Building a Forensically Capable Network Infrastructure

Nik Alleyne

Building a Forensically Capable Network Infrastructure

GIAC (GCIA) Gold Certification

Author: Nik Alleyne, nikalleyne@gmail.com

Advisor: Richard Carbone

Accepted: June 19, 2016

Abstract

The number of computer related security incidents continue to grow yearly, resulting in the need for ensuring network infrastructures are built to be forensically capable. During the period January 2011 to December 2015, the number of reported computer security incidents grew over this four-year period from 1,281 to 3,930. Similar to the increased number of reported computer security incidents, was the increased number of exposed records. During this same period, the number of exposed records jumped from 413 million to 736 million, with 2013 and 2014 having over 2 billion records exposed. Some challenges with becoming forensically capable, relates to understanding the business needs, identifying the people to support that need and ultimately the technology or tools to support business needs.

1. Introduction

As the number of computer related security incidents continue to grow yearly, the need for ensuring network infrastructures are built to be forensically capable becomes even more relevant. During the period January 2011 to December 2015, the number of reported computer security incidents grew from 1,281 to 3,930. Similar to the increased number of reported computer security incidents, was the increased number of exposed records. During this same period, the number of exposed records jumped from 413 million in 2011 to 736 million. While this may seem minimal, for the years 2013 and 2014, the number of combined exposed records totaled 2.2 billion (Risk Based Security, 2016).

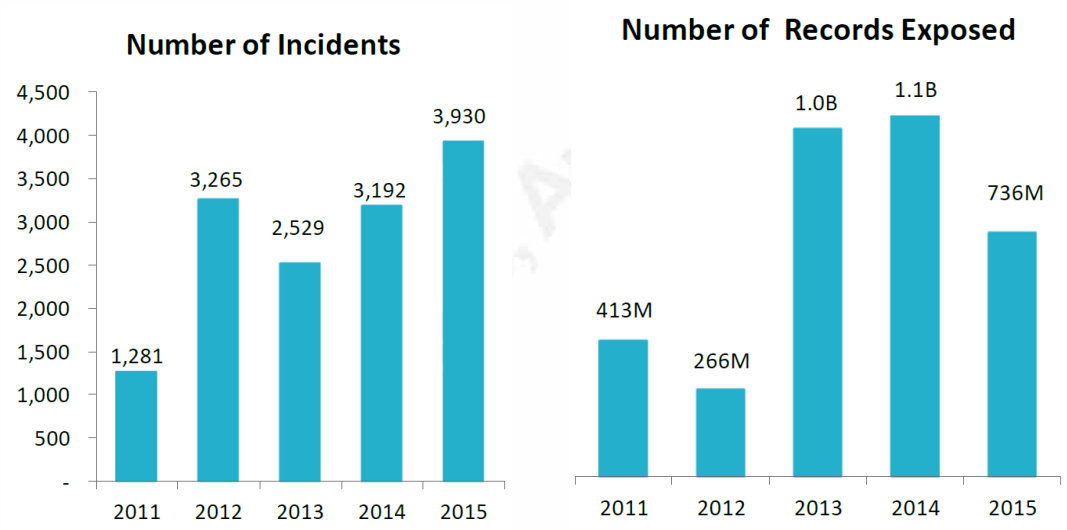

Figure 1: Number of Incidents and Number of Records (Risk Based Security, 2016).

A more detailed review of the compromises over this period shows that 77.7% of all incidents had an external source and 64.6% were caused by hacking. Of the exposed records, 58.7% of these exposures were caused by hacking. However, overall hacking and web based activities accounted for 89.3% of all exposed records (Risk Based Security, 2016). This 89.3% further reinforces a heavy reliance on Internet activities as evidenced by more businesses moving to the cloud.

Looking at specific examples, four hacking incidents resulted in 237.8 million exposed records. Similarly, one database misconfiguration resulted in 191 million

exposed records. There were also 46 incidents, which had an exposure of more than 1 million records (Risk Based Security, 2016).

Looking specifically at Canadian organizations, while they may not be in the news as often as other nations, this does not mean they are immune to this type of activity. Canada was number three in terms of security incidents reported by country in 2015, behind the United States (US) at 1,593 and the United Kingdom (UK) at 246. In reference to exposed records, in 2015 Canada came in at number four with an average of approximately 400,000 exposed records per incident.

With an even deeper focus on Canada and specifically looking at the Canadian Revenue Agency (CRA), the Heartbleed vulnerability, which was unpatched, resulted in the compromise of over 900 Social Insurance Numbers (SINs) being exposed (CBC News, 2014). While this may seem like a small number when looking at a Canadian population of approximately 36 million (worldometers.info, n.d.), the reality is that even the compromise of one SIN can be catastrophic.

As we head into what seems like an interesting future, we see that the trend of exposure and compromise continues. Thus far, for 2016 there has been a reported 1,456 computer security incidents. These have resulted in losses of approximately 870 million records (Risk Based Security, 2016). Considering this and the fact that it is more than likely one's network may be soon compromised if it has not already been, the need for a forensically capable network infrastructure becomes even more apparent.

2. Building a Forensically Capable Network

2.1. What is meant by forensically capable?

A forensically capable network allows a forensic investigator, network security analyst, intrusion analyst, etc., to retrace the steps of any potentially identified security issues. This allows security personnel to not only fix the current issue but also prevent and mitigate it. These issues may include but are not limited to identification of fraud, policy violations, security incidents, auditing, forensic investigations, inappropriate usage, etc.

It is important to understand that this data can also be used for operational purposes such as establishing baselines, identifying operational efficiencies or deficiencies. However, the objective of this paper is strictly from a security perspective. Thus, our focus is on forensic investigations, intrusions, policy violations, etc.

2.2. Becoming forensically capable

In becoming forensically capable, network infrastructures need to be planned and designed from the perspective of capturing and retaining data that can be leveraged for forensic investigations. Once the network infrastructure has been planned, designed and ultimately implemented, the data that is captured must be appropriate and relevant to what is being secured and the type of investigations that need to be completed. Data should also be readily available and in a manner that is easily understandable. Most importantly, when recruiting, personnel who can add context to data in that environment become more relevant.

