A portable network forensic evidence collector

by Bruce J. Nikkel

nikkel@digitalforensics.ch

Originally published by Elsevier in Digital Investigation
The International Journal of Digital Forensics and Incident Response
Vol. 3, No. 3 (10.1016/j.diin.2006.08.012)

October 29, 2006

Abstract

A small portable network forensic evidence collection device is presented which is built using inexpensive embedded hardware and open source software. The device offers several modes of operation for different live network evidence collection scenarios involving single network nodes. This includes the use of promiscuous packet capturing to enhance evidence collection from remote network sources, such as websites or other remote services. It operates at the link layer allowing the device to be transparently inserted inline between a network node and the rest of a network. It is simple to deploy, requiring no reconfiguration of the node or surrounding network infrastructure. The device can be preconfigured in the forensics lab, and deployment delegated to staff not specifically trained in forensics. Details of the architecture, construction and operation are described. Special attention is given to information security aspects of live network evidence collection.

Keywords: Network forensics, Live network evidence, Live network acquisition, Live network forensics, NFAT, PNFEC

1 Introduction

The device presented here is a proof-of-concept network forensic evidence collector. The work done in building this experimental prototype explores a number of issues in the area of network forensic evidence collection. Some new ideas such as the use of a portable network forensics collector with multiple distinct modes of operation is shown. The separation of data collection from data analysis is demonstrated, and enables the delegation of certain duties to staff external to the forensics lab. The concern for information security issues is addressed, both in the design and in the device operation.

1.1 Background

Much work has been done in developing the area of network forensic collection. Very early on, intrusion detection systems (IDS) were found to be a useful source of evidence[1]. The use of network forensic analysis tool (NFAT) systems to capture network traffic for evidence purposes has been developed in the industry, and is well documented[2]. Both these areas rely heavily on promiscuous packet capturing technology[1].

Enterprise networks are large and complex, and network traffic only exists for a brief moment during transit. Capturing this traffic for the purpose of evidence requires collectors to be at the right place, at the right time. This requirement has led to an increased focus on permanently deployed, widely and strategically distributed collectors.[3][4].

Collecting evidence from remote network services (often outside the jurisdiction of the investigator) is also an important part of network forensics. It has been shown[5] that evidence collection from remote network services (web servers, ftp servers, etc.) is fundamentally similar to evidence promiscuously captured from a network segment. As a result of this generalization, the same technologies for packet capturing can then be used to aid in evidence collection of remote network sources. This concept is demonstrated here as one of the three modes of operation used by the network forensic collection device.

Finally, recent work to improve evidence acquisition from live network sources[6] is used as a basis

[1] Often implemented using pcap libraries

for designing the architecture and operating principles of this network forensic collection device.

1.2 Motivation and scope

The technologies and concepts used in building this collection device are not new (Ethernet bridging, promiscuous packet sniffing, etc.). However, the design goals and intended usage satisfy a current need in the area of network forensic evidence collection. This device fills a gap between basic portable network packet analyzers and full featured NFAT systems. Also, the use of promiscuous packet capturing as a standard component to assist and enhance evidence collection from remote network services (www/ftp servers, DNS/Whois servers, peer-to-peer activity, etc.) is shown.

Portable network packet sniffers such Network General's Sniffer Portable, or portable analyzers from Fluke Networks, have existed for many years. These portable devices are designed more for short term collection, troubleshooting, and network traffic analysis. They are not specifically designed for network forensic evidence collection.

NFAT devices such as Sandstorm's NetIntercept, CA's eTrust Network Forensics (formerly SilentRunner) and Niksun's NetDetector, have also been on the market for some time. While these are intended for network forensic evidence collection, they are relatively expensive, and designed more for permanent deployment on a network. These devices are very feature rich in areas of forensic analysis, and offer far more than simple packet collection.

The device presented here offers more than basic portable packet sniffing, in that it is specifically designed with digital forensics and network evidence collection in mind. The device is not intended to compete with full featured, permanently deployed NFAT systems. The focus is limited to acquisition, not analysis, and the intended use is for temporary situations involving single network nodes.

The network forensic evidence collector presented (Figure 1) is portable, inexpensive, and built using existing open source software. It can be used on demand, and driven by specific cases or incidents. Deployment can be done very quickly by staff not specifically trained in digital forensics, and without reconfiguration of the surrounding network. The device has several intended modes of operation which are described in detail.

