

Network Forensics

Methods, Requirements, and Tools

November 2007

By J. Scott Haugdahl

Founder and CTO, Bitcricket

Abstract

Network forensics includes the recording and analysis of network events to figure out the nature and source of information abuse, security attacks, and other such incidents on your network. This is typically achieved by recording or capturing packets long term from a key point or points in your infrastructure (such as the core or firewall) and then data mining for analysis and recreating content.

This white paper looks at the many aspects of forensics ranging from compliance, to law enforcement, to user behavior. We briefly summarize findings from Carnegie Mellon that studied various forms of IT espionage and sabotage inside the enterprise. Requirements to consider in evaluating commercially available tools are examined. Finally, a practical example of using such a tool is demonstrated to detect anomalous activity.



The Many Faces of Forensics

Quick, what's the first word that comes to mind when you hear the word 'forensics'? C.S.I. perhaps? Despite being an IT guy, I still have morbid thoughts of a pathologist slicing into a cadaver along the lines of Jack Klugman in Quincy M.E. Or collecting stuff that might contain the perp's DNA at a crime scene (fingerprints are so yesteryear.)

Webster's defines forensics as "The use of science and technology to investigate and establish facts in criminal or civil courts of law."

I think this fits well with corporate needs for network forensics using analysis tools. Many vendors have tools with catchy taglines on variants of "Retrospective Analysis," "Business Forensics," "Turn Back the Clock," and so on. Unlike real-time, the basic premise behind network forensics is to mine data (usually via packets) and perform post analysis to reconstruct content or gather intelligence as to why certain things happened. In some ways, forensics is like detailed hindsight.

There are several areas where forensics can be applied. Samples of some broad categories include:

- Compliance: Oops, someone sent out company confidential financial information in an unencrypted email or used IM to gossip about a coworker's medical condition, a HIPPA violation.
- Troubleshooting: Why did your network meltdown this morning? Why do your CRM users often experience poor performance in the afternoon?
- Hackers: What was hacked, how, and by whom? Often goes hand-in-hand with intrusion detection systems (IDS) to see what damage if any, was done. It's also a good way to verify that intrusion *prevention* systems (IPS) are working too.
- Verticals: Why did the core switch peg during a critical trading hour? Why are doctors losing wireless connectivity? Is our converged data + VoIP transport operating smoothly?
- Law Enforcement: In particular, CALEA (the Communications Assistance for Law Enforcement Act of 1994), which states the requirements of carriers to assist law enforcement in executing electronic surveillance. CALEA is of interest more so outside the enterprise – i.e. Internet service and Internet backbone providers.

Returning to Webster's definition, analysis tools can be used to establish facts as to a particular network-related event that is disturbing. By network, I mean the entire infrastructure from fabric to nodes to users – let's not forget about the human element.

According to searchnetworking.com, Marcus Ranum is credited with saying “Network forensics is the capture, recording, and analysis of network events in order to discover the source of security attacks or other problem incidents.” In other words, we want to document such incidents for analysis and recreating events, not unlike “bagging and tagging” a real crime scene.

Thus, by virtually all definitions of the term, forensics is traditionally associated with crime solving. This puts us more in the overall thinking of “when illicit things happened, what happened, and how can they be prevented?” whether from inside or outside sources. Contrast this to troubleshooting as mentioned above. Forensics is criminal. A slow network is not. Although one could argue that a dead network is criminal.

There’s also a relatively new category of forensics – enterprise forensics that focuses both on user activity and what drives or doesn’t drive the business (analytics + behavior). Is what we’re seeing on the network consistent with business objectives? Apdex (a measure of application satisfaction with respect to end-users) is a great metric for this. Now we’re crossing the boundary over into Application Performance Management (APM) along with user behavior and IT vs. business expectations.

Some companies in fact have attempted, with limited success, to focus on the behavioral aspect of forensics. According to the book *Digital Evidence and Computer Crime*, “Behavioral evidence analysis provides a systematized method of synthesizing the specific technical knowledge and general scientific methods to gain a better understanding of criminal behavior and motivation.”

