

VISUALISATION OF PF FIREWALL LOGS USING OPEN SOURCE

Submitted in partial fulfilment
of the requirements of the degree of

MASTER OF SCIENCE

Rhodes University

Dirk Coetzee

Grahamstown, South Africa

January 2015

Abstract

If you cannot measure, you cannot manage. This is an age old saying, but still very true, especially within the current South African cybercrime scene and the ever-growing Internet footprint. Due to the significant increase in cybercrime across the globe, information security specialists are starting to see the intrinsic value of logs that can ‘tell a story’. Logs do not only tell a story, but also provide a tool to measure a normally dark force within an organisation.

The collection of current logs from installed systems, operating systems and devices is imperative in the event of a hacking attempt, data leak or even data theft, whether the attempt is successful or unsuccessful. No logs mean no evidence, and in many cases not even the opportunity to find the mistake or fault in the organisation’s defence systems.

Historically, it remains difficult to choose what logs are required by your organization. A number of questions should be considered: should a centralised or decentralised approach for collecting these logs be followed or a combination of both? How many events will be collected, how much additional bandwidth will be required and will the log collection be near real time? How long must the logs be saved and what if any hashing and encryption (integrity of data) should be used? Lastly, what system must be used to correlate, analyse, and make alerts and reports available?

This thesis will address these myriad questions, examining the current lack of log analysis, practical implementations in modern organisation, and also how a need for the latter can be fulfilled by means of a basic approach. South African organizations must use technology that is at hand in order to know what electronic data are sent in and out of their organizations network.

Concentrating only on FreeBSD PF firewall logs, it is demonstrated within this thesis the excellent results are possible when logs are collected to obtain a visual display of what data is traversing the corporate network and which parts of this data are posing a threat to the corporate network. This threat is easily determined via a visual interpretation of statistical outliers. This thesis aims to show that in the field of corporate data protection, if you can measure, you can manage.

Acknowledgements

Several individuals contributed to my studies and I would like to thank them for their contribution.

I would firstly like to thank my late mother, Mara Coetzee, for all her support and caring and for making life possible.

To my father, Frans and to Lucia Coetzee, thank you for all your support and caring and for making sure that I am on time for departures to and arrivals from Grahamstown.

I would also like to thank my parents-in-law, Karin and Johan van Niekerk, for looking after my son over weekends, after hours, holidays and leave days while I was studying.

As always, to my wife, Bianca Coetzee and son, Louis Coetzee, thank you for all your support.

Thank you to my supervisor, Barry Irwin, for all the time he spent making sure we received all the required information. Thank you for keeping me to strict deadlines. Without your guidance and inspiration none of this would be possible.

Finally, I'd like to thank Rian Olivier for taking the time out of her busy schedule to proof read my assignments, Zakhele Khuzwayo, Sean Greven and Sean White for all the advice and support, and Spalding Lewis for being my copy and language editor.

Contents

Table of Contents	i
List of Figures	vi
List of Tables	viii
Glossary	ix
1 Introduction	1
1.1 Problem Statement	3
1.2 Research Methodology	3
1.3 Research Scope and Limitations	3
1.4 Research Goals	4
1.5 Document Conventions	4
1.6 Document Structure	5
2 Literature Review	6
2.1 Crime	6
2.2 Cyber Crime	7

2.3	What is a log data	8
2.3.1	Collection of logs	10
2.3.2	Log Management	10
2.4	Hacktivsm	10
2.4.1	Viruses	11
2.4.2	Trojan Horse	12
2.4.3	Rootkit	13
2.4.4	Firewalling	13
2.4.5	Intrusion Prevention Systems	13
2.5	Network Telescopes	14
2.6	SIEM	14
2.6.1	What does a SIEM do?	15
2.6.2	Why is a SIEM necessary?	15
2.6.3	Statistics with regards to SIEM	17
2.7	Criteria for a good SIEM system	18
2.8	FreeBSD	19
2.9	Packet filter	20
2.10	Digital forensics	21
2.10.1	Define forensic readiness	22
2.10.2	Necessity of an organisation to be digital forensically ready	22
2.10.3	Digital forensically ready preparation and components	23
2.10.4	Digital Forensics and data logs	24
2.11	Summary	24

3	Data Collection	25
3.1	Importing of logs into the database	29
3.2	Summarization of data entries	29
3.3	Compression Evaluation	30
3.4	Data Collected	32
3.5	Database Layout	32
3.5.1	Table traffic_copy	32
3.5.2	Table srcip_dstip_dstport_hour_structure	33
3.5.3	Table ip_lookup	33
3.5.4	Table country_codes	33
3.6	Data Analysis	34
3.7	Summary	35
4	Visualisation System	36
4.1	Hardware and Software specifications	37
4.2	Network and DMZ Architecture	38
4.2.1	Client connectivity to the Internet	39
4.2.2	Internet Connectivity to Web servers in DMZ	40
4.2.3	Host based firewalls	40
4.2.4	DNS Traffic	40
4.2.5	NTP Traffic	41
4.2.6	SMTP Traffic	41
4.2.7	Remote management / Access	41

4.2.8	Firewall Rules	41
4.2.9	PFLogs	42
4.3	Visualisation System Layout	42
4.3.1	Source IP Graph	43
4.3.2	Destination IP Graph	44
4.3.3	Google Maps	49
4.3.4	IP Search Graph	51
4.3.5	Ports Search Graph	52
4.3.6	Dual IP Search Graph	54
4.3.7	Search Database	55
4.3.8	Locate IP	57
4.4	Summary	59
5	Case Study - PC infected with Malware	61
5.1	Top 20 Flagged IPs	63
5.2	IP 10.133.2.71	64
5.3	Destination Port 25/tcp	66
5.4	Destination Port 447/tcp	69
5.5	Destination Port 44416/tcp	70
5.6	RDP connection to IP 10.133.2.71	71
5.7	Summary	74

6	Conclusion	75
6.1	Research evaluation	76
6.2	Expectations during the start of the research	77
6.3	Lessons learned	78
6.4	Future Work	78
6.5	Conclusion	79
	References	80
	Appendix	86

List of Figures

2.1	Cyber Crime Example	8
2.2	SIEM Architecture	16
2.3	Distributed SIEM Architecture	17
3.1	Data Collection Process	26
3.2	Database layout	34
4.1	Network Architecture	39
4.2	Visualisation System	43
4.3	Flagged IPs	47
4.4	Detailed Flagged IPs	49
4.5	Google map	50
4.6	IP Search Graph	52
4.7	Destination Ports Search Graph	54
4.8	Search Database	56
4.9	Locate IP Query Screen	58
4.10	Locate IP Results Screen	59