Figure 1: The portable network forensic evidence collector

In addition to issues of digital forensics, very close attention is given to aspects of information security. Packet sniffing devices provide easy extraction of information which may otherwise be protected by access controls at the server or application level. This introduces an information security risk. These concerns are addressed in the design, construction, and intended operation of the device.

1.3 Design goals and decisions

There were a number of design goals and deliberate decisions made in building this portable network forensic evidence collector (abbreviated PNFEC for rest of this paper). The design decisions were influenced by such factors as device performance, disk capacity, operational simplicity, information security, and previously published requirements involving the acquisition of live network evidence.

The design decisions and goals outline the scope of the device and its intended operation. Some of these include:

- focus on traffic collection between a network and a single network node
- separation of collection from analysis (PNFEC focuses strictly on collection)
- specific intended modes of operation (investigator mode, server mode, user mode)
- facilitate deployment by IT staff not specifically trained in forensics
- pay close attention to information security issues (including compliance with policy and regulations)
- remote administrative access is optional and not required for deployment
- compact and portable for use in small cramped wiring closets or racks, and for easy shipment to remote locations

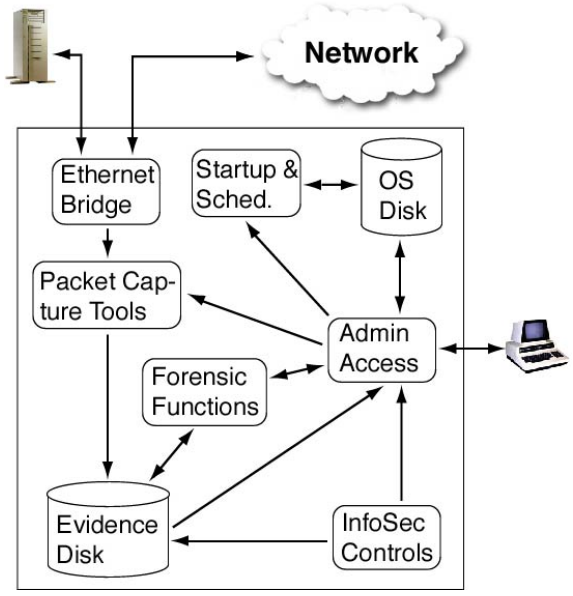

Figure 2: Functional architecture diagram

- rapid deployment requiring little or no planning, no reconfiguration, and no disruption of surrounding infrastructure
- transparent, invisible/stealthy inline operation
- inexpensive and feasible to build

2 Architecture and design

2.1 Device components

A high level functional architecture diagram of the PNFEC is shown in figure 2. The Ethernet bridge transparently passes packets unaltered between a server or workstation and the network. This traffic is promiscuously captured with various pcap based capture tools and stored on a large evidence hard disk. The operating system, additional software, configuration files, and investigator activity logs are stored on a separate disk (compact flash). Administrative access allows an investigator to control and configure various aspects of the device. This includes device startup, scheduling, configuration of capturing filters, forensic functions such as evidence preservation, and transferring evidence off the device. Controls to ensure the security of evidence collected are also a key component of the PNFEC. They include strong authentication and access control of the administration interfaces, logging and accounting of all investigator activity, and the possibility of encrypting captured data for storage and during network transfer.

2.2 Network topology

The device employs link layer bridging between a single node and the rest of the network, making it completely transparent after insertion. There are a number of benefits from this characteristic. The PNFEC becomes very easy to use and operate (plug and play), both for investigators, and for IT staff who are assisting with deployment. There is no significant planning or assistance required by network administrators. In emergency or time critical situations, the device can be quickly inserted into a network, and later removed, with no configuration or topology changes required.

Since the device forwards Ethernet frames unaltered between the bridging interfaces, it is completely invisible to the network layers above. This stealthy operation is useful when collecting evidence of malicious activity such as hacking attacks, fraudulent activity, or abuse. Attackers do not see the device and are unaware that their activities are being captured.

When dealing with switched networks or VLANs, a number of potential issues arise when attempting to promiscuously capture traffic. Typically, a dedicated or *mirrored* port[2] needs to be configured for a packet capturing device to collect all traffic on a switched segment. In addition, VLAN configurations and restrictive MAC address based port security can also cause difficulty when adding a packet sniffing device to a network. These problems are avoided by using inline bridged packet sniffing as described here.

