrusion on January 29, 2005.

4.3. Intruder activities

The D: volume of the server contained a folder “WUTemp\tmp,” that was hidden by the Hacker Defender rootkit. This “.tmp” folder contained hundreds of pirated movie files, most of which contained the word “German” in their filename. Also notable was the name of the first file in the “.tmp” folder: “D:\WUTemp\tmp\# BOX IS FULL -DONT UPLOAD #” The intruder(s) were obviously aware that there was limited space on the server but this warning was evidently disregarded, resulting in the “disk full” message that alerted the IT Security Department to the intrusion.

The memory image preserved by the IT Security Department contained an IRC exchange over a connection established by the intruder, as shown in Fig. 5. This exchange was identified with the UNIX utility strings, but a binary editor could also have been used to view this information.

Interestingly the IRC exchange is in German, which is consistent with the language in which the movie files are named.
4.4. Confidential data

The sensitive data of concern were stored in SQL database files. The functional reconstruction determined that the intruder(s) had access to all volumes of the compromised computers, and therefore had access to the database files that contained the confidential data. Further investigation into the intent of the intruder(s) and the presence of any evidence of the download of these database files was performed.

All of the database files on the server were last accessed on April 13, 2005 at 21:40. The only other files being accessed at this time were movie files, a fact that potentially indicated the accessing of the database files by the intruder(s). However, the database files were also last modified on April 13, 2005 at 21:40, and a review of the Application Event Log entries showed that the server was shutdown at this time due to a lack of space on the server. It was this shutdown that caused the last access and modified times of the database files to correspond with the times of last access of the movie files. This was not evidence, therefore, of the intruder(s) specifically accessing the database files with confidential data.

The last accessed date of a file only shows when it was most recently touched. Therefore, it was necessary to rely on network-level logs for the analysis. NetFlow logs between February and April showed significant amounts of data being transferred from the compromised server to external computers, most likely relating to the upload and download of movie files found on the compromised server. Because of the significant losses in the NetFlow logs, it was not possible to compare the number of bytes transferred over the network with the sizes of the files in question.

The database files on the desktop were last accessed and last written on the day that the system was shutdown, April 15, 2005. The shutdown of the system caused the SQL server to shutdown, and the last modified dates for the database files to be updated. Therefore, it could not be determined whether the intruders downloaded the database files on the desktop before the last day of shutdown. The most likely way that the large database file could be transferred from the server to an external system was via the FTP server installed by the intruder, and it was necessary to rely on network-level logs for the analysis.

Examination of the NetFlow logs between January and April 2005 for both of the possible desktop IP addresses revealed no flows relating to any of these IP addresses that exceeded six packets. Therefore, the database file containing confidential data does not appear to have been downloaded to an external computer from either of these IP addresses. As mentioned above, because the desktop’s IP address changed over time, it is possible that it was assigned an IP address other than the ones identified through the hard drive analysis.

4.5. Determining intent

It is important to consider all available evidence when determining the intent of the network intruder(s). In this case, there was evidence of widespread, largely indiscriminate targeting of systems on the victim network. There was no evidence that the intruders were specifically seeking sensitive data such as those found in the database files, nor was there evidence that the database files had been accessed and downloaded by the intruder(s). The available evidence indicated that the primary intent of the intruders was to store digitized movies on the compromised systems for others to download. It must also be stated, however, that although the confidential data in the databases were not specifically targeted, the intruders had the ability to download them after gaining unauthorized access to the systems.

5. Important points and lessons learned

Identifying the potential sources of information about the intrusion is an essential step in a network intrusion investigation. To do this successfully and thoroughly, the Examiner must understand where on a network information can be found (Casey, 2004). As information in an intrusion investigation tends to be fragile and transient in nature, delays in identifying and querying information sources can result in the loss of the information altogether. The right questions need to be
asked of the right people. The correct tools need to be used to obtain the most complete and accurate data set, and proper procedures for documenting the integrity and authenticity of the evidence must be followed. A methodical search of the network for potential sources of evidence is the preferred approach, and the creation of a network-specific “digital evidence map” can facilitate this search (Casey, 2004).

The Digital Forensic Examiners in this case study were assisted by the well-prepared IT Security Department of the victim organization, who identified the rootkit installed by the intruders and the computers with confidential data that were potentially compromised. The IT Security Department also performed initial data preservation steps prior to the Examiner’s arrival on the scene, and was able to provide detailed information for identifying the relevant evidence sources on the network.

The more traditional digital forensic investigations typically involve the analysis of bit-stream images of computer media that were not powered on when preserved. In intrusion investigations, there are typically a large number of computers that contain pertinent information, and the information is of a type that does not necessarily require the preservation of the entire bit-stream image of the computer system. If the target data are configuration settings and logs, creating a logical copy of the logs with a tool that preserves MAC times and generates a hash algorithm of the files copied would be the more efficient and effective solution. In addition, some situations require that digital evidence be collected remotely. Context documentation and chain of custody would need to be maintained in these scenarios.

Examiners had to learn a significant amount about NetFlow and flow-tools in order to identify and parse out the relevant information from the NetFlow log files. In many intrusion investigations, there are often unique sources of evidence that forensic examiners need to become familiar with in order to complete the investigation.

If circumstances permit, the preservation of memory is an important step to take in an intrusion investigation. Volatile memory contains, among other things, live processes and often their associated data. These data may well include information that the user of the system entered such as passwords, URLs, chat conversations, etc. Studies suggest that data persist for several hours or even days (Farmer and Venema, 2005). Some of these data may also exist in page files or swap space on the hard drive.

Any process that is run to capture memory will use some portion of that memory and thus disturb its state. It is especially important, therefore, that memory is preserved correctly. A simple and well-understood program, such as dd, should be used to capture memory.

Given the numerous disparate sources of evidence usually involved in network investigations, findings and conclusions must be corroborated with multiple independent sources. Any conclusions reached with incomplete but corroborated evidence should be accompanied by appropriate caveats. Incomplete evidence, such as NetFlow logs with missing datagrams, may be distracting and erroneously divert the focus of the investigation. It is important for the Examiner to know the limitations of the evidence under review, especially when the investigation is occurring some time after the initial intrusion took place.

A network logging infrastructure designed with an appreciation of the importance of digital investigations and forensic principles significantly assists network investigations (Casey, 2005). Implementing internal monitoring of network activities is not feasible for some organizations. More detailed and accurate conclusions about an intrusion may be reached if a more complete picture of the intruder’s activities can be established. Similarly, limitations are placed on conclusive statements if logging is implemented but only a portion of the traffic is actually being logged.

Acknowledgements

Ms. Reust would like to thank her colleagues at Stroz Friedberg LLC for their support and assistance with this article, particularly Eoghan Casey.

Mr. Johnston would like to thank Ms. Erin Kelly for her proofreading efforts and Mr. Eoghan Casey for his encouragement and tea.

References