5.1	Top 20 Flagged IPs	64
5.2	Detailed table of Case Study	65
5.3	Summary graph for Case Study	65
5.4	Destination port 25/tcp search graph - Source IPs	66
5.5	Destination port 25/tcp search graph - Destination IPs	67
5.6	Totalhash Output Example	68
5.7	Destination port 447/tcp search graph - Source IPs	69
5.8	Destination port 447/tcp search graph - Destination IPs	70
5.9	Destination port 44416/tcp search graph - Source IPs	70
5.10	Destination port 44416 search graph - Destination IPs	71
5.11	Combined netstat output for Case Study	72
5.12	Virus report for Case Study	73

List of Tables

3.1	Text log compression ratio	31
3.2	Transfer speed and time	31
3.3	Firewall log data Collected	32
4.1	Hardware and Software specifications	38
5.1	TCP three way handshake	62
5.2	Destination Port 25/tcp Top 10 Destinations	67
5.3	Destination Port 447/tcp Top 10 Destinations	70
5.4	Destination Port 44416/tcp Top 10 Destinations	71
5.5	netstat output	72
5.6	Observed Ports	74

Glossary

ACL	Access Control list
ADSL	Asymmetric Digital Subscriber Line
ARPANET	Advanced Research Projects Agency Network
BZ2	bzip2
CARP	Common Address Redundancy Protocol
CCTV	Closed-circuit Television
CD	Compact Disc
DDoS	Distributed Denial-of-service
DLP	Data loss prevention
DMZ	Demilitarized Zone
DNS	Domain Name System
DoS	Denial-of-service
DVD	Digital Video Disc
FAT	File Allocation Table
FTP	File Transfer Protocol
GB	Gigabyte
HTTP	Hypertext Transfer Protocol

HTTPS	Hypertext Transfer Protocol Secure
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
ICT	Information and communications technology
IP	Internet Protocol
IPS	Intrusion Prevention System
IPSec	Internet Protocol Security
L2TP	Layer 2 Tunneling Protocol
MB	Megabyte
MS-DOS	Microsoft Disk Operating System
NTP	Network Time Protocol
pcap	Packet Capture
PF	Packet Filter
SCP	Secure Copy Protocol
SFTP	Simple File Transfer Protocol
SIEM	Security information and event management
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SQL	Structured Query Language
SSH	Secure Shell
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
USB	Universal Serial Bus
VPN	Virtual Private Network

1

Introduction

ONE of the earliest Information Security risks that necessitated the development of protection was the computer virus, which was at the time primarily transferred via floppy disk [Thadani, 2013]. As the years passed, however, technology became more sophisticated. In today's world most computers and cell phones are connected to the Internet [Kende, 2012]. As technology is suppose to make life easier, people became smarter. The days when war between countries were fought with explosives, guns and knives are "replaced" by cyber war. There is no reason to expose any humans to dying "on the front" and more due to the fact that technology can be used to break into systems, fly remote controlled attack drones and attack or spy on the enemy or target. Cyber space is here and it is real, but so is Cyber war. Looking at the comment "If there is fighting in the streets, we can expect fighting on-line - attributed to an Estonian official during the 'Cyberwar' in May 2007" it is also mentioned that a "digital civil war" may break out [Paget, 2012].

The conventional burglar, extortionist, fraudsters was replaced by the cyber criminal that sits in front of his computer, looks for vulnerabilities on your computer systems and exploit them one by one. This can be done for various reasons from personal satisfaction to the point of Industrial espionage having significant financial reward. Organisations Internet facing servers, if found, may be exposed for fun, theft or hacktivism.

In this day and age an organisation must plan and incorporate Information security on a strategic level in the organisation as Information Security is key. The strategy should be turned into practise by having on the perimeter of the organisation items or solutions such as, Demilitarised zones (DMZ's), Intrusion prevention systems (IPS) devices, Data loss prevention (DLP), content filtering and Log analysis.

In the early days of the computer era systems were developed with minimal logging functionality, primarily used for debugging purposes. As time past, storage became cheaper and increased in size dramatically [Pingdom, 2010]. Logging was implemented within programs and systems as a standard [Kreps, 2013]. Logs are not only used as a mechanism for debugging and problem finding but as information and evidence in a court of law. If an electronic logs strategy is implemented, aligned to best practices and law, logs can be used as evidence, can prove, without a doubt, that a connection was made from A to B using protocol C and depending on the amount of data logged even what transactions were done or data exchanged.

Logging are supported by many systems and programmes in this day and age ranging from server, network equipment, database, access control, firewalls, Data loss prevention, Intrusion Prevention systems, up to the Asymmetric digital subscriber line (ADSL) router running at home. According to an article from govinfosecurity [Chabrow, 2012] “22 percent [of organizations] collect log data and process them exclusively with their SIEM systems” as per findings by SANS. The amount of data that flows through our networks is enormous therefore organisations should decide what should be logged, why the data is logged, how long the data must be kept, how it must be collected and stored in order to be used in a court of law as well as comply with legislation. In the case where log collection are implemented in a way that covers the legislation aspect, organizations business requirements, Information and communications technology (ICT) requirements and best practices, logs can be used not only as proof in a court of law but up to a level that the logs can be used to identify problems, bottlenecks as well as tell a story how a security event occurred. Firewall logs can be used not only for auditing purposes but for having the capability to see what is happening on your network, how good or bad your firewall is configured.

Due to the volume (number of log entries and size in MB) of data that travels over networks it is difficult to have all systems logging all events, correlate, analyse the logs and present it in real time as meaningful data in this day and age. Data can be Gigabytes in size within minutes.

The solution hints to be one where a magnifying glass is taken and a specific issue is investigated in a lot of detail, rather than scanning across a wide field and spotting the outliers.

1.1 Problem Statement

A major problem experienced by many organisations within South Africa are the fact that there are no visibility on what is running on the organization network nor pro-actively monitoring of logs in order to identify problems, cyber attacks, virus breakouts or unauthorised access attempts. This is only on the negative side. On the other side of the coin are systems not monitored in order to identify for example events that happen outside a baseline. An example where a flag must be raised is when a employee working within the finance department is only authorised to work during working hours, Monday to Friday 9am to 5pm, and the employee authenticate to the financial system 2am over a weekend. This event must be logged, a red flag raised and be investigated immediately.