The device is carefully configured to prevent injection of traffic onto the attached network, and to prevent modification of traffic during transit. The PNFEC is a proper Ethernet bridge[7], and forwards all Ethernet frames without alteration (unlike routers which may fragment packets and modify header fields such as the TTL or hop count, etc.)

2.3 Information security

There are a number of issues involving information security which need to be addressed when capturing network traffic on enterprise networks. A useful guide discussing forensics with regard to corporate information security has been published by the IAAC (Information Assurance Advisory Council)[8]. Promiscuous network collection devices such as the PNFEC allow the extraction of information which

[2]Some vendors have their own acronyms for these ports, such as SPAN (Cisco) or RAP (3Com)

might otherwise be protected by access controls at the server or application layer. On networks containing sensitive or critical data (finance, healthcare, government, military, etc.) policies and regulations may be in place which influence the design and operation of network capturing capability.

As network traffic is captured from the network, it is stored on a local hard disk inside the PNFEC. Under certain circumstances, there may be requirements for this captured data to be adequately secured. This can be accomplished by using an encrypted filesystem on the evidence hard disk[3]. This protects the captured information in the event that the device is lost or stolen (always an issue with compact portable devices) or if the disk is replaced or discarded.

If the network management port is configured to allow remote access, there are issues involving both the authentication of the investigator, and the confidentiality of transferred data. To ensure strong authentication, the device uses SSL client certificates (X.509) for https access, and ssh keys (RSA or DSA) for secure shell and sftp access. Network protocols such as ssh and https also ensure the protection of data during transfer[4].

After collecting relevant data and securely transferring it off the device, it may be desirable to wipe the evidence hard disk. The device has provisions for securely wiping the entire evidence disk to ensure proper destruction of old data. As this is a time consuming process and may not always be needed, the disk can also be quickly reformatted without secure wiping.

To avoid capture of sensitive or confidential information unrelated to a case, the PNFEC has a number of possibilities to control the filtering of captured data. Most of this filtering functionality is already built into tools such as tcpdump, and uses such criteria as MAC addresses, IP addresses, or protocol port numbers. In addition, the placement of the device is deliberate and intended to aid in restricting the data collected. Traffic capture is done only on the interface connected to the network node (the "NIC" interface). This ensures that only traffic involving that particular node is captured, and not unrelated traffic on the "HUB" interface connected to the rest of the network.

[3]This requires a key or passphrase to be entered on startup

[4]IPSEC can also be used to achieve equivalent security. This has not been thoroughly tested on the PNFEC, but could be used in the future

To facilitate audit and review of investigator activity, all investigator access and activity performed on the PNFEC is logged. Custom PNFEC scripts and wrappers also create log entries describing each event. These logs all reside on the compact flash drive, not on the evidence disk. Wiping or reformatting the evidence disk will not destroy the record of investigator activity.

If the device is in operation on untrusted networks, care must be taken to ensure the device itself is resistant to attack and compromise. The PNFEC device is hardened and stripped of unnecessary services. Only software required for operation is installed. The configuration of network services, daemons, and the OS kernel, is defensive and explicit (ie. default parameters are reviewed).

2.4 Evidence preservation

A standard component of forensic evidence collection is the preservation of evidence using a cryptographic hash or cryptographic signature. The PNFEC includes various utilities for creating hashes (md5, sha1, etc.), and for evidence signing using tools such as PGP or Openssl. The collected network traffic is stored as individual tcpdump files, each with a unique filename[5]. The use of cryptographic tools ensures the integrity of the files is preserved, and can be verified at a later date.

A script is used to create hashes of the acquired tcpdump files. The cryptographic hash is reported and also logged together with the rest of the investigator activity on the separate compact flash disk. If the hash of a particular capture file needs to be verified at a later point in time, the log entry can be used for verification.

3 Construction

This section briefly covers the technical construction details of the device.