Unfortunately, the same data that is captured to assist with investigations can be used against its owners. As a result, protecting data confidentiality, availability and integrity (CIA) becomes critical. By controlling access to data through appropriate firewall rules, access control list (ACL), etc., one can achieve proper CIA. However, controlling access is not just about physical but also logical separation, which includes separation of duties. While Analysts may need read only (RO) access to data for their analyses, engineers may need administrative access to update or upgrade the devices that capture and forward data.

While it is typical to see network deployments with one logging destination, e.g. Security Events Information Management (SEIM) or log aggregator, leveraging multiple logging destinations can be quite beneficial. In an effort to ensure that the tools used to make the network infrastructure forensically capable are effective, sufficient intelligence should be added to those tools to ensure that proper context and relevance are gained. This intelligence can be as simple as the operating system (OS) running on an end system. Additionally, knowledge of the applications running on devices helps to answer the “who, what, when, where, why and how.” This level of intelligence can be added

through asset data, rules or any other such facilities available in the tools to assist with maintaining this data.

Time configuration, is one of the most critical components in becoming forensically capable. It can be the differentiator between knowing an event occurred today, verses it occurring yesterday. It is also what can help to ensure successful tracking of events across devices. Most importantly, it results in consistency across multiple devices, multiple technologies, multiple sites, multiple time zones, etc. When building a forensically capable network, consideration must be given to using an accurate, reliable and dedicated timeserver.

Another challenge in becoming forensically capable is the decision to use software-based agents. Such agents have their advantages and disadvantages. However, understanding the challenge they pose is important and ultimately must be the driver for the chosen direction.

3. Prioritizing

3.1. Giving priority to the business needs

In becoming forensically capable, business needs take precedence over everything else. It is the area that should be given priority and focus. As technology professionals, it is easy to be caught up with the latest tools and technology. However, without the business there would be no need for the tools and technology.

Additionally, in prioritizing, the people supporting the effort of becoming forensically capable must be next. Once completed, it is important that processes are then developed to support the effort. This is then followed by identifying the technology needed to support the effort.

3.2. Identifying the sources of forensically capable data

In identifying forensically capable data within the network infrastructure, it is easy to become inundated. Data logs from web servers, routers, switches, Intrusion Prevention Systems/Intrusion Detection Systems (IPS/IDS), mail servers, authentication systems, etc., are but a few sources of log data. Deciding which level of logging to use

for this data is a critical step. From a Syslog perspective, logging at the Informational level is more relevant than logging at the debug level as informational messages may contain information such as IP address, usernames, etc. Alternatively, Debug messages are typically more voluminous and while it may contain relevant information; this may be more than what is required from a security perspective. Figure 2 shows an example Syslog configuration file and some of the available options, while Figure 3 shows sample output of Informational messages and Figure 4 sample Debug messages.

```
# $FreeBSD$
#
#       Spaces ARE valid field separators in this file. However,
#       other *nix-like systems still insist on using tabs as field
#       separators. If you are sharing this file between systems, you
#       may want to use only tabs as field separators here.
#       Consult the syslog.conf(5) manpage.
*.err;kern.warning;auth.notice;mail.crit                /dev/console
*.notice;authpriv.none;kern.debug;lpr.info;mail.crit;news.err   /var/log/messages
security.*                                      /var/log/security
auth.info;authpriv.info                         /var/log/auth.log
mail.info                                       /var/log/maillog
lpr.info                                        /var/log/lpd-errs
ftp.info                                        /var/log/xferlog
cron.*                                          /var/log/cron
!-devd
*.=debug                                        /var/log/debug.log
*.emerg                                         *
# uncomment this to log all writes to /dev/console to /var/log/console.log
#console.info                                   /var/log/console.log
# uncomment this to enable logging of all log messages to /var/log/all.log
# touch /var/log/all.log and chmod it to mode 600 before it will work
#*.*                                            /var/log/all.log
# uncomment this to enable logging to a remote loghost named loghost
#*.*                                            @loghost
# uncomment these if you're running inn
# news.crit                                     /var/log/news/news.crit
# news.err                                      /var/log/news/news.err
# news.notice                                   /var/log/news/news.notice
# Uncomment this if you wish to see messages produced by devd
# !devd
# *.>=info
!ppp
*.*                                             /var/log/ppp.log
!*
```

Figure 2: Sample Syslog config file found in /etc/sysog.conf (Zeising, n.d.)

- %ASA-6-106012: Deny IP from IP_address to IP_address, IP options hex.
- %ASA-6-106015: Deny TCP (no connection) from IP_address/port to IP_address/port flags tcp_flags on interface interface_name.
- %ASA-6-106025: Failed to determine the security context for the packet:sourceVlan:source_address dest_address source_port dest_port protocol
- %ASA-6-106026: Failed to determine the security context for the packet:sourceVlan:source_address dest_address source_port dest_port protocol
- %ASA-6-106100: access-list acl_ID {permitted | denied | est-allowed} protocol interface_name/source_address(source_port)(idfw_user, sg_info) interface_name/dest_address(dest_port) (idfw_user, sg_info) hit-cnt number ({first hit | number-second interval})

Figure 3: Sample Syslog Informational message (Cisco, n.d.).

- %ASA-7-108006: Detected ESMTP size violation from src_ifc:sip|sport to dest_ifc:dip|dport;declared size is: decl_size, actual size is act_size.
- %ASA-7-109014: A non-Telnet connection was denied to the configured virtual Telnet IP address.
- %ASA-7-109021: Uauth null proxy error
- %ASA-7-111009: User user executed cmd:string
- %ASA-7-113028: Extraction of username from VPN client certificate has string. [Request num]
- %ASA-7-199019: syslog

Figure 4: Sample Debug messages (Cisco, n.d.).