The major problem that must be addressed is the fact that the myriad of electronic logs, from access logs (physical and logical), system logs, to firewall and anti virus logs must be proactively monitored 24 x 7. A baseline of normal activity must be defined where after any activity below or above the baseline will raise an alert in an easy manner with minimal additional workload.

Without proper tools data on this level analysis would not be possible, and it is part of the aim of this thesis to indicate ways to turn large amounts of data into usable information.

1.2 Research Methodology

Data gathering and analysis were primary used as research methodology during the research project. Data from an corporate organization were use combined with custom built scripts and programs to collect, analyse and visualise the relevant data.

1.3 Research Scope and Limitations

For the purpose of this thesis PF logs from the DMZ firewalls will be used to present graphs and tables for denied logs only in order to identify possible security risks such as the use of programs or scripts to connect into or out of the organizations network to steal, leak information, cause damage to the organizations assets. This thesis must be viewed in the South African context as data was only collected from a company within South Africa.

1.4 Research Goals

The aim of this research was to build a proof of concept system that could store, analyse, and visually present graphs as well as statistics of FreeBSD¹ PF² firewall logs by using an easy to use open source solution that could be used within environments where FreeBSD and PF are installed. The goals of the proof of concept system will be to:

1. store and forward the PF logs using a technology such as syslog.
2. store syslog events in a MySQL database.
3. correlate log entries (events per hour) .
4. present summarised blocked logs in the format of graphs, tables and maps.
5. identify blocked connections into the or out of the organization network.
6. identify alerts by analysing the PF logs with the proof of concept Visualisation System.

1.5 Document Conventions

Conventions that are used within the remainder of the document are as follows:

1. Where mention is made to an application or particular site, a URL will be provided as a footnote.
2. The date and time format used within this document is Year-Month-Day hour:minute:second
3. The format used for ports are port number/protocol as per Internet Assigned Numbers Authority (IANA³) for example Hypertext Transfer Protocol Secure (HTTPS) will be listed as 443/tcp.
4. Where sensitive IP addresses were found, the first and second octet will be masked with xxx.yyy

¹<http://www.freebsd.org/>

²<http://www.openbsd.org/faq/pf/>

³<http://http://www.iana.org/>

1.6 Document Structure

Remainder of this document is structured as follows:

Chapter 2 includes the Literature Review and provides an overview of key concepts that will be covered during this thesis. This chapter will also present work published by researchers working in the same field as the author of this thesis.

Chapter 3 describes how the source data was collected and imported into a database and the structure of the logs and database.

Chapter 4 deals with how the proof of concept Graphing system were designed and how the system works. This chapter will also cover the network architecture that were implemented during the data gathering period. The configuration of the firewall and network will also be discussed.

Chapter 5 constitutes the majority of this document. One case study is presented and explained in order to show the reader how firewall logs can be used to identify security alerts and how logs can be used to assist a firewall administrator to build a firewall rules template from firewall logs.

Chapter 6 the findings obtained during the research are reflected upon.

2

Literature Review

THIS chapter will cover concepts that are used within this thesis. Several definitions will be given with an explanation of how they integrate and complement each other. Overviews will be given of work performed by other researchers working in the same field of study. Digital forensics will also be covered in detail due to the fact that logs often function as legal evidence.

The days of thieves stealing an organization's data by breaking into a given physical location are mostly in the past. Contemporary cyber crime is most often committed in front of the computer rather than on the street, with the perpetrator hunting vulnerabilities in an organization's Internet-facing servers. If found, the modern cyber criminal exposes these weaknesses in order to steal the data.

2.1 Crime

A crime is defined as when rules that are implemented and approved by a governing authority or body are broken or not adhered to [Oxford-Dictionaries, 2014].

In the world today there are certain rules and regulations implemented depending on where you stay and circumstances for example, taking from a person something that does not belong to you without the owners freely-given consent is considered as theft and is a crime [SAPS, 2013].

2.2 Cyber Crime

Cyber, according to the Oxford English Dictionary Online, describes that which is “relating to or characteristic of the culture of computers, information technology, and virtual reality: the ‘cyber age’” [Oxford-Dictionaries, 2013].

From this definition, it is easy to see how the concept of ‘cyber crime’ was gradually developed. Cyber crime can be defined, therefore, as a criminal act done using a computer or other means of electronic communication, such as public or private network [Yar, 2005].

Cyber crime can be defined more specifically as 1) the unlawful downloading of copyrighted material such as software, music, videos; 2) the act of using electronic technology to gain unauthorised access, initiate information leakage, or steal money from banks or institutions; 3) deception, writing or distributing viruses, worms, and Distributed Denial-of-service (DDOS). Even the act of publishing Intellectual property or confidential information electronically is considered as cyber crime in some cases [Wall, 2007, Maat, 2009].

The legal parameters of the concept of cyber crime should also be noted, namely that when user credentials are used without the authorization of the owner in order to gain access to an electronic system, this act is considered as a cyber crime under the category of identity theft [Granova and Eloff, 2004, Steyn et al., 2007].

In Figure 2.1 examples of cyber crimes are displayed in the outer white block. Moving to the grey block, examples of mediums that are in use by cyber criminals today are displayed, however the mediums that cyber criminals use grow by the day. Within the middle gold circle examples of the impact that cyber crimes have on the world are displayed. Cyber crime is a world wide problem and increasing every year [Sahil, 2011]

