

A Method for Forensic Previews

Timothy E. Wright 2005-03-16

1. A Classic scene from the information security professional's work life

One of your systems administrators pokes his head in your office door. "The print spooler machine may have been compromised. Can you help me take a look? Some odd files have appeared -- that's all we know right now." Your pulse steps up a few beats: you told Operations on more than one occasion that they should address the availability issues faced by critical servers. The print spooler was one of those servers. If it is hacked, it will have to be taken out of production, and there will be serious consequences due to the service interruption. At least you have documented your interactions with Operations: email is forever, you tell yourself. With that thought, you ponder your options to get the organization through this as painlessly and quickly as you can. There is no backup machine, and obtaining a bit-for-bit copy of the spooler's file space is not practical without taking the machine off line. Since there is no solid evidence that the spooler is hacked, it makes sense to do some reconnoitering before taking the machine out of production for extensive forensics. The things you would like to look at include process and network activity, the status of significant binaries, user and group accounts present, the permissions these accounts have, and so on. But how to proceed with this forensic "preview" of the spooler? You do not wish to damage original evidence, and if the spooler is not hacked there is nothing to worry about. On the other hand, what if it *is* hacked?

2. The preview process

During any computer forensics operation, the state of the target machine must be left as undisturbed as possible. This underlying principle applies to all forensics activities, ranging from the field preview to the full blown examination in a lab. Nevertheless, there remains an important distinction between a preview operation and lab work: by its nature, the preview is very likely to contaminate original evidence. Examinations in an evidence preservation lab use backup copies of evidence, thereby preserving the initial state of crime scene equipment. Why, then, would an investigator undertake a preview operation? There is often no choice, as the opening scenario demonstrates. But perhaps previews are not that far out of line. After all, risking damage to the original evidence is something an investigator faces during the initial steps of most forensics work. Some level of interaction with the crime scene computer is normally required to obtain a backup for later processing. This issue may even be exacerbated

when the crime scene computer is something other than a workstation (such as a mainframe), in which case, significant interaction may be required to backup any evidence.

Where computer forensics is concerned, the idea of *less is more* carries great weight. The less an investigator has to do to interact with and extract information from evidence (or what may become evidence), the better. In the case of the preview, the goal is to determine whether or not a given target machine has been compromised by some unauthorized agent. This determination has to be made without seizing the target machine and forensically processing a backup of its file space.

Following the preview, appropriate next steps may be taken if there has been some sort of compromise. For example, if a machine is simply infected with a virus, perhaps running a virus scan will be sufficient; if a machine has been turned into a "warez" site, perhaps removing it from production and putting it through a full forensics examination is in order. [ref 1] Clearly, the outcome will depend on the sensitivity of the data assets involved, the standing policies of the organization, and the professional assessment of the investigator.

3. The Four Step Plan

We have established what a preview is, and why an investigator might undertake such work. Now, we turn our attention to the broad steps that comprise the forensic preview activity:

1. Related research
2. Passive network operations
3. Active network operations
4. Active host operations

As we precede through these steps the investigator's activities become progressively more interactive with the target machine and, hopefully, more revealing of the machine's disposition. Unfortunately, as the preview becomes more interactive, it also becomes more dangerous to the state of evidence. Therefore, it is important that the investigator stops the moment a compromise is evident; continuing on would needlessly risk damaging original evidence. With this approach, it may be possible to determine that a given host has been compromised without, for example, having to directly interact with the operating system looking for a root kit.

Before outlining these steps further, a couple of important guidelines deserve attention:

- *Always consider the possible legal ramifications of investigatory activities; consult with your organization's legal counsel in advance of such activities.* For example, some of the steps outlined below may constitute a violation of privacy, given the right circumstances
- *Document all investigative activities taken.* The whole reason to do a forensic preview is to determine, without disruption to production services, whether or not a target machine has been compromised. If it has, the investigator may need to account for the interactions that have taken place as a result of the preview. A compromise does not necessarily translate into a full blown investigation: whether or not a target machine suddenly becomes a crime scene computer is contingent on the type of compromise, organizational policy, and the investigator's judgment. Regardless, all previews are the same in that the target machine **could** become a crime scene computer. If this happens, the investigator's preview documentation will become the start of a chain of custody [[ref 2](#)]