3.3. Choosing between open source and commercial tools

The challenge with choosing tools is often making a choice between free and open source software (FOSS) and commercial off-the-shelf software (COTS). However, the situation may be such that an organization does not accommodate open source tools. Additionally, the question about whether existing FOSS tools address the business needs, must be asked. Even if existing FOSS tools do address the business needs, consideration still has to be made about whether the necessary human resources are in place to ensure the tools and technology can be supported on an ongoing basis. Of greater importance when selecting FOSS tools and technologies, is whether the organization provides training for these technologies. Often, efforts have been made to train and retain employees to support these tools and technologies but these efforts may still result in a high turnover rate as information security personnel are highly sought after. Considering

the JP Morgan breach, which resulted in the loss of gigabytes of data and customer information, high staff turnover, was consider as one of the primary reasons for this compromise. Top executives who had intimate knowledge of JPs' environment and its system were no longer with the company and this affected the response to the compromise (Son & Riley, 2014).

QRadar	Tcpdump	ArcSight
McAfee	Wireshark	SolarWinds
Splunk	Gigamon	Cisco
Niksun	Fortinet	Baracuda
SyslogNG	Netscout	DLP

Table 1: Sample of tools and vendors which may help in becoming forensically capable.

From a COTS and FOSS tools perspective, as shown in Table 1, there are numerous tools that may assist in implementing a forensically capable network. From a full packet capture perspective, tools such as *tcpdump*, *tshark* and *snort* can be used. The obvious concern with full packet capture is that storage may become problematic, when storing large amounts of data for longer periods. This may be needed for compliance or regulatory needs. In the absence of full packet capture, flow data can be extremely useful and is often the next best thing. A number of tools exist to assist with flow analysis. Some of these tools include *System for Internet Level Knowledge (SILK)*, *NFDump/NFSen, nTop, flow-tools, argus*, etc.

Event collection is just as important as full packet capture and flow. Similar to the collection of packets and flows, various tools can be used for collecting and correlating log events. Alient Vault's *OSSIM*, *Enterprise Log Search and Archive (ELSA), GrayLog, SyslogNG, OpenSOC, Elastic+Logstash+Kibana, OSSEC, Prelude-LML, Splunk*[1] are but some of the available tools. Bandwidth recording is also of significant benefit to becoming forensically capable. For example, if a network typically runs at 10Mbps

[1] There is a 500MB per day limit for the free version.

during the nights when a small number of persons are online, then a spike for a continued period should be an immediate cause for concern.

However, even after all considerations are given to FOSS, it may come down to the decision that it makes sense to go with COTS instead. COTS software choice is often easier to use, deploy and support personnel are readily available, as is training. This does not eliminate the concern about staff turnover but does mitigate the risk, as replacing personnel comfortable with COTS is often easier than FOSS.

3.4. Capturing the important data

In Section 3.3, it was discussed that forensically capable data can be obtained from numerous sources. However, with all of this data, it is important to understand that from a security perspective, efforts must be made to capture the data that is considered important to the organization. One of the most significant and impactful sources of forensic data, is authentication logs. However, there is a choice to be made about logging successful and failed authentication events. For business reasons it may be a choice of one or the other. However, efforts should always be made towards collecting both, as authentication failures can help identify potential reconnaissance or other potential security issues.

While obtaining authentication events is critical, obtaining events to access for critical data is just as critical and should always be obtained. This should be for both failed and successful attempts, as similarly to authentication events, failed attempts can be an indicator of other potential and more severe issues. Capturing of malware events from anti-malware tools is also critical. However, there should be no need to capture all malware events; instead capture the ones that are not cleaned or blocked. This may seem controversial, as it may be felt that all malware events should be captured. However, reports can be run on the specific anti-malware tool to determine statistics, trends and patterns that provide additional learnings for malware that has been blocked, quarantined or treated in some other way.

Firewall events, are yet another crucial piece of log data that should be captured, as ultimately, much of the traffic traversing the network will pass through a firewall. Once again, in some instances, it may be desired to log only permitted/allowed or

denied/dropped firewall traffic. However, it is beneficial to have both permitted and denied attempts as denied attempts are always indicative of either a potential problem, some type of reconnaissance or something worthy of investigating. For example, one firewall deny for a single host within 24 hours is likely nothing to worry about. However, 24 denies to one or more destinations with the same source address within several minutes is likely worrisome and thus may require an investigation.

Elevation of privileges or an attempt to gain higher-level privileges, from a lower privileged account, should be also tracked. Ultimately, one of the primary objectives of an attacker, is to allow him or her to gain administrative access to a compromised computer, in order to further access the network. In the case of environments such as Microsoft Windows Active Directory, this may be an attempt to become a member of critical groups such and Domain Admins, Enterprise Admins, etc. In the case of Linux, it may be an attempt to "sudo" or "su" to the root account.

Previously, it was mentioned that hacking and web based activities were responsible for 89.3% of all exposed records in 2015 (Risk Based Security, 2016). This means that where proxies are used within a network infrastructure, its data should be leveraged to not only determine statistics about sites users have visited but also to identify suspicious sites. Data from proxies can assist in pointing directly to links that can then be correlated with other data, including source and destination IPs, domain names, users, etc.

3.5. Protecting Confidentiality, Integrity and Availability (CIA) of data

Protecting the confidentiality, integrity and availability of logs can be challenging, as it depends on multiple factors, including protocols, processes, people, technologies, etc. Protecting the data when using UDP Syslog can be a challenge, as this data is sent using clear text. This can result in a number of primary threats, such as masquerading, modification and disclosure. Additionally, there are secondary threats related to message stream modification. This is where an attacker may delete, replay or even alter Syslog message delivery sequences and content (Miao, Ma, & Salowey, 2009).

However, all is not lost, as Syslog can be transported securely using Transport Layer Security (TLS). This in turn mitigates the risk previously identified. Additionally,

while Syslog typically uses UDP port 514, TLS Syslog uses TCP port 6514 (syslog-tls), which is dedicated for this transport type (Miao, Ma, & Salowey, 2009). In addition, TLS can also be used to secure other clear text protocols that may be used for transporting data in forensically capable network infrastructure.

Another option for moving clear text data is to leverage Secure Shell (SSH) tunnels. These tunnels can be used to transport unencrypted traffic, through a network via an encrypted channel (Chamith, 2012) and is done by leveraging the port-forwarding feature within SSH. While not the final alternative, another option is to leverage the Simple Network Management Protocol (SNMP) version 3. SNMPv3 ensures message integrity by verifying that each received message has not been modified in transmission through the network. Importantly, SNMPv3 can ensure that messages are not being replayed, while ensuring that the contents of each received message, is protected from disclosure during transmission (WebNMS, 2009).