The big question here is whether or not unusual behavior can be predicted in advance based on characteristics of anomalous activity. So what if Johnny does a 2 gig file transfer on Friday afternoon? Maybe it’s a routine back-up. Maybe it’s a one-off OS patch. Furthermore, who cares if our WAN utilization is 100% during that time, as long as it’s fair access for all and all access for one when no one else is on at the moment? Naturally, illegal file sharing and such is cause for alarm, but there will always be unpredictable anomalous spikes that don’t fit baselines.

I used to teach in my network analysis and troubleshooting courses that having 100% utilization is not necessarily a bad thing and should not be setting off SNMP traps and alarms all over the place. Of course, you don’t want one user or application to consume all the bandwidth for extended periods of time. But brief bursts of 100% can actually be a good thing. After all, if no application can ever utilize 100% of the pipe, then perhaps we should optimize things. The key is to get on and off the network as quickly as possible – big pipes (and low end-to-end latency) help to achieve that. But I digress.

Forensics tools need to provide the flexibility to blend real-time analysis with post-capture forensics and allow you to optimize the tool for your given situation. Is 100% capture to disk of massive amounts of data important to you? Where do you need to cover (capture) in your network, what is the nature of the traffic, and what are the capture bandwidth requirements? How long do you need to keep the data around? How important are the distributed aspects and how efficient is the data conveyed to centralized consoles or distributed consoles shared by multiple engineers (investigators)? Do you prefer that the forensics data mining and subsequent analysis be carried on at the remote engines or brought back to the console to analyze locally and/or take off-line?

Security from the Inside

With the billions of dollars spent protecting our corporate networks from the outside world, when will we begin to pay serious attention to what happens *inside* our network?

The world famous Carnegie Mellon Computer Emergency Response Team (CERT) published an interesting paper entitled "Comparing Insider IT Sabotage and Espionage: A Model-Based Analysis." This 108 page (gulp) study cannot be summarized in its entirety here, but if network security is of interest to you, I highly encourage downloading it. The best part is that it was funded by the DoD, and as such, is freely available.

The study examined "the psychological, technical, organizational, and contextual factors" which contributed to espionage and sabotage against IT. Their research led to the following red flags:

- Saboteurs had personal problems outside the workplace.
- Stressful events such as internal reorganizations increase the likelihood of malicious acts.
- Poor work ethics including performance or tardiness were often observed before and during sabotage.
- Insiders had a tendency to "set things up" such as creating back door accounts.
- Organizations failed to detect or ignored policy rules such as forbidden downloads.
- A lack of access control for both physical locations and on-line computing resources.

The report goes on to provide recommendations for further research to mitigate the risk. One of the repeated themes is to acquire "improved data" related to things like interrelationships, stressful events, policy enforcement vs. technical rule violations, research tools for auditing and monitoring, etc.

While personal problems can be tough to deal with, especially with touchy regulations like HIPPA, we can deploy tools to help us with technical matters.

Tools and What to Look For

Almost any device that collects and stores packets to long term storage (i.e. a disk subsystem of some sort, from a single drive to RAID to SAN) qualifies as a forensics tool. This includes a wide variety of appliances from analyzers to intrusion detection systems.

There are clearly some performance tradeoffs. When comparing tools, such tradeoffs need to be considered depending on the bandwidth (i.e. up to 20 Gbps of raw data over a full duplex 10 Gig link) and how much, if any, pre-filtering is done. Beware that any filtering means that you are *not* getting all the packets off the link that may be critical later.

When it comes to storage, a RAID 0 system with eight drives taking turns writing data in a round robin fashion is substantially faster than writing to a single drive. Don't believe vendor's performance claims – test for yourself in a real network – yours.

Also consider what real time data is provided while collecting such packets other than simply packet and byte counts. Remember, the higher up the stack you go, the more CPU you consume. I've seen appliances that have special ASICs for this type of stat processing, but not combined with a long term data store. Some systems will even perform real-time expert analysis but watch the CPU consumption.

