NetworkMiner:
Network Forensic Analysis Tool

By Russ McRee – ISSA member, Puget Sound (Seattle), WA, USA chapter

Prerequisites

Winpcap
Windows XP recommended,
but works well for static analysis on Vista.

Similar Projects

Wireshark[1]
Driftnet[2]
Cain & Abel[3]

Introduction

I had the pleasure of participating in the 20th annual FIRST Conference in Vancouver, B.C., as a speaker and attendee, just before beginning the process of writing this month's column. Given that FIRST is the Forum of Incident Response and Security Teams, I was hopeful that I would discover some relevant topics for *toolsmith* and I wasn't disappointed. That said, the most influential conversation I enjoyed at the conference was a chat with Richard Bejtlich (*The Tao of Network Security Monitoring*) and Raffael Marty (*Applied Security Visualization*) where the discussion's focus was largely on new ways to interpret network data captures. Having long embraced network security monitoring and more recently security data visualization, I went searching for new tools that bring a different element to traffic analysis. Enter NetworkMiner 0.84,[4] the strong results of Erik Hjelmvik's development efforts.

Erik was kind enough to provide us with a number of details regarding NetworkMiner. For instant gratification though, you can find almost everything you need on the NetworkMiner wiki.[5]

Erik's goal is for NetworkMiner to become a full blown Network Forensic Analysis Tool (NFAT), available for free as an open source application. NetworkMiner is focused on the extraction of relevant events and information about hosts and users on a network, and providing that information in an intuitive user interface. Further focus is on analyzing and parsing PCAP files rather than on performing live sniffing with NetworkMiner. Simply, there are several other applications that are better at sniffing packets like Wireshark or tcpdump. Erik goes so far as to not recommend the use of a Windows OS if hoping to perform packet sniffing properly on a high speed network. That being said, NetworkMiner can be used to sniff data, either by using WinPcap or by using Raw Sockets. NetworkMiner is an excellent compliment to network security monitoring systems as a tool for attack investigation, and it can also be used to conduct behavior analysis of a compromised machine, potential rogue host, or malicious user.

The following are some of the things planned for future implementation:

- A proper reporting tool
- Faster parsing of large PCAP files
- Implement even more protocols
- Statistical methods to do protocol identification (protocol fingerprinting) of a TCP session or UDP data, or identifying the correct protocol based on the TCP/UDP packet content rather than port number, eliminating non-standard port identification failures. See Bejtlich's discussion regarding PIPI (Port Independent Protocol Identification) and Dynamic Application-Layer Protocol Analysis.[6]

Erik maintains a list[7] of more minor features he's planning to add to NetworkMiner.

If you'd like to get a look at new upcoming versions of NetworkMiner, as well as have access to a large amount of PCAP files, apply for a membership to the private NetworkMiner beta testers mailing list.[8]

Before detailing a bit of NetworkMiner usage, allow me to highlight its capabilities as a forensic data collector:

- OS Fingerprinting
  - TCP SYN and SYN+ACK using OS fingerprinting databases from p0f and Ettercap
  - DHCP via the Satori OS fingerprinting database from FingerBank
  - The MAC-vendor list from Nmap

1 http://wireshark.org.
2 http://ex-parrot.com/~chris/driftnet.
3 http://www.oxid.it/cain.html.
4 http://sourceforge.net/project/showfiles.php?group_id=189429.
5 http://networkminer.wiki.sourceforge.net/NetworkMiner.
6 http://taosecurity.blogspot.com/2006/09/port-independent-protocol.html.
7 http://sourceforge.net/tracker/?group_id=189429&atid=929293.
8 https://lists.sourceforge.net/lists/listinfo/networkminer-betatesters.

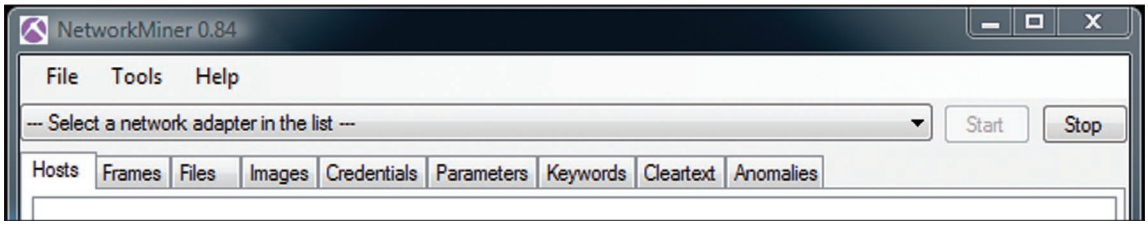

Figure 1 – NetworkMiner UI

- File extraction via PCAP parsing with supported protocols including FTP, HTTP, and SMB
- Credentials grabbing from supported protocols
- Clear text parsing inclusive of keyword search functionality
- Wireless sniffing and parsing with AirPcap adapters[9]

Enough said; let's play.