3.1 Hardware details

The PNFEC is roughly the size and weight of a small textbook[6]. It is built using a net4801 embedded mainboard (Figure 3) from Soekris Engineering, which has a 586 class CPU and 128MB of RAM. The board has three 100Mb/s Ethernet interfaces, two RS232 serial ports (one internal, one external), and a USB 1.1 interface. The board also

[5]Based on a timestamp from the capture start time

[6]approximate weight and size: 1.1kg / 2.5lbs and 3.5cm x 14cm x 25.5cm / 1.4in x 6in x 10in

has a standard 3.3 volt PCI slot, as well as a mini-PCI type III slot.

Figure 3: Internal layout

The PNFEC has two storage devices. A 1GB CompactFlash module holds the operating system, additional software, configuration, and all logs of investigator/administrator activity. A standard 2.5 inch hard disk is mounted in the empty half of the enclosure[7]. The harddisk shown is 100GB and used only for collected evidence/data storage. The entire harddisk may be securely wiped without affecting the operational functionality of the device.

Figure 4: Status LEDs

The front (back?) of the device has four status LEDs (Figure 4), one of which (red) is user programmable. The PNFEC blinks various operational status codes every minute to show the device is functioning and collecting data. This is useful when the device is deployed standalone, without a terminal or management connectivity.

Two of the three Ethernet ports (Figure 6) are used to place the device inline between a server or workstation and the rest of the network (a transparent bridge). The remaining Ethernet port, and the external serial port are used to managed the device. These do not need to be configured or connected for the device to function. If a device is preconfigured in a forensics lab, only the two bridging interfaces need to be connected during deployment. Data collection should automatically start either on powerup, or when scheduled.

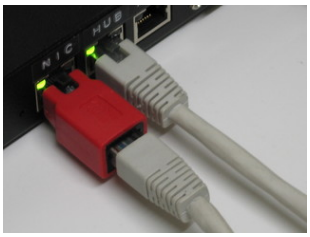

Figure 5: Ethernet crossover adapter

The Ethernet ports do not have MDI/MDI-X autodetection. When connecting to a network node such as a PC or Server, an Ethernet crossover cable or crossover adapter (Figure 5) may be needed if the node does not support autodection[8].

3.2 Operating system

The current operating system used in building and testing the PNFEC device is OpenBSD 3.8. This operating system was chosen for partly for its stability, security, and robustness[9], and partly due to the author's familiarity with the OS. OpenBSD supports transparent bridging of Ethernet frames between two network interfaces. It has extensive security and cryptography support, and the OpenBSD code base is audited to reduce the number of vulnerabilities. This focus on security makes it an ideal platform for building attack resistant network forensic tools.

The default OpenBSD installation is already quite minimal. The goal is to have only network services and daemons running which are explicitly required for PNFEC operation. For remote access and evidence file transfer, the only services required are secure shell (TCP port 22), and/or a secure webserver (TCP port 443). No other network services need to be running.

Much of the functionality required to run an inline evidence collection device is already included by default with OpenBSD. For example:

- secure access (ssh, sftp, https)
- packet capture (tcpdump)

[7] Soekris supplies mounting brackets to mount the HDD above the mainboard, but heat is better distributed when mounted as shown here.

[8] Newer computers usually support MDI/MDI-X autodetection.

[9] This is not meant to imply that other Operating Systems such as Linux or other BSDs are lacking in these areas.

- encrypted filesystem capability (vnd, vnconfig)
- tools for evidence preservation (md5, sha1, rmd160, openssl)
- transparent bridging support (brconfig)
- disk wiping and reformatting tools (dd, newfs, disklabel)

Figure 6: External ports and interfaces

Configuration of the three network interfaces is critical in setting up the PNFEC device. The first interface is reserved as a management port and can have its own IP address configured for secure access over a network. The other two interfaces are configured in the OS as a bridge (using brconfig) and sit between a network node and the rest of the network. One interface labeled "HUB" is connected to the network (via a hub or switch), the other labeled "NIC" is connected to the workstation or server. The OS kernel will forward Ethernet frames unaltered between these two interfaces.

The two bridge interfaces (where capture takes place) do not have TCP/IP configured, and therefore have no IP addresses with which to communicate. They are "up" at the link layer only. This makes it resistant to probes and other attempted connectivity, and also prevents PNFEC-generated traffic (such as DNS or ICMP) from being injected onto the monitored network.