3.1 Step 1: Related Research

In the first step, the investigator uses the process of information discovery to research activities related to the target machine. This is not unlike the process of information discovery described in the Field Guide series of forensic articles on SecurityFocus. [[ref 1](#)] Of interest are log data and network flow information made accessible at the enterprise level, including:

- File space monitoring (e.g., logs of unexpected changes to files)
- Intrusion detection system (IDS) activity - network and/or physical
- Firewall activity
- Network flows
- Relevant service/application activity
- Interviews with relevant parties (e.g., system administrators, application administrators and users)

The idea is to find evidence of a compromise without interacting with the target machine on any level. Of course, success will depend on the monitoring in place (and that the logs in question are not stored on the target machine), as well as the quality/quantity of information provided by relevant parties.

If evidence of a compromise is found, the investigator should stop the preview and consider

handling the target machine as a crime scene computer. Otherwise, the preview should continue to Step 2.

3.2 Step 2: Passive Network Operations

In this step the investigator uses downstream/inline utilities to observe the target machine's ingress and egress traffic. There are a variety of ways to do this, including network taps, network IDS rules, and span ports on switches. Outside of the use of a span port, sniffing on a switch is not necessarily recommended since it may involve poisoning the ARP cache of the target host (changing the host's state, and perhaps interrupting its services). If the target is on a hub, or is wireless, sniffing becomes a safer choice to implement.

The duration used to monitor traffic depends on the investigator's comfort level with the situation. If the target machine is fulfilling a critical function, or stores highly sensitive data, it may be unreasonable to spend a lot of time in this step.

As in Step 1, if evidence of a compromise is found, the target machine may need to be viewed as a crime scene computer. If nothing of interest turns up, the preview should head to Step 3.

3.3 Step 3: Active Network Operations

By Step 3, the safer, non-interactive means of checking the target machine for compromise have been tried. From here on, the target machine's state will be altered by the activities of the preview. The investigator must minimize these activities to prevent significant harm to potential evidence.

In this step, the two primary tools of interest are port and vulnerability scans.

Port scans will not drastically change the state of a target machine. Nevertheless, the investigator should be aware that a listening service may write out log entries or start and stop processes upon connection establishment. If the target machine is running a network IDS, a port scan may cause a change in network disposition: the scanner could become blocked. The investigator should work with the system administrator to determine what services might interact with a port scan. If there is an IDS or firewall on the target machine, it may be possible to configure the scanner with a trusted address.

Unlike the port scan, vulnerability scans can cause significant changes in the state of a target

machine. The degree of change depends on how the scanner is configured, with more robust configurations leading to ham-fisted probes and attacks. The system administrator may be able to help fine tune a vulnerability scan, so as to not unnecessarily disturb a host's state. For example, if the target machine has been patched against vulnerability X, it does not make sense to check for X. One reasonable approach is to tune the vulnerability scanner to check for services commonly deployed by script kiddies and malware. Precise and simplistic scans are best: less time will be needed and fewer changes to the target machine's state will result.

Once again, if evidence of a compromise is discovered, the investigator should decide whether or not the target machine becomes a crime scene computer. If no compelling evidence turns up, the preview should advance to Step 4.

3.4 Step 4: Active Host Operations

Here, we directly interact with the target machine's operating system by way of a user account. The careful notes the investigator has been taking all along will carry even more weight in this step, since the activities herein are all but guaranteed to change the target machine's state. Items of interest include basic facts about the target machine's OS, process information, log file data, account information, and the status of file space.

To begin with, the investigator may wish to change the administrative password on the target machine. So long as this is documented, there's little reason that it would jeopardize any evidentiary value. If there is a compromise, it may be negligent to not take steps that help block an attacker's administrative access -- the investigator should consult with legal counsel in advance of preview activities.

In this step, we are concerned with the following information targets:

1. Basic system information
2. Running processes
3. Timed jobs
4. Log files
5. User and group accounts
6. File space status

Utilities that aid in gathering the above should come from a known, secure source. It is recommended that such programs be run off of read-only media (e.g., CD-R) to manage the

risk of using compromised programs on the target machine. However, there is a catch: many utilities are not self-contained and may rely upon the use of libraries and other resources on the target machine. It is impractical to fully avoid this situation; after all, by its very nature the forensic preview interacts with what could become original evidence.