I'll impose an additional requirement that the tool have some way to cull data from multiple (or continuous) packet traces from the disk storage. This data mining capability should be provided by the tool rather than having to write your own code or scripts.

Finally, what happens when the drive fills – is it a FIFO (first in first out) or does it stop? How much control do you have? Can you specify the number and size of the trace data? Hint: Use manageable increments that allow you to read individual files manually if desired (remember, they will have a file time/date for easy retrieval). I recommend keeping them to 512 Kbytes bytes or less. Anything larger will take FOREVER to read back into your analyzer. Believe me, I've tested this with multiple analyzers from different vendors and it can take as long as 15 to 20 minutes just to crunch one file (under 1 Gigabyte in size) with all options turned on (i.e. all stats, expert system, etc.). Think of it like a Word or spreadsheet document – would you create one that's a gigabyte or more size and try to manage it?

Typical data mining features:

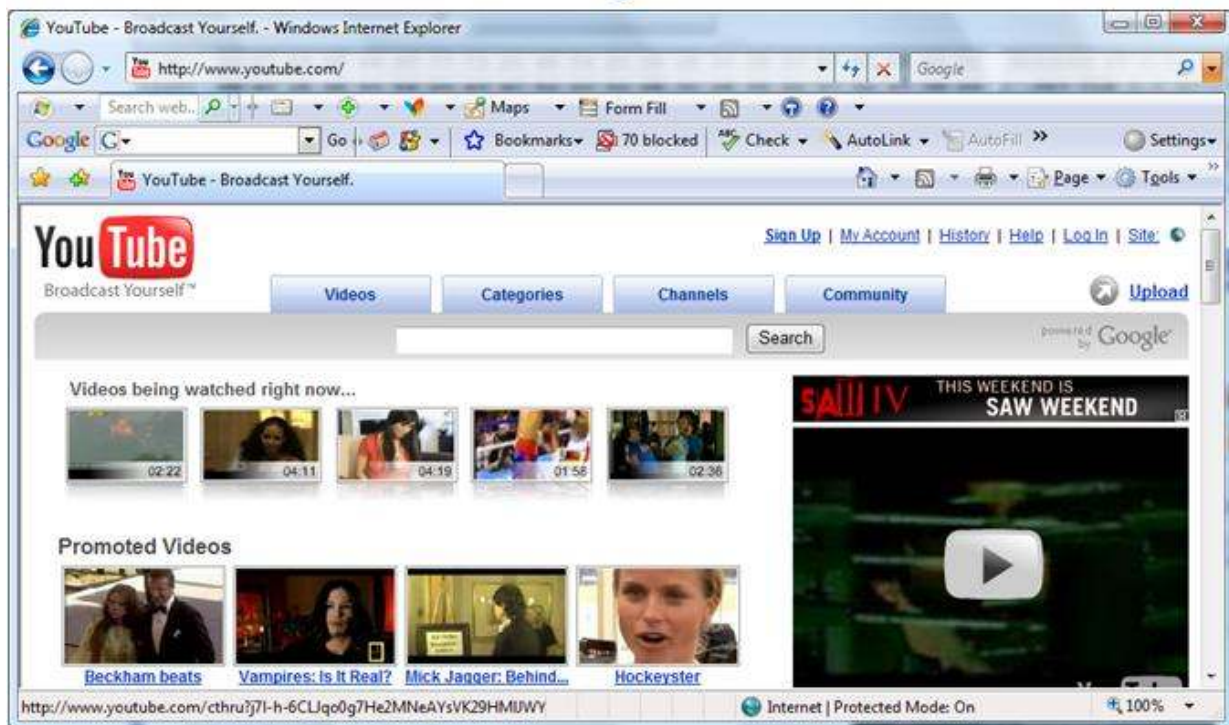
- Specifying the time range (if know) when a suspect event occurred