No admin traffic is sent out or received by the interfaces used to capture network traffic. The admin interface is deliberately excluded from the bridge configuration, and is out-of-band in relation to the bridging interfaces. Because of this separation, the admin interface can be connected to an entirely different IP subnet than the network segment being monitored. DNS queries caused by the packet capturing tools can be done through the admin interface if desired, or disabled [10].

The default OS configuration is modified to provide increased logging, and stronger authentication. The use of ssl client certificates as well as ssh authorized keys is used instead of traditional password authentication. The PNFEC system is configured to use this strong authentication for administrative access and for the secure transfer of evidence from the device.

3.3 Additional software

Some additional software is needed to perform various functions required to operate and administer the device. Various tools for trouble shooting and testing are added, as well as some additional packet capturing and pcap management tools such as tcpflow and tcpslice. The standard BSD ports and packaging system is used for installing new software. This packaging system provides easy download and installation of many useful open source software packages.

In addition to the open source software and standard Unix functionality used to construct the PNFEC device, there are a handful of custom wrapper scripts for simplifying various activities required for PNFEC administration and operation.

3.4 Further information

More information about the embedded hardware used can be found at soekris.com. More information about the OpenBSD operating system can be found at openbsd.org. Information about the pcap interface and tcpdump can be found at tcpdump.org.

Since the technology use to build this device is based on open source software (OpenBSD, tcpdump, etc.) and standard networking configurations (Ethernet bridging), it is fully possible to implement this using other compatible hardware. Other operating systems such as Linux or other BSD derivatives could also be used, provided they offer transparent bridging capability, and the software needed for traffic capturing, scheduling, security, etc.

[10] By modifying /etc/resolv.conf or using "tcpdump -n" for example

4 Operation

4.1 Modes of operation

The PNFEC device has three intended modes of operation. There is no technical difference between these modes, and they differ only in the device placement and the source of evidence being collected. The modes of operation are:

- Investigator mode (collection of evidence on remote network services by a single investigator)
- Server mode (suspected attacks, intrusions, or abusive activity directed against a single machine)
- User mode (abuse, suspected criminal activity, or policy violation originating from a single machine)

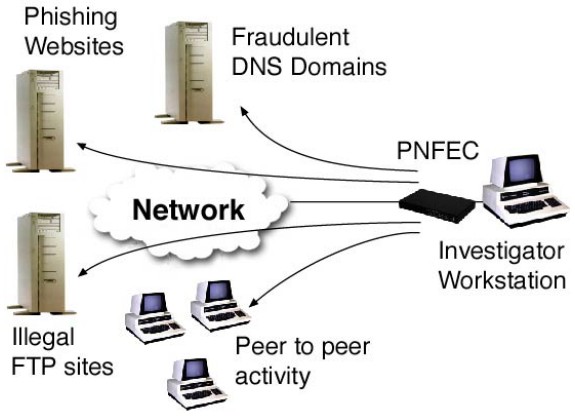

Figure 7: Investigator mode of operation

Investigator mode involves an investigator capturing activity during an investigation or evidence collection of remote network services (Figure 7). These services may include remote ftp or web sites, peer-to-peer networks, DNS/Whois data collection, etc. The PNFEC is inserted between the investigator's own workstation and the network, collecting all traffic generated by the investigator.

Server mode involves capturing activity during an attack, intrusion, or abuse directed against a single server or other network node (Figure 8). The PNFEC is inserted between the server being misused and the network. All traffic coming to/from the server can be collected.

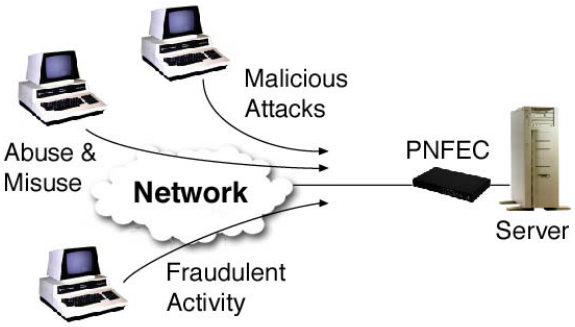

Figure 8: Server mode of operation

User mode involves capturing network activity generated by a single end user or end user machine (Figure 9). The PNFEC is inserted between the end user/machine and the network, and collects evidence of abuse, criminal activity or policy violation originating from the user/machine . In such cases, the PNFEC can be discreetly deployed in a wiring closet at the Ethernet hub/switch or patch panel.