Using NetworkMiner

One negative, and I'm more prone to blaming Windows Vista than anything else, but NetworkMiner on Windows Vista gets pretty hosed up looking for `wpcap.dll`, even if you *Run as Administrator*. This issue really only affects conducting captures from the Vista PC which, like Erik, I recommend against. You can use Raw Sockets but reliability suffers. Instead, grab your PCAPs from a *nix host and conduct static analysis on the Windows machine running NetworkMiner.

Let me go on record as saying this: NetworkMiner certainly lives up to its name. The UI is incredibly simple, but there are some subtleties that, once uncovered, will leave you smiling at the realization of the tool's usefulness. The UI will offer you a network adapter drop-down menu and nine tabs giving you scads (it has to be a real word; I saw it on a BYU wiki) of data on hosts, frames, files, images, credentials, parameters, keywords, cleartext, and anomalies (Figure 1).

As is always the case with our lab tests for *toolsmith* content, I sniffed the wire during both regular traffic patterns and intentional malware outbreaks. For NetworkMiner I let loose a quaint little IRC-based Trojan with a binary referred to as `camda.exe`, tucked neatly in a URL suggested to me by my SPAM filter, promising interesting pictures. My SPAM filter's always timely with suggestions of this nature; the pictures were awful but the malware was great! The resulting PCAP taken during analysis allowed me to validate the prowess of NetworkMiner in a distinctive fashion. For your own testing, I'm offering up `camda.pcap` via email request only.[10] **WARNING:** The domain name you will see in the *Hosts* tab, while inactive, is both adult and hostile, with registrar originations in Turkey. Further, you will be reconstructing actual malware if you use this PCAP while testing NetworkMiner.

Starting with the *Files* view, we can immediately determine that we received a little present named `ddos.exe` from our friends at <content removed for the children in the audience>.info. Hmm…I wonder what ddos.exe does. NetworkMiner rebuilt it from the HTTP GET request and wrote `ddos.exe.octet-stream` to the assembled files directory in the default NetworkMiner hierarchy. If you feed the `.exe.octet-stream` to VirusTotal.com you will find that the payload was identified 24 out of 33 times (Figure 2).[11]

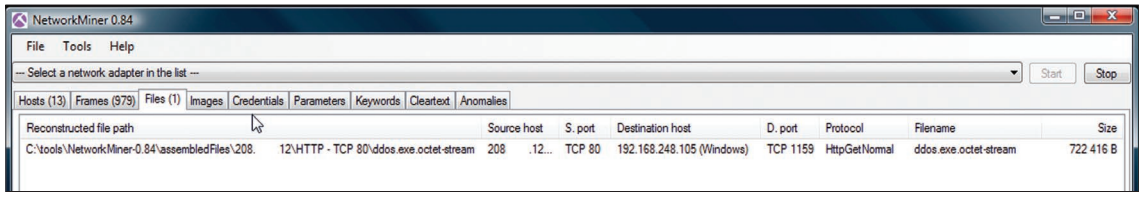

Figure 2 – NetworkMiner reconstructs ddos.exe

NetworkMiner will grab certificates for you in a similar fashion, and is even useful for building media files from streams.

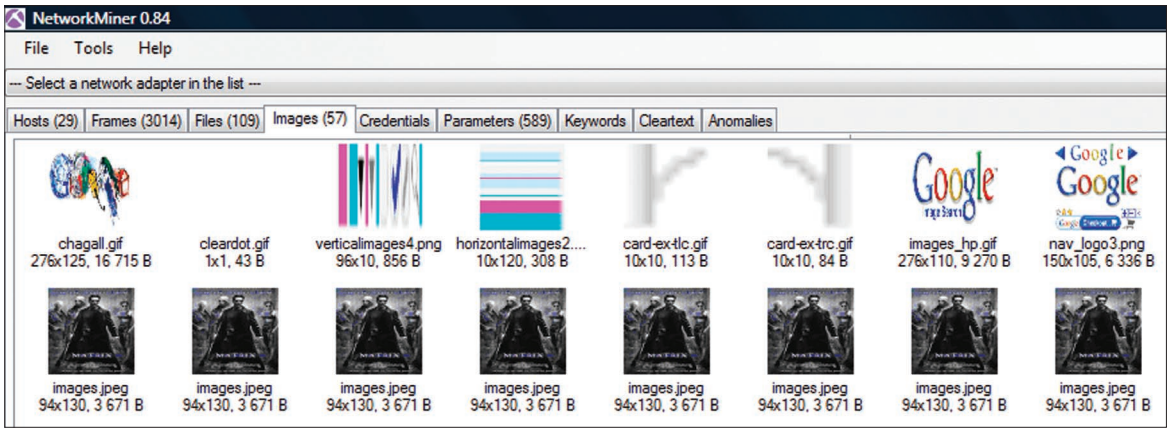

Figure 3 – The Matrix (I can't help it)

9 http://networkminer.wiki.sourceforge.net/NetworkMiner.

10 holisticinfosec@gmail.com.

11 http://www.virustotal.com/analisis/4e1dad92775925b844b397a0be133eae.