Data availability is a critical component of being forensically capable. However, the challenge with this results in the question of how much data can be kept and for how long. For example, compliance and regulatory concerns may require data be kept for 3 months or 3 years. However, in the event of a security incident, will this data be enough to trace back the real start date and time of the event? The data which is available (or lack thereof) can have a significant impact on measuring the time to detection or the alternatively considered “detection deficit” (Verizon, 2015), which is a critical measurement in order to understand how and when a specific activity started and ultimately its duration.

For devices that are responsible for forwarding their data to one or more alternate destinations, ensuring these devices are monitored to track when they go offline or stop logging is an important step. However, if the devices are known to go offline during certain periods then there is no need to track the devices during those periods. Having a network specifically for management purposes also assists in making a network infrastructure forensically capable as it results in the segregation of production network data from that of management data. Ultimately, ensuring one can trust the data, i.e. logs, flows, etc., is an extremely important step in becoming forensically capable.

Author Name, email@address

3.6. Controlling access to data

Controlling access to data can be done in multiple ways. For devices that are responsible for forwarding any type of data (logs, packets, etc.,), to one or more remote destinations, it is important that these devices are monitored to determine when they go offline. Similarly, ensuring device configuration is only modified by authorized personnel is critically important. Unauthorized configuration may result in forwarding being disabled or logs, flows, etc., being rerouted to alternate destinations.

Managing access may seem like a challenge, as various persons may require different access to the same data. This is not necessarily a bad thing and implies that users must be given the appropriate privileges based on their roles. For example, an analyst may need read only access to data to be able to query it. However, an engineer may need access for configuration purposes, performing of upgrades, etc. Another important measure is to limit access based on subnets, IPs, VLANs, etc. For example, if all security personnel are in a subnet called "security," then only allows that subnet access. Similarly, if all access is done from a specific "jump server," then only that jump server should be allowed to access the destination. Restricting access is critical, as it can be the difference between data whose integrity remains intact and one that is compromised.

3.7. Leverage intelligence data

Ultimately, the data used in an environment must be leveraged with all available intelligence. For the purpose of this paper, intelligence here relates to what is known about the environment. Some SIEMs, log aggregation or security tools allows the use of asset data. This asset data can be used to build intelligence by specifying information that is known about a remote host such as operating system (OS), known application, known and expected listening ports, etc.

Another excellent example of intelligence is understanding the network baseline and its bandwidth utilization. For example, if a network runs at 100 Mbps between 9 am and 5 pm, why is there suddenly a spike around 4:30 pm and 5 pm with bandwidth utilization of 200 Mbps? Understanding the network infrastructure and the role of devices in this infrastructure, can be the difference between knowing whether this is a new remote backup device that has been brought online and functioning during this time or

whether a compromise has occurred. This same concept applies to other services. Understanding the time at which services are executed and monitored can be critical to understanding the security implications.

Not all networks allow clear text protocols, thus it should be clear whether this type of activity is expected on a network. If it is expected, then it should be clear also which devices are allowed to use clear text protocols. On the flipside, is encrypted traffic allowed? Although asking this question may seem strange, it is a question that needs to be asked. Many networks allow for encrypted traffic but many of them also allow that traffic to be decrypted so that it can be monitored.

3.8. Understanding the importance of time

In a forensically capable network infrastructure, there is nothing more important than time. It is the mechanism through which timelines are associated with events so as to draw conclusions towards the "when" it happened. In ensuring that time is both correct and consistent, efforts should be made to leverage Network Time Protocol (NTP) rather than local system clocks. More importantly, devices should be configured to use no less than two NTP servers, should the primary fail.

Considering many operating system vendors have adopted Kerberos as the centerpiece of enterprise level interoperability (Walla, 2000), efforts should be made to leverage Kerberos time requirements. By default, Kerberos allows for a maximum time skew of 5 minutes between a client and a server. This mean, if the time on the local server and the time on the client differ by more than 5 minutes, then Kerberos will return a KRB_AP_ERR_SKEW error (Kohl, 1993). If using a Microsoft environment consisting of Active Directory, then configuring devices to obtain their time from the Primary Domain Controller (PDC) Emulator is the recommended step. However, first the Active Directory Primary Domain Controller (PDC) Emulator should be configured to obtain its time, from one or more remote source(s), which is then used to feed the network. Figure 5 below provides an example of how time can be configured in a Microsoft Windows environment leveraging the "w32tm.exe" utility.

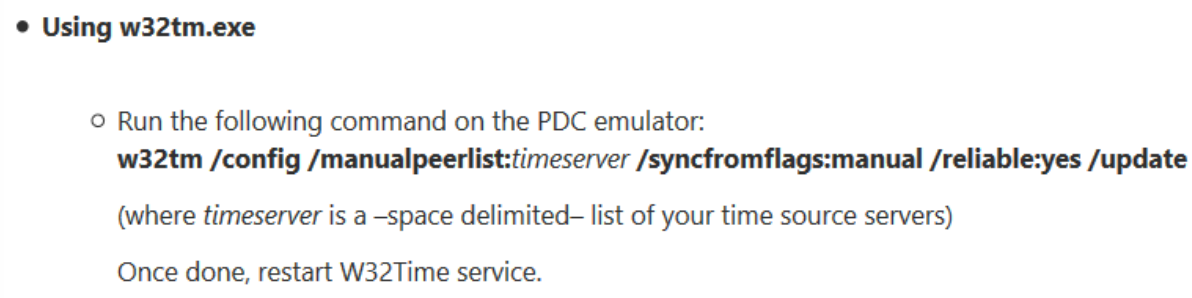

Figure 5: Configuring NTP on a Windows PDC emulator using w32tm.exe (AMHIL, 2013).