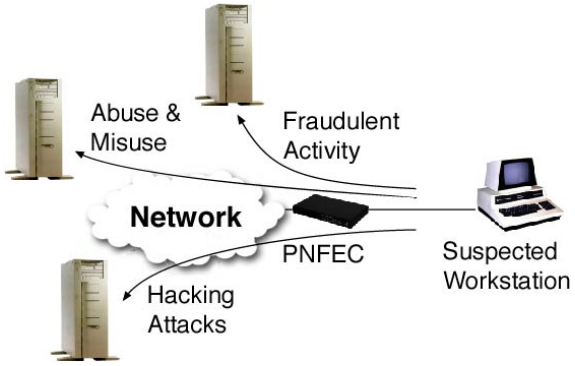

Figure 9: User mode of operation

4.2 Parameters of collection

The parameters for collecting network traffic are quite flexible, and can be adapted to fit a particular situation.

The device can be preconfigured in the lab to automatically start collecting network traffic on power-up[11]. This is useful if the insertion of the device has been delegated to other staff, and a certain degree of plug-and-play functionality is desired. The device can also be configured to start and stop collection at various scheduled times. For example, if suspected malicious activity is originating from a particular

[11] Repeated reboots do not overwrite previously collected data.

workstation only during weekends, the device can be set to collect from Friday afternoon to Monday morning, without collecting data during the week.

The tcpdump software used in the device has a number of filtering possibilities. Specific MAC addresses or IP addresses can be explicitly captured or ignored as needed. Various protocols can be captured or ignored as necessary. For example, consider an investigator on a forensic lab network, collecting evidence from remote websites and other Internet servers. Traffic activity between the investigator and local lab servers (local fileserver access, internal email, etc.) is not needed for collection. The PNFEC can be configured to capture only Ethernet frames with the MAC address of the Internet router. This not only reduces the amount of traffic collected, but also protects unrelated (and possibly sensitive) information from being captured and stored on the evidence disk.

There are a number of different tools used to implement capturing, and to control the scheduling of particular capture activity. Tcpdump is included with the operating system by default, and is the primary tool used in the PNFEC device. Other tools such as tcpflow or ssldump are also quite useful (though these can also operate directly on tcpdump files, making them somewhat redundant). Standard Unix functionality such as *cron*, *at*, and various startup scripts are used for scheduling and automatically starting or stopping the capture of network traffic.

4.3 Points of insertion

The PNFEC device is simple to insert between a node and a network. It can be inserted in seconds without user noticeable network disruption[12]. There are typically two points where insertion into the network takes place.

For *investigator mode* and *server mode* the device can be inserted right next to the server or investigator workstation (Figure 10). With the PNFEC powered on, simply plug the network cable into the "HUB" port on the PNFEC, and reconnect the server/workstation to the "NIC" port (using a crossover cable or crossover adapter if needed). The server or workstation and the surrounding network infrastructure should continue functioning as before, without any noticeable difference.

[12]The network equipment will notice brief Ethernet linkdown/linkup activity which may generate an alert or log entry.

For *user mode* the device is used to collect evidence of illegal, or abusive activity originating from a single network node. In such cases, the use of a PNFEC device should not be obvious to the suspect, and insertion should be done discreetly, away from the suspect PC. To accomplish this, the device insertion can take place in a wiring closet or rack, between the cabling patchpanel and the hub or switch (Figure 11). With the PNFEC powered on, simply plug the suspects patch cable into the "NIC" port (using a crossover cable or crossover adapter if needed), and reconnect the suspect's hub/switch port to the "HUB" port on the PNFEC. The PC and surrounding network infrastructure should continue functioning as before, and the suspect should not notice any disruption.