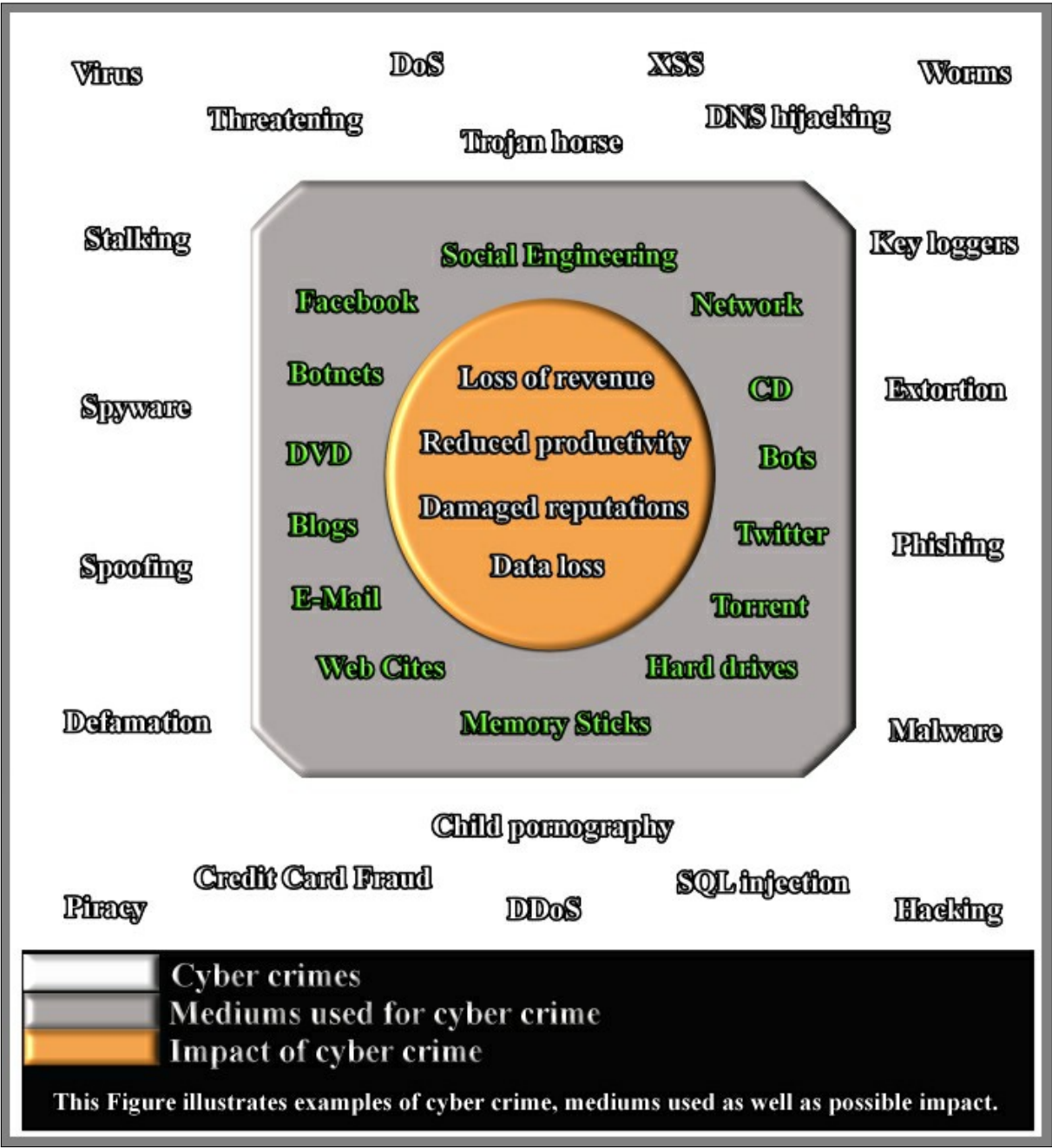


Figure 2.1: Cyber Crime Example

2.3 What is a log data

The core of log data are logs, also known as log messages, which can be defined as the response generated when an event is triggered by software, devices or computer systems. Below are several examples of how log data are defined and used in real life:

1. As soon as a Universal Serial Bus (USB) storage is connected to a USB port on a FreeBSD server, a log entry will be added to the `/var/log/messages` file [Lucas, 2007, Lavigne, 2007].
2. When an user authenticates to a FreeBSD server, a log entry will be added to the `/var/log/auth` file (default path) [Lehey, 2003].
3. When a PF firewall rule allows connections to a server, logging is enabled `log` or `log all`, the PF rule are triggered and the initial packet or all packets are logged to `/var/log/pflog` (default path), depending on the log settings chosen in the PF rule [Hansteen, 2014]
4. In cases where developers added the functionality to display activities performed by an application, known as `debug` or `verbose`, such as Secure Shell (`ssh`) `-v` (`-v` enables `verbose`) that prints messages during each action assisting in fault finding when an `ssh` session can not be established [Barrett, 2005].

Severity and log level

When enabling and configuring data logging, there are several levels of severity and detail that are saved in the log message, a process that can be configured. In this case there are several severity levels whereof the severity level 0 is a high-severity. With this severity level critical events are logged. With low severity level all events are logged, even the minute events. These two severity levels are used mostly during fault finding or identifying critical events. Verbose levels depend on the software as well as the vendor [Sadowski, 2010]. The higher the severity and detail levels, the more events are logged, the more storage is required and the more details are recorded for fault-finding or forensic evidence.

Combinations of severity and detail levels can be configured:

1. High severity and low detail: Low storage required and only critical events with minimal information are available for analysis.
2. Low severity and low detail: Medium storage required and all events are available with minimal information available for analysis.
3. High severity level and high verbose: Medium storage required and only critical events, with detailed information, are available for analysis.
4. Low severity and high verbose: with detailed events: High storage required and all events, with detailed information, are available for analysis.

2.3.1 Collection of logs

With log data defined and the identification of which log data is valuable we must then identify the storage required to save the logs. Should logs be stored locally, for example, or is a remote location more suitable? Logs stored locally require huge amounts of storage, with the risk of someone compromising them, either by altering or deleting logs. It is also difficult to manage and monitor logs on several servers that are storing their logs locally. In the case of remote logging the question becomes whether logs should be stored remotely in real time, requiring additional dedicated bandwidth, or whether the logs should be sent at scheduled times to the remote server. In cases where logs are saved remotely on a central server, huge amounts of storage must be attached to the server and high powered servers are required for any analysis [Sadowski, 2010].

2.3.2 Log Management

A critical part of identifying an unauthorised access, whether data was stolen and when, is log management key. Implementing security solutions, software, policies and procedures is fruitless if the logs do not exist. These logs may be access logs from routers, logs indicating who logged on and from where, and which user accessed what data and when. This is part of accountability. In several reports it is revealed that when data is stolen from a organization they are not ultimately able to identify the attack, whether it was internal or external [Armerding, 2012], for precisely this reason.

Physical and logical security controls must be linked to a trusted time source and all data received must be stored in a secure physical and logical way. Integrity, availability and accountability are key when the need arises to use the data in a court case, in fault finding and in investigations. [Pan and Batten, 2005]

2.4 Hacktivism

Hacktivism describes the use of computer equipment or digital tools to promote political ends, such as free speech or human rights, or to expose corruption and promote global freedom. Hence a 'hacktivist', or a hacker that hacks for political reasons or social causes is seen as a criminal. Whereas a hacker breaks into systems to steal information or to cause harm, a hacktivist does the same but for political reasons [Whitney, 2004].