- Applying a filter, if desired, to a specific user (by physical or IP address) or protocol (such as HTTP)
- Scanning for specific content (sometimes using a text filter without a specific offset)

Once the content is mined, a nice bonus feature is to be able to recreate the original look (or sound) of the content (email and any attachments, web page, database query and results, video, VoIP conversation, etc.) from the packet stream. An example of content recreation is shown in the following figure.

| Packet | Size | Source Logical | Dest. Logical | Protocol | Summary |
|--------|------|----------------|----------------|----------|--------------------------------------|
| 10 | 647 | 192.168.11.6 | 208.65.153.251 | HTTP | C PORT=51649 GET / |
| 11 | 1150 | 207.68.178.12 | 192.168.11.6 | HTTP | R PORT=51647 HTML Data |
| 12 | 64 | 208.65.153.251 | 192.168.11.6 | HTTP | Src= 80, Dst=51648, .A...., S= 76562 |

| | | |
|-------|---|-------------------------|
| 0198: | 20 57 69 6E 64 6F 77 73 20 4E 54 20 36 2E 30 3B 20 53 4C 43 43 31 | Windows NT 6.0; SLCC1 |
| 0220: | 3B 20 2E 4E 45 54 20 43 4C 52 20 32 2E 30 2E 35 30 37 32 37 3B 20 | ; .NET CLR 2.0.50727; |
| 0242: | 4D 65 64 69 61 20 43 65 6E 74 65 72 20 50 43 20 35 2E 30 3B 20 2E | Media Center PC 5.0; . |
| 0264: | 4E 45 54 20 43 4C 52 20 33 2E 30 2E 30 34 35 30 36 3B 20 2E 4E 45 | .NET CLR 3.0.04506; .NE |
| 0286: | 54 20 43 4C 52 20 31 2E 31 2E 34 33 32 32 29 0D 0A 48 6F 73 74 3A | T CLR 1.1.4322)..Host: |
| 0308: | 20 77 77 77 2E 79 6F 75 74 75 62 65 2E 63 6F 6D 0D 0A 43 6F 6E 6E | www.youtube.com..Conn |
| 0330: | 65 63 74 69 6F 6E 3A 20 4B 65 65 70 2D 41 6C 69 76 65 0D 0A 43 6F | ection: Keep-Alive..Co |
| 0352: | 6F 6B 69 65 3A 20 56 49 53 49 54 4F 52 5F 49 4E 46 4F 31 5F 4C 49 | okie: VISITOR_INFO_LI |
| 0374: | 56 45 3D 69 45 69 70 52 33 4F 32 34 65 6B 3B 20 4C 4F 43 41 4C 45 | VE=iEipR3024ek; LOCAL |
| 0386: | 5F 50 52 45 46 45 52 45 4F 43 45 3D 38 36 64 31 64 30 38 65 65 66 | DDREFERENCE=8641d308e5 |



From Packets to Content

There are a number of tools that satisfy many of the aforementioned requirements. If you're in the market, take a look at Network Instruments, Network General (NetScout), Niksun, Packet Forensics (could the name be any more direct?), Solera Networks, and WildPackets to name a few. All provide various turn-key hardware configurations. NI and WP will also sell you only the software, allowing you to deploy using your own hardware. Pricing runs the gamut, so be sure to compare and evaluate carefully.