Along these lines, as files are accessed on the target machine, the times and dates of these accesses will overwrite values in the relevant file metadata. This could make it difficult to show or know that an attacker has made similar accesses, and highlights the tradeoff of forensic previews: *in exchange for not taking a target machine out of service, there may be some contamination to possible evidence.*

Thought must also be given to data capture during the preview. The investigator might use a network agent to transmit and remotely store all information (e.g., [cryptcat](#), [SBD](#)). Any such agent should use strong encryption to ensure the integrity and confidentiality of transmitted information. As an alternative, data could be stored locally to a diskette or USB drive. The volume of data collected should be quite small, consisting of the text output of various utilities, along with copies and excerpts of logs.

To proceed through Step 4, a script or program could be used to collect most, if not all, of the information desired. [[ref 3](#)]

Item 1: Basic System Information

Here, we need to collect the basic facts about the target machine. While it is unlikely that this will yield evidence of compromise, the information establishes a context and helps to inform the preview.

What to capture:

- Hardware configuration (though, nothing requiring an interruption of service, like rebooting to get into BIOS, and so on)
- Operating System used, including version and patch level
- Network configuration (IP and MAC addresses assigned to all NICs, ARP cache)
- Major applications installed (though, not necessarily running), and, if possible, their patch levels
- Purpose of the target machine

Item 2: Running Processes

Under this item, processes listening for network connections are of primary interest. Open ports should be compared with what the system administrator believes should be open. Noting the services commonly associated with these ports can also be useful: if the target machine is suddenly offering an IRC service there could be reason for concern. Of equal importance are unusual outbound destinations or traffic types (for example, perhaps the target machine is not hosting IRC, but there is traffic seen going to an IRC server).

Processes that are not listening to a network port can be of interest, too (e.g., a sniffer process monitoring all of the network traffic on the target machine).

What to capture:

- A list of all running applications (with as much detail as possible: name, owner, resources consumed, duration of execution, process ID, libraries and files used, etc.), broken down by
 - Applications listening for network connections
 - Applications not listening for network connections
- A list from the system administrator of the applications that should be running

Item 3: Timed Jobs

A timed job is one that is scheduled to execute at some point in the future, perhaps iteratively. It may be that the scripting used in a timed job has been altered for malicious purposes. Thus, the investigator should be careful to not only find out what jobs exist, but to inspect their related programming.

What to capture:

- A list of all timed jobs, broken down by
 - Jobs to be run at the system level
 - Jobs to be run at the account level
- Results of reviewing (in whatever capacity is useful) scripting used in timed jobs

Item 4: Log Files

For this item, the investigator should gather system/application alerts and log entries. It is possible for preview activities to end up in the log files under review - notes maintained by the investigator will explain such entries.

The investigator should not overlook host-based firewall and network IDS logs. There may also be tremendous value in reviewing logs that are generated by proprietary applications.

What to capture:

- Important system level messages (such as errors, house keeping, application related messages)
- Account access events (authentication and authorization) at both the system and application levels -- to the extent possible, note the fundamental details
 - Who (i.e., account in question)
 - What (i.e., type of event)
 - When
 - Where (i.e., from where did the access originate)
 - Why (i.e., what was the perceived purpose of the access)
 - How (i.e., through what type of channel did the access happen)
- Important application level messages (e.g., web servers, host firewalls, host intrusion detection systems, etc.)

Item 5: User and Group Accounts

Here, we want to see if there are any unauthorized accounts on the target machine, and whether or not any accounts have been assigned unjustified access permissions.

What to capture:

- A list of all individual and group accounts
- A list of all currently active accounts (for example, who is on the system right now? What are they up to?)
- A list of critical file resources (such as data files, applications, etc.) on the target machine, along with their assigned permissions

Item 6: File Space Status

Last, the investigator should enumerate file permissions (note the overlap with **User and Group Accounts** above), look for unauthorized file activities, and check for unusually named and hidden files. Doing more than this is not practical from a time perspective, and could cause an undue processing burden. If the target machine should become a crime scene computer, there will certainly be occasion to make a file space backup, search for strings of interest, examine slack and unused blocks, and build a timeline of activities.