Anonymous is a group of hackers that believe that all information should be free. The group claim to have no structure or leader and not are not part of a group but "everything and nothing" [Olson, 2012]. Anonymous are seen in some instances as heroes but according to the law they are labeled as a group of hacktivist hackers.

An example of a reported incident where hacktivists made additional information including video, data and email data available to the public in regards to the Steubenville, Ohio incident where a girl was allegedly raped. "In this case, as in all others, Anonymous explains its actions as contributing to the common good, by exposing corruption and offering an avenue of justice for the victims" [Marczak, 2013] In this case Anonymous, a hacktivist group, ignored the privacy barrier and made information about this issue available to the public in order to generate controversy and, potentially, social change.

Another hacktivist incident, again involving Anonymous, saw the group claiming to have stolen credit card data from the United States Security Department, making a U.S Homeland Security's employee contact information, as well as credit card details, available on the Internet. As Cody Stultenfuss stated in his article "Hackers target US security think tank", "They took money I did not have, I think why me? I am not rich." to the *Associated Press* [NDTV, 2011].

While some may argue that hacktivism promotes social good, according to law hacking where unauthorised is illegal therefore hacktivism is an illegal activity.

In the article "2011 the year of the hacktivist", Verizon data breach report reveals [Warwick, 2012] it is stated that during the year 2011 hacktivists stole 58% of data, and hacktivist attacks are on the rise. This article further states that: intrusions into smaller organizations are often not detected due to a lack of skilled resources and technology. On the other hand, larger organizations can also miss the signs of intrusions, as the skilled resources and technology required are often only appointed and installed to comply with minimum requirements. It is for this reason that logs must be mined for suspicious activity due to the fact that analysed logs usually identify breaches [Warwick, 2012].

2.4.1 Viruses

Computer viruses are small computer programs or scripts that were written to negatively affect or damage a computer. Some of the first computer viruses was made known in the 1970s, including Creeper, an early virus written by Bob Thomas, which was a self a replicating program first detected on the Advanced Research Projects Agency Network (ARPANET) [Judith, 2012].

The first virus for Microsoft Disk Operating System (MS-DOS) was BRAIN, released in 1986 [TechExpert, 2007]. BRAIN replaced the MS-DOS boot sector that was formatted with File Allocation Table (FAT) with a copy of the virus, and the original boot sector was moved to a sector marked as bad. The damage caused by viruses is widespread and covers a broad spectrum; with the advancements in technology viruses are used by criminals to steal data, destroy data encrypted data and held for ransomed [Wilding, 2006, Gifford, 2009].

Anti-virus software, including anti-malware and anti-spy-ware, is a must for organizations [Mills, 2012]. The software and definition files must be updated daily as there are constant “0 day” exploits that are made known. Storage mediums such as CD’s, DVD’s, USB [Brown, 2011], storage on cell phones and tablets, memory sticks are used on a daily basis and can all transfer viruses. Without the employee knowing, malicious software could be transferred from medium to medium or unknowingly downloaded from the Internet. If there is no updated Anti-virus package running on the organization’s computer to which the devices are connected, data integrity could be compromised, even sent to the outside world (data leakage). Most organizations’ computers are connected to the Internet, and by default HTTP (80/tcp), HTTPS (443/tcp) and SMTP (25/tcp) are all allowed to access the Internet. They are therefore used by hackers (external threat) and employees (internal threat) to leak organization data [Asaf et al., 2012].

2.4.2 Trojan Horse

The Trojan horse is so named as a reference to the mythical conclusion of the seige of Troy given in Ancient Greek poetic accounts. When the Greeks found they could not penetrate the city of Troy, a wooden horse was given as a peace offering [Burgess, 2001]. The gift was accepted and moved to within the walls of the city. However, during the night Greek soldiers hiding inside the hollow body of the horse opened its false door and unlocked the gates of the city while it slept. To return to the subject of this thesis, unless you are a developer who understands the programming language used to develop the program, have the source code of the program and have the time to verify every piece of a program’s code, no one will know what other programs are hidden within a program. Also referred to as ‘back doors’ for obvious reasons, trojan horses are known to be used for collecting data as per a designed program code; in many cases they send data to a server hosting a service such as File Transfer Protocol (FTP).

2.4.3 Rootkit

Rootkits are software tools developed to gain administrative access to a computer. Rootkits are usually installed after a computer is compromised when an attacker uses an exploit or attaches the rootkit to a trojan horse. Depending on how the rootkit is developed it may serve as a back-door into a computer, modifying logs; it may even act like a worm and infect other computers [Rouse, 2008]. The collective name 'virus' should in this day and age be seen as an advanced program that has the ability to intelligently spread, collect information, destroy information sniff packets, be controlled as part of a bot network or even gather information about a network or system over long periods of time.

2.4.4 Firewalling

Some typical Network Security devices are firewalls and Intrusion Prevention Systems (IPS). A firewall (access list) is the first line of defence. It will only allow data to flow as per specified policy (source Internet Protocol (IP), source port, destination Internet Protocol (IP), destination port and protocol). For example, if data flow in a policy is specified to allow web traffic (port 80) from the Internet to the organization web-server using only HTTP, only HTTP traffic will be allowed to the specified server on the specified port. In the case of a hacker sending a Structured Query Language (SQL) injection over HTTP, the firewall will allow the traffic to the web-server as long as the hacker sends the traffic to the allowed destination server IP and port. [Sammut, 2012].

2.4.5 Intrusion Prevention Systems

An IPS will inspect the payload of a packet and either perform a packet analysis, attack signature matching or both. If in this case the IPS is installed after the firewall and it has the capability to 'drop' identified attacks the IPS will 'receive' the packets from the firewall, inspect them and in this case drop the SQL injection before it reaches the web-server. According to IBM their Proventia Network IPS can monitor, detect and block attacks, to name a few of its functions [IBM, 2011]: "Application attacks, Attack obfuscation, Cross-site scripting attacks, Data leakage, Database attacks, DoS and DDoS attacks, Drive-by downloads, Insider threats, Instant messaging, Malicious document types, Malicious media files, Malware, Operating system attacks, Peer-to-peer, Protocol tunneling, SQL injection attacks, Web browser attacks and Web server attacks".