Alternatively, if using a UNIX based platform, configuration of NTP can be done by editing configuration file "/etc/ntpd.conf." Figure 6 shows a snapshot of a UNIX based NTP configuration that leverages four NTP servers for time synchronization.

```
server 0.rhel.pool.ntp.org iburst
server 1.rhel.pool.ntp.org iburst
server 2.rhel.pool.ntp.org iburst
server 3.rhel.pool.ntp.org iburst
```

Figure 6: RedHat default ntpd.conf file with 4 NTP servers (RedHat, n.d.).

While it is one thing to configure devices to use NTP rather than a local time server, it is even more important to decide whether to use a local time zone rather than Coordinate Universal Time (UTC). The decision can be affected by a factor such as whether the business is local or global. However, it would always be best to configure devices to leverage UTC as it does not change for Daylight Saving Time (DST) and is the basis for civil time and time zones worldwide. However, it is important to understand that no country or territory uses UTC for official local time (timeanddate, n.d.). Local applications can then present the time to users based on their local time zone.

3.9. Agent vs Agentless

A significant decision in building a forensically capable network is whether to leverage software-based agents or whether to go agentless. Agents are typically small pieces of software installed on a device to address a specific issue. These are typically used for collecting and forwarding logs. For example, Splunk Universal Forwarder in Figure 7 can be used to retrieve logs from a device running Microsoft Windows. Each option brings their own benefits but must be carefully considered when becoming forensically capable.

Agentless networks reduce the attack surface by eliminating the need for additional software that may introduce one or more potential vulnerabilities. It also reduces the need to manage additional software that may become cumbersome over time. If considering a deployment of only one agent, this should not be a problem to manage and maintain. However, for deployments that have tens, hundreds or even thousands of agents, managing them can be a daunting task. Overall, agentless networks are easier to manage but may require additional credentials to gain access to administrative shares, etc.

While agentless obviously brings benefits, so does using agents. One immediate benefit of agents is that they provide features that ensure confidentiality, integrity and availability of the transmitted data. Additionally, agents can ensure reliable, secure data collection from remote sources and can scale to tens of thousands of remote systems, collecting a large number of bytes without a significant impact on network performance. Agents may also allow for throttling, buffering, data compression, transport over various available ports, scripted inputs, centralized management and SSL security (splunk, n.d.). Ultimately, if presented with the choice of going with agents, it is much easier to manage a forensically capable network with less software to support.

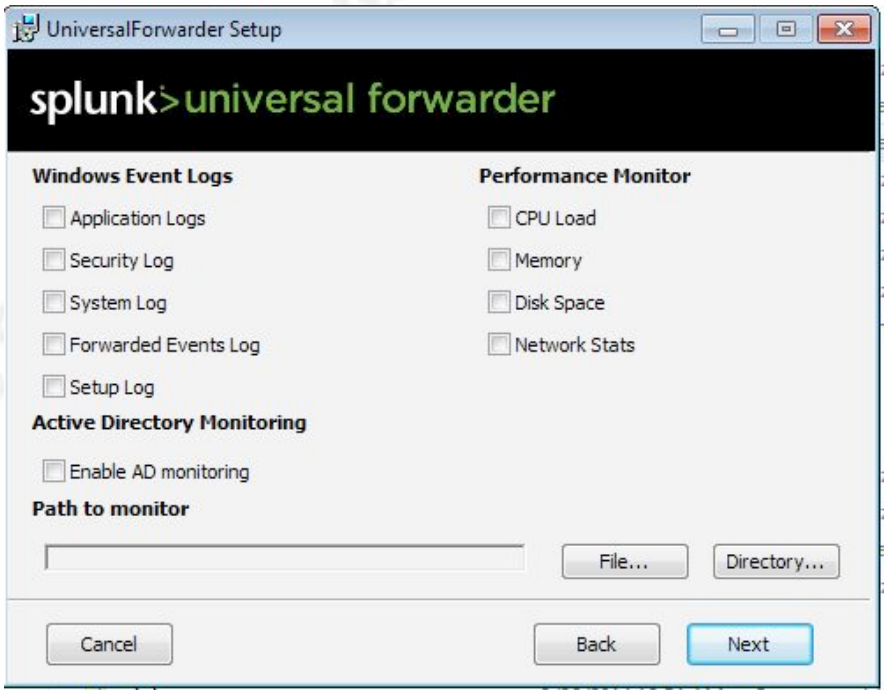

Figure 7: Splunk Universal Forwarder and the various configuration options (Alleyne, 2016).

4. Case Study

4.1. Executive Summary

Having recently experienced a compromise of its network infrastructure, in which there was significant financial and reputational loss, Fictional Inc. has decided to revisit their network architecture design and configuration. The primary driver for this architecture review was Fictional Inc.'s inability to detect a compromise while learning of it through a third party. More importantly, in an attempt to leverage the data within their infrastructure to learn about the compromise, Fictional Inc. found there was no way to identify when this activity started, from where, by whom and how, as their existing logs showed no trace of the reported activity.

4.2. Results of Architecture Review

In an effort to become forensically capable, Fictional Inc. first conducted a review of its network infrastructure, as shown in Figure 8. Because of the review, Fictional Inc. learned that it has a Demilitarized Zone (DMZ) consisting of a VSFTPD (Evans, n.d.) File Transfer Protocol (FTP) server, an Apache Web Server (apache.org, 2016) and a Postfix (postfix.org, n.d.) email server, all being run atop the Linux operating system (OS). It also has its general user network that consist of user workstations based on Microsoft Windows and Apple Mac OS X, file, database and directory services servers leveraging Microsoft Active Directory. Most importantly, Fictional Inc.'s network also consists of a perimeter firewall, a perimeter router and multiple layer two/three switches while also allowing access to its network via mobile devices. Finally, during this process, Fictional Inc. also identified that it has "cloud" hosted services.

Additionally, through the review, Fictional Inc. identified software and hardware, which were not authorized to be used within its network infrastructure. To address these findings, Fictional Inc. has decided to move forward with implementing the necessary steps needed to become forensically capable.