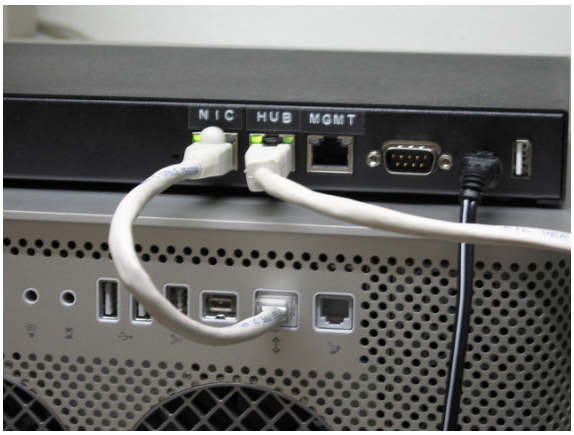

Figure 10: Device inline next to a server or workstation

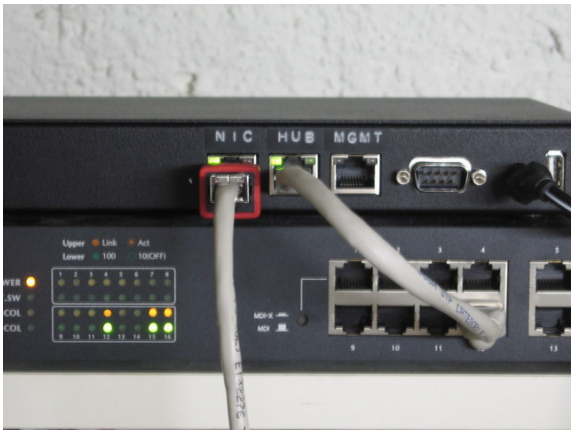

Figure 11: Device inline between a patchpanel and hub/switch

4.4 Administrative access

Administrative access is used for a number of activities. These may be carried out in the forensic lab before and after deployment (when the physical device is in the lab), or they may be done during device operation. Administrative access is achieved over a network using the separate "MGMT" port on the PNFEC, or by connecting a terminal to the serial console. It is possible to attach a modem or mobile phone to the serial port for remote access to the device. However, many organizations would consider this a very serious breach of network security.

Before deploying the device for data collection, admin access allows the investigator to configure the filters and tools used. The power-up behavior and scheduling of capture jobs can also be configured, and the device can be tested before sending it out for deployment.

During operation, admin access is not specifically required. The two admin access methods (network and serial) are provided if needed, and can be used for monitoring, reconfiguration, and online transfer of evidence from the device to the lab. Admin access may also be used to set a passphrase for encrypting the evidence hard disk, or to cleanly shutdown the device[13].

After evidence collection is complete, a number of other administrative activities can be carried out (in the lab, or remotely over the network). These include disabling active or scheduled capturing activity, secure transfer of evidence off the device, and wiping or reformatting the evidence drive in preparation for future use.

Other administration activities may include device maintenance, firmware and software updates, adding/removing investigative users, and auditing of investigator activity logs.

At the moment there is no comfortable user interface for PNFEC administration. Configuration and admin activity is done by hand using command line tools and scripts. However, a web-based admin interface, as well as a menu-based console interface are under development.

[13]When using encrypted filesystems, clean shutdowns are recommended over an abrupt power-off.

5 Conclusion

5.1 Advantages

There are a number of advantages the PNFEC offers for network forensic evidence collection.

It is compact and portable making it useful for crowded and cramped wiring closets and racks. It is also invisible on the network, causing no disruption, and requiring no reconfiguration of the surrounding infrastructure. Since it is only capturing traffic from a single node, it can be used in a switched or VLAN environment without mirrored ports or additional port configuration on the switching device.

It can be preconfigured in a forensic lab, with the insertion or deployment delegated out to IT staff or network administrators. This is useful to reduce travel by forensic investigators (the preconfigured device can be shipped), and also makes better use of investigator time and resources.

The device is built using inexpensive hardware and uses free open source software. This allows the device to be carefully scrutinized and verified for correct operation. The device construction is feasible, and can be easily built by an engineer with hardware and Unix experience.

The device does not modify or inject traffic as it acts as a bridge at the link layer. Ethernet frames are forwarded unchanged through the device (MAC address, hop-count, fragmentation, MTU, etc. remain untouched). It supports the collection of multiple protocols, including IPv4, IPv6, IPX, NetBios, or any other Ethernet based network traffic.

5.2 Problems and other issues

A number of problems have arisen during the development of this device. Some of these issues simply require more time to solve, while others are more fundamental issues which have resulted in trade-offs for desirable features.