2.5 Network Telescopes

Network telescopes are usually installed within a public subnet that does not host any services or any legitimate traffic. These telescopes will capture and analyse all traffic received within the subnet. Analysing and categorising these packets will help identifying Internet worm spread, DOS attacks, scanning of IPs and services within the subnet [Moore et al., 2004]. By implementing these network sensors in several locations across cities and even countries, virus spreads and attacks can be viewed on a global scale while they likewise pinpoint the precise origins of attacks.

Several papers discussing network telescopes were found concentrating on the analysis of network telescope data. The research work conducted by Irwin [2011] and Cowie [2012] were used as reference for logs analysis in this project.

In the article “Deep Packet Inspection using Parallel Bloom Filters”, techniques are made available to detect pre-defined data strings within packets travelling at wire speed. These techniques were used within my research project to identify usable data at high speeds [Dharmapurikar et al., 2003].

Deep packet inspection using regular expressions, as published in “Fast and memory-efficient regular expression matching for deep packet inspection” [Yu et al., 2006], states that using their DFA-based implementation can be faster than the widely used NFA implementation.

2.6 SIEM

A SIEM solution is mainly used within the Information Security environment to collect logs from several different devices and applications across an IP network. It can then correlate events, identify risks, as well as take action on triggered events. There are many SIEM (Security information and event management) [NetworkComputing, 2012] systems in the industry that does analysis of logs, correlation with a central console. Examples of such opensource and commercial systems are Alienvault⁴, OSSIM⁵, Securityonion⁶, IBM Security QRadar⁷, Security Information and Event Management (SIEM)⁸.

⁴<http://www.alienvault.com/>

⁵<http://www.ossim.org/>

⁶<http://securityonion.net/>

⁷<http://www-03.ibm.com/software/products/en/qradar-log-manager/>

⁸<http://www.mcafee.com/us/products/siem/index.aspx/>

2.6.1 What does a SIEM do?

A SIEM solution does centralised log analysis. In the case where an attack is detected by the SIEM solution the latter will take action by informing the relevant engineers or taking counter measures, all depending on functions and configurations. SIEMs are also used for identity management, identifying problems on the network and lastly for policy monitoring [Karlzén, 2009]. SIEM's are a crucial part of any organization.

SIEM's are designed to collect data logs from several devices such as firewalls, switches, servers, Intrusion prevention systems using different protocols such as Simple Network Management Protocol (SNMP), Internet Protocol Security (IPsec) , Simple File Transfer Protocol (SFTP), Secure Copy Protocol (SCP) and SYSLOG. In several cases a software agent must be installed to collect the data and can be costly, taking time and effort into consideration. This kind of log collection can be done in a either push or pull state. The pull action is where logs are retrieved by the SIEM as per scheduled time intervals and push action is when the device sends the logs to the SIEM as per scheduled time intervals.

2.6.2 Why is a SIEM necessary?

Security events - an abnormal network activity that was identified or a spike in data flow above the baseline, for example - can be an early warning of a possible hacking attempt (Security breach). It has been shown in several articles that in some cases organizations only found out days after the attack that their security were breached and data was stolen [Mills, 2012, Lewis, 2013, Woollaston, 2015]. If the organization in the article implemented a logging solution with a baseline and event management they would have realised that their data had potentially been exposed to unauthorised access through their network.

In Figure 2.2 a high-level architecture is made available to the reader where examples are given of what systems can connect with different mechanisms to the SIEM solution, as explained in Section 2.6.