Detecting Anomalous Activity

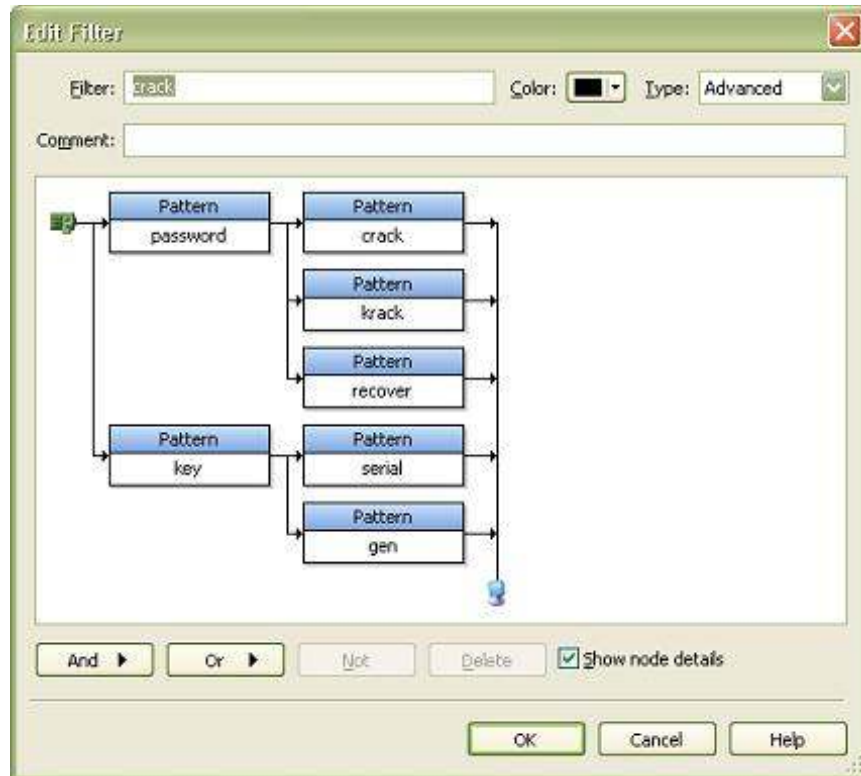
I'll now discuss a practical forensics technique that illustrates a means for detecting a type of "technical" activity as mentioned in the CERT study: "Organizations failed to detect or ignored policy rules such as forbidden downloads". Earlier, I referred to this as red flag #5. For illustrative purposes, we are going to check for a user searching for and downloading password cracking tools or illegal software key generator utilities. If detected, there's a high probability that the user will actually use such a tool.

One of the things I love about advanced network analysis and forensics tools is the ability to easily apply custom triggers and triggers to find stuff inside packets. Triggers are a special case of filters that allow us to start a packet capture and/or send an alert if we are capturing in real time.

In this case, we are going to take advantage of a special capability of the analyzer to search for an arbitrary word or pattern anywhere in a packet. This "sliding pattern match" does not require any prior knowledge of where the pattern might be. To optimize performance a bit, we are going to start at offset 54 inside the packet, which tells the analyzer to start at the beginning of the TCP payload. No sense in wasting CPU cycles looking for application data in the data link, IP, and TCP headers.

The following screenshot (from WildPackets OmniPeek) shows such a filter, using a combination of AND along with OR conditions. We start by looking for the words 'password' or 'key'. If password is found, it must match (AND) the words or patterns 'crack' OR 'krack' OR 'recover'. Thus phrases like password crack, password krack, password recover, password recovery (or the reverse) will be found. Likewise for 'key serial' <number>, 'key generator', 'keygen', and so on. I've also told the analyzer to ignore case.

Test it by searching on-line with your favorite search engine (which will trigger a hit right there) or going to any website containing such tools. The filter and/or trigger will hit immediately – even if the tool is named something else, purveyors of such tools love to fill up their underlying web site code with key words to gain search engine positioning.



A Forensics Search/Filter/Trigger

This is the tip of the iceberg when it comes to real time or forensics data mining. Such tools can be invaluable in assisting you to effectively combat potential sabotage and espionage in your network.

Conclusion

There are many tools and techniques for network forensics. Make a checklist of your specific requirements starting with the type of forensics desired (compliance, hacking, loss of proprietary data, etc.) then features desired (throughput requirements, data mining requirements, access to distributed appliances, and so on) and finally, evaluate the tools themselves.

J. Scott Haugdahl is Founder and CTO of Bitcricket, a company that specializes in on-site network analysis consulting, hands-on analysis training, network troubleshooting, and software tools for the network engineer. For more information, please peruse the Bitcricket web site at www.bitcricket.com.