Since the PNFEC is inline between a network node and the network, availability becomes a critical issue. If the device would suddenly shutdown, malfunction, or stop forwarding Ethernet frames between interfaces, the network node would completely lose connectivity to the network. While this risk exists, it should be mentioned that the prototype device has been running continuously in a

testing/research lab for many months with no availability problems due to heat, hardware or OS instability[14]. Also, OpenBSD has support for the spanning-tree protocol, allowing for bridge redundancy using multiple devices. This has not been tested with the PNFEC device, but could be an option for evidence collection in highly available environments.

Under heavy network loads, and when capturing entire Ethernet frames, the device will drop packets. This can be significant, especially when transferring large amounts of data between two 100Mb/s nodes on the same Ethernet segment. To a certain extent, device optimization may reduce this problem.

Even though the Ethernet bridging and forwarding of Ethernet frames through the PNFEC is done within the OS kernel, it will still have a certain effect on network performance, adding a degree of latency. Further testing still needs to be done to determine the precise impact on performance.

The device is only intended to be used on demand, and is not permanently deployed on a network. This approach limits the amount of data collected, and any network activity taking place before deployment is obviously lost.

It is expected that the device will often be simply powered on during insertion, and powered off during removal. Since this does not result in a clean shutdown, there is a risk of data loss. Some work has been done to reduce this risk, however, further testing in this area still needs to be done.

One of the desirable features of this device is the ability to delegate the insertion of the device to IT staff not specifically trained in forensics. While great care has been taken to simplify the insertion process, cabling issues (especially when crossed cabling is involved) can still be confusing.

A desired feature is the ability to use the device without any separate management channel (network or serial). While this massively simplifies and speeds up the insertion process, it also requires blindly running the device with no possibility of remote connectivity. Any useful activity requiring a network connection via the management port (NTP, SNMP, monitoring, remote troubleshooting or tuning, etc.) are not possible.

[14]There were availability interruptions due to testing, experimenting, and development work by the author, but the hardware and OS remained stable.

Time synchronization without a configured Ethernet management port for NTP traffic can be accomplished using an external GPS or DCF77 clock connected to the serial or USB ports. However, when placed in a data center, server room, or wiring closet, the radio signal reception may be severely limited.

5.3 Future work

While the device is currently in a usable state, work in several areas is expected to continue. Administrative functions are currently done largely on the Unix command line. Work is being done to create comfortable menu-based (via terminal/ssh) and web-based (via https) administration interfaces. Additional optimization work also needs to be done to reduce packet loss when collecting entire frames on fast LANs.

The concepts and tools discussed here are readily transferable to other hardware platforms and operating systems. Future work may include construction using Linux and other BSD systems, as well as with other embedded hardware.

As of this writing, most of the testing and development involved the use of tcpdump/pcap based tools. This may be expanded in the future to include data collection from pf (packet filter) logging, kernel/snmp statistics, snort logs, and other interesting data.

Issues of evidence disk capacity need further development. Depending on the collection parameters used, the captured traffic may require compression (provided via gzip). In addition, the precise handling of a full disk needs to be further developed.

References

[1] Peter Sommer, Intrusion detection systems as evidence, Computer Networks 31, 1999

[2] Eoghan Casey, Network traffic as a source of evidence: tool strengths, weaknesses, and future needs, Digital Investigation Vol 1 No 1, 2004

[3] Yongping Tang, Thomas E. Daniels, A Simple Framework for Distributed Forensics, Proceedings of the 25th IEEE International Conference on Distributed Computing Systems Workshops, 2005

[4] Ren Wei, A Framework of Distributed Agent-based Network Forensics System, DFRWS 2004

[5] Bruce Nikkel, Generalizing sources of live network evidence, Digital Investigation Vol. 2 No 3, 2005

[6] Bruce Nikkel, Improving evidence acquisition from live network sources, Digital Investigation Vol. 3 No 2, 2005

[7] IEEE Standard 802.1D, Media Access Control (MAC) bridges. Technical Report ISO/IEC 10038, ISO/IEC, 1993

[8] Peter Sommer, Directors and Corporate Advisors' Guide to Digital Investigations and Evidence, Information Assurance Advisory Council (IAAC), September 2005

Bruce Nikkel is working on a PhD in network forensics at Cranfield University. He also works for Risk Control at UBS AG in the IT investigation and forensics team. He has worked for the Bank's IT Security and Risk departments since 1997. He holds an MSc. in Enterprise Network Management and is CISSP and Encase EnCE certified. He has been an IT/Network professional since 1990.