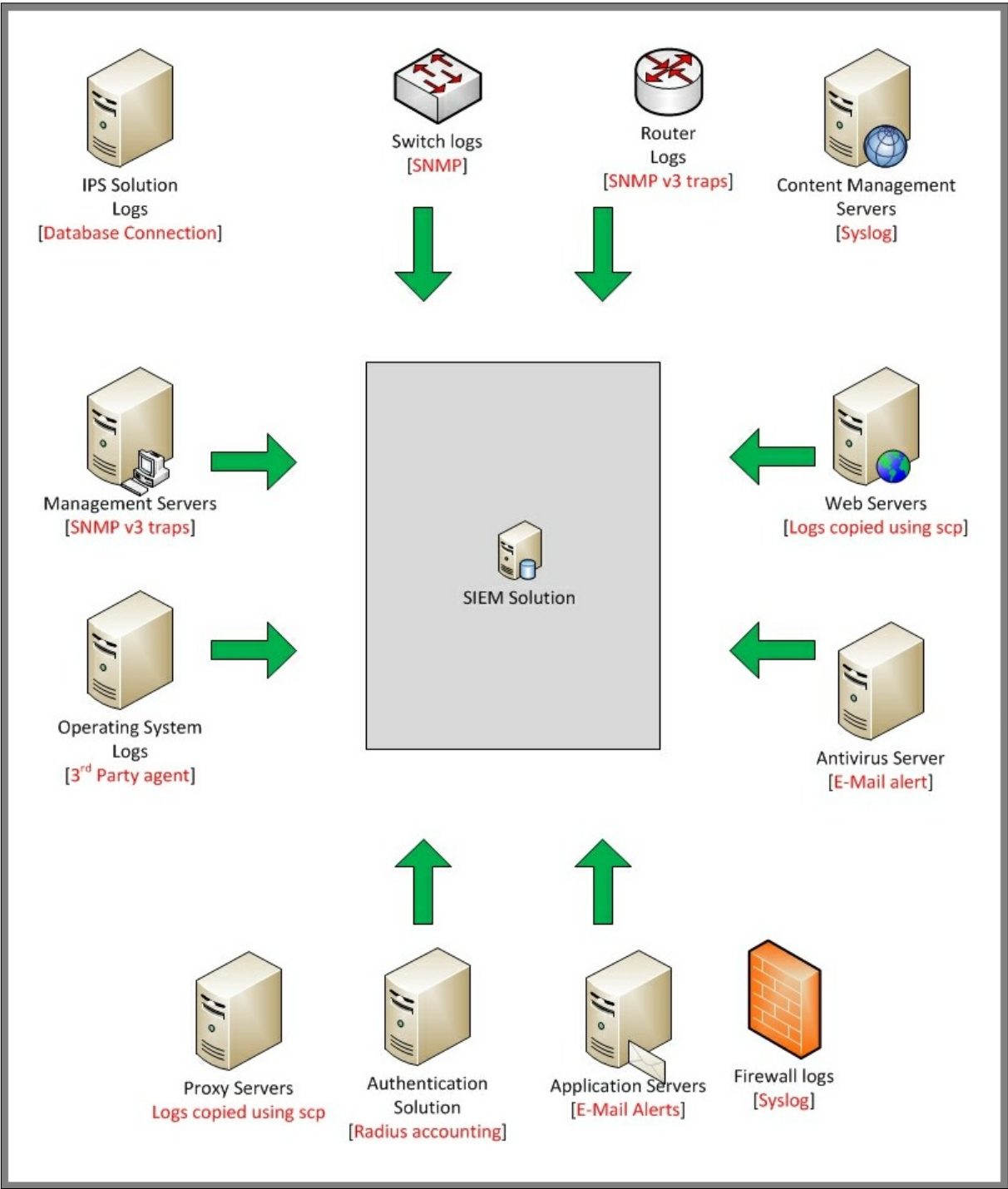


Figure 2.2: SIEM Architecture

In Figure 2.3 a example of a SIEM installation distributed over three cities is presented. Events are gathered in Bloemfontain, as per A in Figure 2.3, and logs and events are passed on to the Local SIEM server (C) in Bloemfontein where logs and events are then correlated. From here on the correlated data is sent on scheduled basis, as per D, to the Master SIEM server in

Pretoria. The same happens for the events and logs that are sent from device,s as per I in Figure 2.3, to the local server (G) where the data is saved and correlated. The next step is to send the correlated data, on a scheduled time, to the Master server in Pretoria where the data is stored. Alerts can be raised in each local server, namely the Bloemfontein Server (C) and the Cape Town Server (G). With all data in a central repository reports can be requested, data analysed and alerts raised.

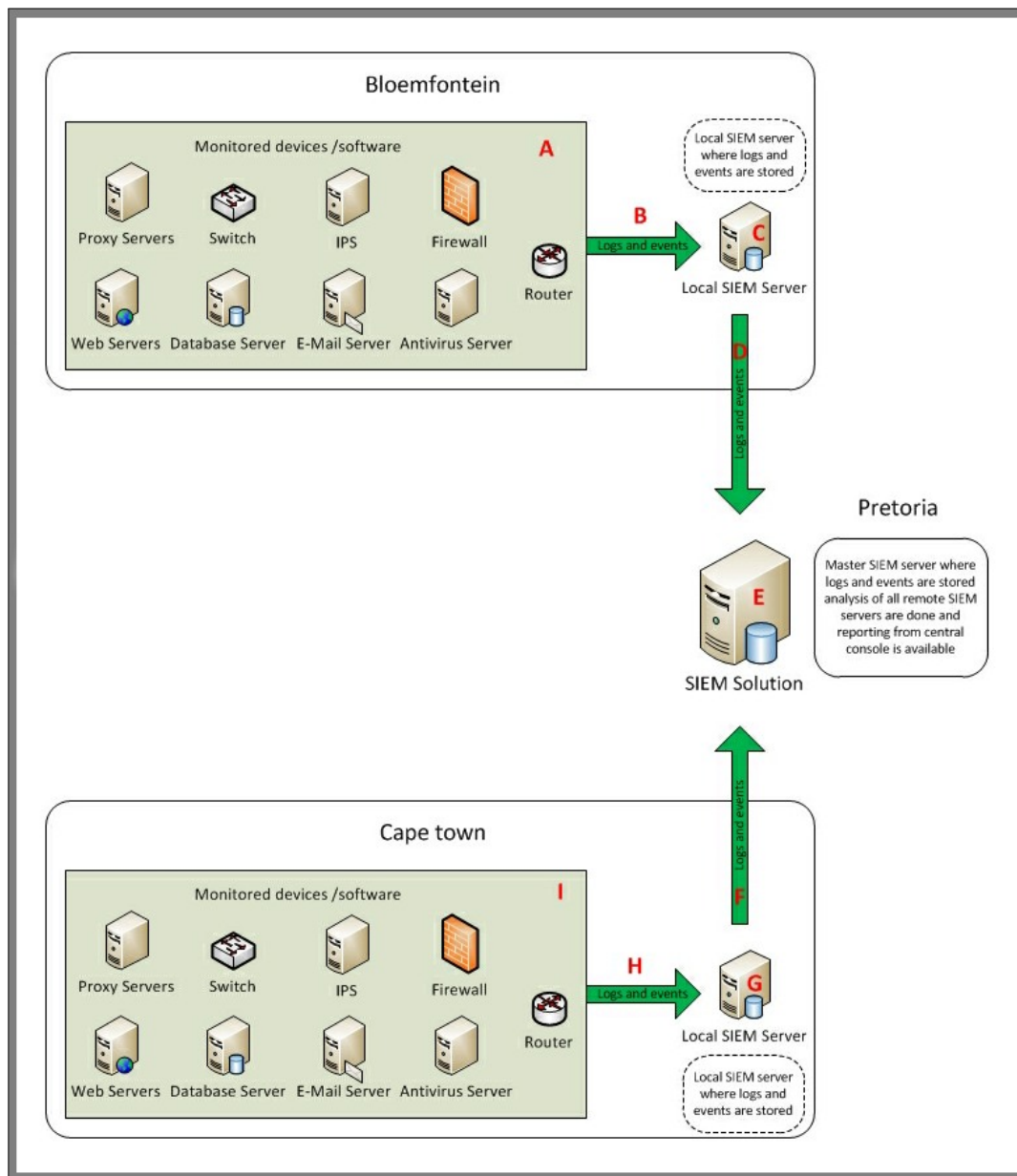


Figure 2.3: Distributed SIEM Architecture

2.6.3 Statistics with regards to SIEM

According to an article from govinfosecurity Chabrow [2012] “SANS found that 58 percent of organizations use a log manager to collect and analyse data; 37 percent employ SIEM systems in some capacity and 22 percent collect log data and process them exclusively with their SIEM systems”.

If 58 percent of the sampled organizations used some kind of log manager it follows that 42 percent of the sampled organizations have installed devices such as firewalls, IPS's and several others without any Log Management and Event Management (SIEM).

In the sampled data it seems that only 22 percent of the sampled organizations are using Security Information and Event Management (SIEM) correctly.

Taking the above statistics into account, less than 87 percent of the sampled organizations will not be able to comply with compliance audits.

Further, in “Magic quadrant for security information and event management” [Nicolett and Kavanagh, 2011] it is stated that as per the survey by Ernst & Young 80% of organizations that are implementing SIEM are doing so to comply to legislation.