What to capture:

- A list of important and critical file resources on the target machine, along with their assigned permissions
- Any local, file space monitoring logs (if they exist)
- A list of unusually named, and hidden files

Overall, this step is clearly more involved than the previous ones due to its fully interactive nature. This makes it an ideal candidate for some level of automation through programming and/or scripting. As with the previous three steps, if evidence of a compromise is uncovered, the investigator will need to determine whether or not the target machine is a crime scene computer. If no such evidence is uncovered, the best the investigator can do is claim a low probability that the target machine has been compromised.

4. Departing Thoughts

There may be concern about the time needed to apply this forensic preview method. Going back to the opening scenario, what if it had to be immediately known whether or not the spooler was compromised? This may be a pointless question for the following reasons:

- Of course it has to be immediately known! Is it really ever okay to put something like this off?
- Because the forensic preview activities do not interrupt a target machine's production service, the investigator should be allowed to come to a conclusion as soon as possible -- not within some arbitrarily short time period. That said, this preview method is designed so that analysis happens as the four steps unfold. Doing otherwise may needlessly contaminate potential evidence
- The first three steps have the potential to be evaluated very quickly. Their speed

depends on how mature an organization's monitoring processes are, and how readily available and knowledgeable the system administrator is

- The last step can be streamlined if the investigator spends time assembling the necessary tools and a plan of attack

Perhaps a more important issue is what to do if a preview fails to reveal a compromise. *A secure target machine is not indicated by a failure to uncover evidence of compromise.* At best, an investigator can only claim a low probability that the target machine is compromised. The next steps depend on three things:

1. The organization's policies with respect to incident handling
2. What has lead the system administrator to suspect a compromise
3. The investigator's judgment given the sensitivity and criticality of the data present on the target machine

Based on the above, the target machine might be removed from production for a more thorough examination. On the other hand, given that nothing was found in the forensic preview, the cost of service loss may outweigh the risk of leaving the machine in production. The decision (and risk) rests with an organization's management.

Perhaps the most compelling reason to use a forensic preview method is that it helps to maintain the evidentiary value of a target machine. By using a repeatable, documented method, and by carefully noting all actions taken, the investigator can rationally account for the state of gathered evidence. This is essential if a chain of custody needs to be established as more rigorous forensics operations take place.

Remember that the forensic preview is not a panacea! The bottom line is that some activities in the preview process can significantly disturb potential evidence. To manage this risk it is critical that organizations formally document and implement a preview procedure for investigators to use. Doing so will establish a sound method that can be applied in most any circumstance, assigning credibility to the actions taken by the investigator, and to the evidence gathered.

References

[[ref 1](#)] For a complete description of the search and seizure process as it relates to computer crime, please see my earlier series of articles titled "[The Field Guide for Investigating Computer Crime](#)" at <http://www.securityfocus.com/infocus/1244>.

[[ref 2](#)] The chain of custody, or chain of evidence is a means of accounting for who has touched a given piece of evidence, when they touched it, and what they did to the evidence.

[[ref 3](#)] A forensics toolkit suitable for previews can be found on the web at <http://www.e-fense.com/helix/>. While still a little rough around the edges, Helix offers tools for preview work on several computing platforms.

Related links

http://www.sans.org/score/checklists/ID_Windows.pdf

http://www.sans.org/score/checklists/ID_Linux.pdf

<http://www.sysinternals.com>

<http://www.cisecurity.org>

<http://www.cert.org>

http://www.cybercrime.gov/s&smanual2002.htm#_IIIA_

<http://www.sleuthkit.org/index.php>

<http://www.securityfocus.com/infocus/1244>

<http://www.cycom.se/dl/sbd>

<http://farm9.org/Cryptcat/>

<http://www.e-fense.com/helix/>

About the author

For the past several years, [Timothy Wright](#) has been investigating computer fraud and abuse in the private sector and, more recently, higher education. He has worked as a Senior Technology Investigator at one of America's largest financial corporations, and as a lead developer within the financial industry, designing and building web-based home banking software. He presently works as an IT Auditor for a university in the midwest United States, and holds an M.S. in Computer Science, and a B.A. in Philosophy.

View [more articles](#) on forensics and computer intrusion by Timothy E. Wright, CISSP, CISA, on SecurityFocus.

Comments or reprint requests can be sent to the [editor](#).

[Privacy Statement](#)

Copyright 2006, SecurityFocus