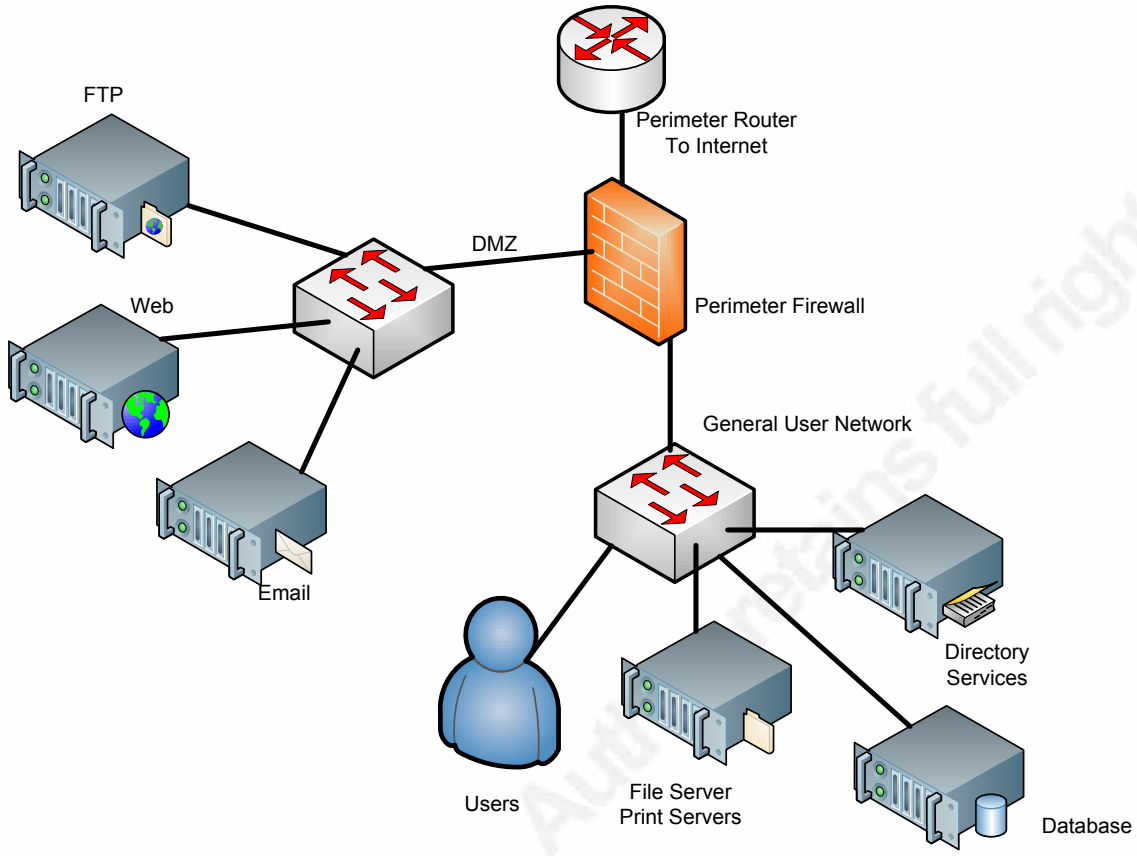

Figure 8: Map of Fictional Inc.'s network infrastructure before becoming forensically capable.

4.3. Moving to a Forensically Capable Network

To address the network review findings identified in 5.2, Fictional Inc. maintained much of the same infrastructure design. However, an additional management VLAN containing a SIEM/Log Collector (Splunk) was added, along with additional components such as a Hyper Text Transport Protocol (HTTP) Proxy on the General User Network, a NTP Time Server on the Directory Services server and a full packet capture device as shown in Figure 9.

In moving to becoming forensically capable, Fictional Inc. has decided to incorporate a mix of COTS and FOSS tools that address the business needs. It has also decided that every device that allows for authentication will have both successful and failed logins captured. Fictional Inc. also chose to monitor critical resources for both successful and failed attempts. More importantly, Fictional Inc. will also capture all

permitted and denied traffic at its firewall and perimeter router. Of greatest significance is that any configuration changes made to devices from a security perspective will be logged.