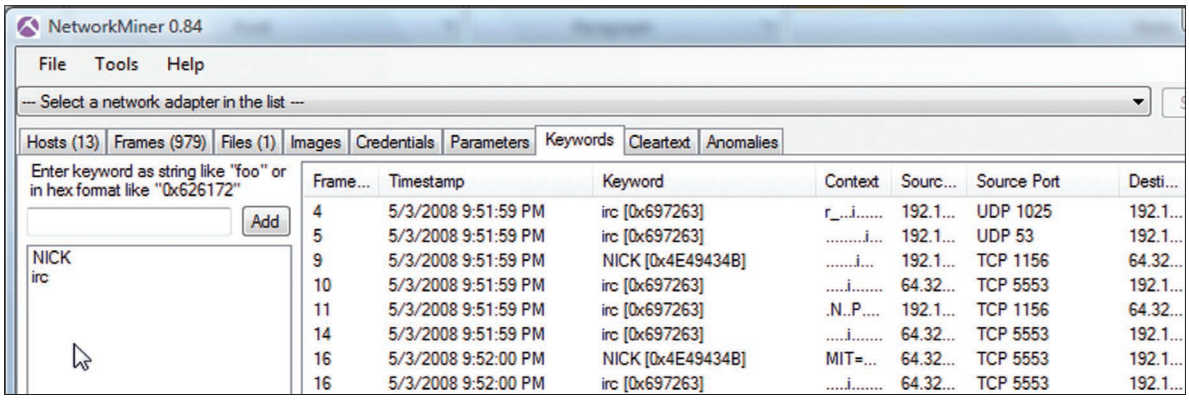

Figure 4 – Typical IRC chatter

You'll likely find the *Images* tab interesting as well. I utilized a generic capture for this to demonstrate the functionality; the images feature is quite similar to Driftnet. I conducted a Google image search for images from *The Matrix* (Figure 3). I know, really original.

The *Keywords* feature is great for typical forensic discovery. Using *Tools-Reset Capture Data* before re-opening `camda.pcap`, I added the keywords *irc* and *NICK*, and then opened the capture (Figure 4).

The bonus of the keyword search is the fact that it points me to the associated frame, seen in (yep, you guessed it) the Frames tab (Figure 5).

Frames kindly offers up the fact that the IRC chatter is occurring with `64.x.y.7` (hope it's not yours). Taking a quick peek in the Hosts tab will confirm our suspicion; `64.x.y.7` is indeed an IRC server.

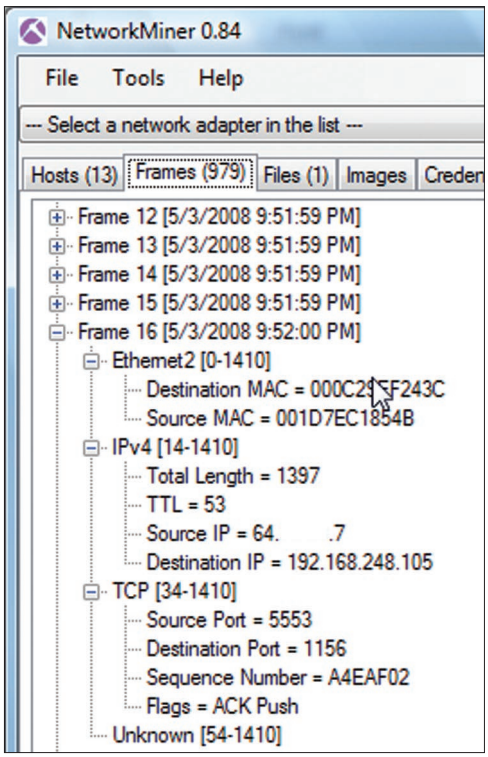

Figure 5 – Where's the bad guy?

The *Credentials* tab should be obvious; if you're passing credentials in the clear, your goodies are up for grabs. Don't forget to check on *Anomalies* for errant behavior, and the *Cleartext* tab will definitely give up some likely data to narrow down via Keywords. The *Parameter* tabs will even satisfy the web crawler in you, identifying exactly what you imagined: input variable/parameters and the strings passed to them.

Benefits and drawbacks

The benefits to NetworkMiner are endless. The almost instantaneous forensic discovery the tool allows simply speaks for itself. If I force myself to find a drawback, it might be the fact that you'll likely uncover too much information and may have to review your privacy policies before proceeding. It is a young tool, and there are numerous functionality enhancements pending, but suffice it to say that if Erik brings them all to light, I'm willing to go right out on a limb here and nominate it for Fyodor's Top 100 Network Security Tools.[12]

In conclusion

I said a few months ago that there are certain tools information security practitioners, no matter their specialty, should have in their toolkits. NetworkMiner is one of those tools. It's easy to use; you'll be underway in no time; and the resulting data will be of assistance no matter the forensic circumstance. I'm certain you will find immediate use for it; if not for you, for someone on your team. Keep an eye on the project website and sign up for the beta program. Enjoy! Cheers, until next month…

Acknowledgments

Erik Hjelmvik, for all the insight, and a great tool.

About the Author

Russ McRee, GCIH, GCFA, CISSP, is a security analyst working in the Seattle area. As an advocate of a holistic approach to information security, Russ' website is holisticinfosec.org. Contact him at russ@holisticinfosec.org.

12 http://sectools.org.