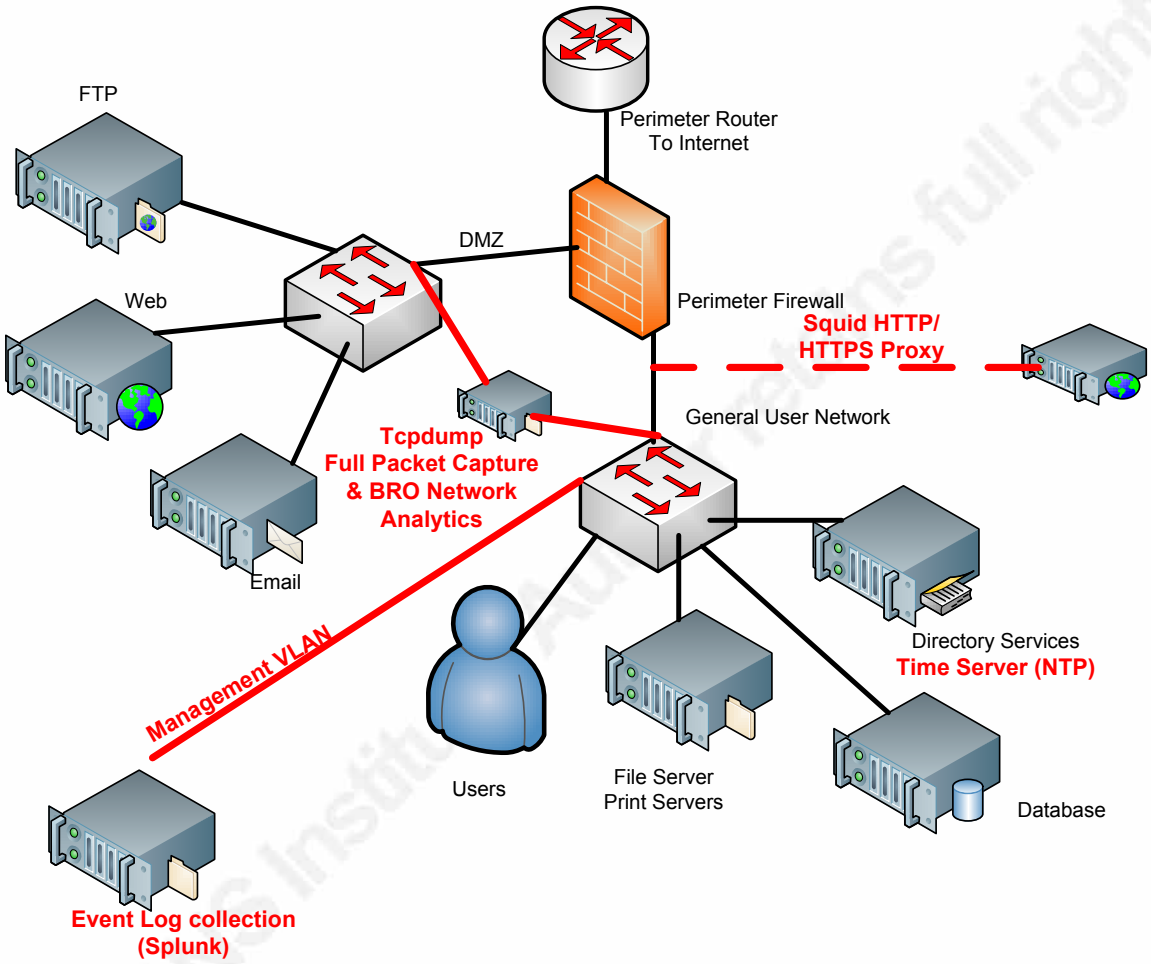

Figure 9: Changes made to become forensically capable.

4.3.1. The Importance of Time

In becoming forensically capable, ensuring proper time synchronization is the first step which should be completed. Fictional Inc., has chosen to create a Network Time Protocol (NTP) hierarchy in which its perimeter router obtains its time from the time servers identified in Figure 6. Its Microsoft Active Directory Primary Domain Controller Emulator (PDCE) server then obtains its time from the perimeter router with all other devices receiving their time from the Directory Services server. Because of the default

requirement for Windows devices leveraging Kerberos to have no more than five minutes time skew, leveraging the time from the PDCE is recommended.

4.3.2. Leveraging HTTP Proxy

By implementing a HTTP/HTTPS proxy, Fictional Inc., gains the capability of potentially monitoring all HTTP/HTTPS traffic that passes through its perimeter firewall device. This proxy becomes extremely important as it will be the primary source for which all HTTP/HTTPS messages can be seen.

4.3.3. Looking Deep Within – Full Packet Captures

To ensure maximum visibility and greatest possible network forensics capability, Fictional Inc. has implemented a full packet capture tool based on *tcpdump*. The traffic seen by *tcpdump* is through a Switch Port Analyzer (SPAN). Specifically the *tcpdump* filter focuses on traffic based on TCP and UDP communication. For business reasons, it was decided that there is no need to capture ICMP traffic, as this type of traffic is not permitted on Fictional Inc.'s network and thus is not needed for forensics purposes. Fictional Inc. has also chosen to accept the risk involved with monitoring only TCP and UDP based traffic. However, while these are being monitored, the decision has also been made not to monitor encrypted communication, as it is not being decrypted prior to capture and storage.

```
remnux@securitynik:~$ sudo tcpdump -nni any "net ( 10.0 ) and ( tcp or udp ) an
d not port( 22 or 443 or 465 or 563 or 636 or 989 or 992 or 993 or 994 or 995 o
r 500 )" -s 1514 -w 'FictionalInc-%F-%T.pcap' -z gzip -G 3600
tcpdump: listening on any, link-type LINUX_SLL (Linux cooked), capture size 151
4 bytes
```

Figure 10: tcpdump command showing capture filter.

Fictional Inc. also intends to leverage existing full packet captures within its network infrastructure as input to its BRO Network Analysis Framework (bro.org, n.d.).

4.3.4. SIEM/Event Log Collection

For event log collection, Fictional Inc. has chosen to leverage Splunk as it is the centralize repository for all event logs. With the data available in Splunk, Fictional Inc. has one location to retrieve all relevant event logs when the need arises for an investigation.

4.3.5. Capturing the appropriate data

While the potential exists to capture all data within Fictional Inc.'s infrastructure, Fictional Inc., has chosen to focus on only a small subset of the available data that is relevant to its security requirements. By focusing only on relevant data, Fictional Inc. puts itself in the strong position of having the data that is required when an intrusion occurs, thus being forensically capable. Additionally, the captured data will be retained for a period of 365 days, thus ensuring Fictional Inc., meets its compliance and regulatory requirements.

From the HTTP/HTTPS proxy perspective, Fictional Inc., has chosen to forward the data in the "access.log" file to its event log collection device. Additionally, for all devices that require authentication, to perform configuration changes, Fictional Inc., has chosen to capture both successful and failed attempts. More importantly all permitted and denied traffic from the perimeter firewall is logged. Similarly, for the perimeter router, all authentication events are logged along with configuration changes. Of greater importance, Fictional Inc. logs all access to their critical resources along with any attempts to elevate privileges.

4.3.6. Controlling Access to Logged Data

In an effort to ensure only authorized personnel can access the data stored within the full capture device and the centralize logging solution, Fictional Inc. has implemented a "jump-box" which allows all users to connect from a single device to access the centralized event logger and the full packet capture device. Additionally, firewall rules have been installed on both devices, ensuring that access is further controlled.

```
root@securitynik:~# iptables -I INPUT -i eth0 -s 10.0.0.20/32 -p tcp --dport 22
 -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
```

Figure 11: rule installed on both the centralized logging solution and full packet capture device.

```
root@securitynik:~# iptables -I INPUT -i eth0 -s 10.0.0.20/32 -p tcp --dport 8000
 -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
```

Figure 12: rule placed on the SIEM to allow access to Splunk console.

4.3.7. Agent vs Agentless

Fictional Inc.'s network has multiple OSes and applications. As a result, its forensically capable network consists of both agent and agentless devices. To collect data from the perimeter firewall, the router and Linux devices Fictional Inc. allows the various logs to be sent via Syslog over TCP and leverages SSL for encryption of these logs. However, for the devices running Microsoft Windows, it has chosen to leverage an agent such as the Splunk Universal Forwarder. By leveraging the Splunk Universal Forwarder agent, Fictional Inc. obtains the benefit of ensuring confidentiality and integrity of the forwarded data is achieved.

4.4. Ready for the Future Compromise

Because of these steps, Fictional Inc. has made their network forensically capable. It is much more likely that through regular proactive review of their logs and captured packet data, that they will be able to detect the next compromise before it is reported via third party. While having the data makes the network forensically capable, it is important that Fictional Inc. perform continual review of its architecture and captured data.

5. Conclusion

The realities of network intrusions and compromise resulting in exposed records continue to be a challenge for organizations. From the data identified in the introduction, organizations should consider themselves as possibly compromised. However, with all of this known information, organizations needs to make a better effort at ensuring their network infrastructures are forensically capable. The starting point should be first identifying the business needs. Once they are understood, efforts can be made to identify the people needed to support them. Ultimately, once the business needs and the people required to address those needs are identified, the next step is to establish the necessary processes and ultimately identify and obtain the technologies that will assist in making the network forensically capable.

While it may seem challenging, becoming forensically capable is not that difficult or expensive, once a needs assessment has been completed. Whether it becomes an expensive venture, is driven more by the business' appetite for FOSS or whether it

prefers COTS. Ultimately, the decision as to whether to become forensically capable should not be made upon finding out that the business has been compromised.

The case study in Section 4 provided a demonstration of a company that was compromised but was not forensically capable. Because of recognizing this, the company took the steps necessary to become forensically capable. These steps started with a review of the existing infrastructure, before making the necessary changes. The changes included leveraging Active Directory for NTP, implementing both COTS and FOSS technology solutions, implementing a management network as well as full packet capture solutions. With these changes, Fictional Inc. places itself in a much better position of detecting the next compromise of its network infrastructure.

6. References

Alleyne, N. (2016, March 27). *Learning about Mimikatz, SkeletonKey, Dumping NTDS.dit and Kerberos with Metasploit - Lab Setup* . Retrieved from securitynik.blogspot.ca: http://securitynik.blogspot.ca/2016/03/learning-about-mimikatz-skeletonkey.html

AMHIL, M. T. (2013, March 1). *"It's Simple!" – Time Configuration in Active Directory*. Retrieved from blogs.technet.microsoft.com: https://blogs.technet.microsoft.com/nepapfe/2013/03/01/its-simple-time-configuration-in-active-directory/

apache.org. (2016). *Apache - HTTP Server Project*. Retrieved from httpd.apache.org: https://httpd.apache.org/

bro.org. (n.d.). *The Bro Network Security Monitor*. Retrieved from bro.org: https://www.bro.org/index.html

CBC News. (2014, July 7). *Stephen Solis-Reyes, accused in CRA Heartbleed hack, has case put over*. Retrieved from cbc.ca: http://www.cbc.ca/news/politics/stephen-solis-reyes-accused-in-cra-heartbleed-hack-has-case-put-over-1.2709556

Chamith, B. (2012, March 21). *SSH Tunneling Explained*. Retrieved from chamibuddhika.wordpress.com: https://chamibuddhika.wordpress.com/2012/03/21/ssh-tunnelling-explained/

Cisco. (n.d.). *Messages Listed by Severity Level* . Retrieved from cisco.com: http://www.cisco.com/c/en/us/td/docs/security/asa/syslog-guide/syslogs/logsevp.html#30538

Evans, C. (n.d.). *vsftpd - Probably the most secure and fastest FTP server for UNIX-like systems.* Retrieved from security.appspot.com: https://security.appspot.com/vsftpd.html

Fee, J. (2013, August 6). *The Beginner's Guide to the Cloud*. Retrieved from mashable.com: http://mashable.com/2013/08/26/what-is-the-cloud/#UNqdc4YXHkq3

Author Name, email@address

Kohl, J. (1993, September). *The Kerberos Network Authentication Service (V5)*. Retrieved from tools.ietf.org: https://tools.ietf.org/html/rfc1510

Miao, F., Ma, Y., & Salowey, J. (2009, March). *Transport Layer Security (TLS) Transport Mapping for Syslog*. Retrieved from tools.ietf.org: https://tools.ietf.org/html/rfc5425

postfix.org. (n.d.). *The Postfix Home Page*. Retrieved from postfix.org: http://www.postfix.org/

RedHat. (n.d.). *22.9. Understanding the ntpd Configuration File*. Retrieved from access.redhat.com: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-Understanding_the_ntpd_Configuration_File.html

Risk Based Security. (2016). *Data Breach QuickView - Data Breach QuickView.* Risk Based Security.

Son, H., & Riley, M. (2014, September 5). *JPMorgan Had Exodus of Tech Talent Before Hacker Breach* . Retrieved from bloomberg.com: http://www.bloomberg.com/news/articles/2014-09-05/jpmorgan-had-exodus-of-tech-talent-before-hacker-breach

splunk. (n.d.). *Splunk Universal Forwarder*. Retrieved from splunk.com: https://www.splunk.com/en_us/download/universal-forwarder.html

timeanddate. (n.d.). *The Difference Between GMT and UTC*. Retrieved from timeanddate.com: http://www.timeanddate.com/time/gmt-utc-time.html

Verizon. (2015, April 15). *Verizon 2015 Data Breach Investigations Report Finds Cyberthreats Are Increasing in Sophistication*. Retrieved from news.verizonenterprise.com: http://news.verizonenterprise.com/2015/04/2015-verizon-dbir-report-security/

Walla, M. (2000, May). *Kerberos Explained* . Retrieved from msdn.microsoft.com: https://msdn.microsoft.com/en-us/library/bb742516.aspx

WebNMS. (2009). *SNMPv3 Overview*. Retrieved from webnms.com: https://www.webnms.com/snmputilities/help/quick_tour/snmp_and_mib/snmpmib_snmpv3.html

Author Name, email@address

worldometers.info. (n.d.). *Canada Population* . Retrieved from worldometers.info: http://www.worldometers.info/world-population/canada-population/

Zeising, N. (n.d.). *Configuring System Logging - Chapter 11. Configuration and Tuning*. Retrieved from freebsd.org: https://www.freebsd.org/doc/handbook/configtuning-syslog.html