2.7 Criteria for a good SIEM system

A presentation by Dr. Anton Chuvakin [Chuvakin, 2011] defines five of the best practices as well as the five worst practices for choosing and implementing a SIEM. There are several other best practices, however, that should also be considered:

1. Create a functional specification of what is required for the solution.
2. What devices are used within the organization and which of these devices, operating systems, and system must be supported?
3. As part of the functional specification, the architecture team must agree with the security specialists and decide on the collection mechanisms that must be supported, for example SNMPv3, syslog, but not third party agents.
4. How user-friendly is the solution and is the rules customizable?

5. What are the limitations in regards to the storing of data, and is it scalable? This is crucial as event data can be Gigabytes of data per device per day. Storage of events complying to legislation and organization policies is important due to the fact that if data is not in the database it can not be analysed or reported on.
6. Make sure that the solution complies with the organization's patch management policy; if not, the policy must be updated.
7. One of the most crucial parts of the solution is date and time. In cases where an organization must collect data from different time zones the solution must be able to handle time zone offsets.
8. Scalability: The solution must be able to process the collected events from all devices at a required rate.
9. Expandability must be considered as more devices are installed on a yearly basis and the number of logs that are logged is increasing by the day.
10. The security requirements must be part of the functional specification. The security requirements will cover how data is collected (whether encrypted or not), what triggers must be available, what mechanisms are required for event escalation and management up to the segregation of duties within the solution.
11. Scalability, in the sense of integrating the alert management into a current ticketing system.
12. The ability to add custom reports and triggers.
13. Depending on legislation, organization policies and archiving standards, data must be kept for X amount of years. The ability to archive data must be a default feature.
14. Security role work-flow must be customizable or fit into a given organization's processes as defined in the functional specification document.
15. Data integrity must be guaranteed depending on organizational policies and functional specifications.
16. Customization of the solution by having the capability to programmability add features.

2.8 FreeBSD

This section will describe what FreeBSD is, where it is utilised in the digital world as well as some of the functionalities offered. FreeBSD is a well known Unix-like operating system that is freely available for download and usage thereof. Many Internet companies such as Yahoo run mostly on FreeBSD due to the stability and performance FreeBSD offers [Lucas, 2007, Goodkin, 2014].

FreeBSD can be installed as the base operating system, and ports can be installed that will enable services for your organization, such as

- Firewalling
- Proxy server
- Webserver (HTTP & HTTPS)
- Database (MySQL)
- Time server (NTP)
- Domain Name server (DNS)
- File server (FTP, FTPS)
- E-Mail mail relay server (Exim)
- DHCP server
- TFTP server
- Syslog server

2.9 Packet filter

In this section packet filter (PF) will be discussed. Since the launch of PF in 2001 by OpenBSD⁹, it has become one of the most powerful free tools used for firewalling, traffic management as well as with load balancing by combining PF with CARP and pfsync [Hansteen, 2014]. The main functionality of PF is to analyse network traffic and match the packets received against the filtering criteria that were configured. The criteria can be a combination of the following:

⁹<http://www.openbsd.org/>

1. Source address: The IP address of the source from where the session is initiated.
2. Destination address: The IP address of the destination IP to where the session is initiated.
3. Protocol: The protocol used to the connection such as Hypertext Transfer Protocol (HTTP).
4. Source port: The source port number from where the session is initiated.
5. Destination port: The destination port number to where the session is initiated such as 80/tcp that are used as a standard for the HTTP protocol.
6. Interface: The network interface where the packets will enter or exit.
7. Direction: The direction in which the packets will be entering or exiting the interface; as an example, an HTTP packet will arrive at the Internet interface em1, the packet direction will be in on em1 and out on em0 where em3 is the interface connection to the network where the HTTP server is hosted.

The PF kernel is packaged as part of the operating system kernel adding an advantage of faster processing (speed) to PF. FreeBSD first added PF into the base system from version 5.3, 2004 and from there it has been part of FreeBSD. Other BSD flavours such as DragonflyBSD¹⁰ and NetBSD¹¹ have also included PF.

2.10 Digital forensics

Combining terminology already discussed, digital forensics describes a proven scientific technique that is used to investigate digital evidence.. Typical examples of digital evidence are cell phones, hard drives, etc. Digital forensics are well known in the world of hacking, fraud, piracy, digital theft as well as in several other areas [Valjarevic and Venter, 2012].

An example of a digital forensics investigation begins when digital theft is identified and the authorities are informed of the crime. The digital forensics team will then investigate the reported crime by examining the crime that was committed, identifying, collecting and storing evidence. All of this however needs to be done in a legally accepted manner.

It should be clear now that the steps that must be followed by the digital forensics team (Collection, Examination, Analysis, Reporting) [Kent et al., 2006] will include but are not limited to:

¹⁰<http://www.dragonflybsd.org/>

¹¹<http://www.netbsd.org/>

1. Compliance with legal processes for the collection.
 - (a) Identification of the trail of evidence (digital data sources) as well as where the possible digital evidence may reside.
 - (b) Authorisation by the owners and users of the identified digital devices to proceed, or a warrant from court to do a search, seizure.
 - (c) Compliance with the legal requirements with regards to the manner in which digital evidence is collected and handled (chain of evidence).
2. Storing the digital content such as logs, hard drives and other identified media.
3. Analysis of the digital evidence.
4. Compiling reports that will be used as evidence during prosecution or civil suits.

2.10.1 Define forensic readiness

With the Internet growing on a daily basis we are constantly exposed to threats and attacks that cause increased risks for individuals. This applies to Government Institutions, public and private organizations as well as individuals. It is imperative that we respond to these attacks to ensure that any given organisations reputation as well as its assets are safe-guarded. To achieve this, evidence must be collected in a legally acceptable manner. Most attacks will lead to legal prosecution, therefore the manner in which evidence is collected, stored and analysed is crucial [Tan, 2001].

The main goal of forensic readiness is to make sure that an organization can collect usable digital evidence in a legally accepted manner, when the need arises, while minimising forensic cost during an incident [Rowlingson, 2004] to ensure a successful conviction of a criminal.

2.10.2 Necessity of an organisation to be digital forensically ready

The digital forensic process forensically collects digital evidence before and/or after the event in a legally accepted manner [Rowlingson, 2004, Burrows, 2013].

As we know, most organizations do not want to spend money on technology or on the creation of additional processes that does not provide a return on investment. An example of this is the

implementation of additional security measures, such as an intrusion prevention system, that will guard or warn the organisation against possible hacking attempts.

A typical response from management might be that there is no budget for a system. In these cases the day that a website is compromised the security section will be called, required to explain how and why the website was compromised. A usual response at this point is for management to request the implementation of what is necessary to prevent the breach in future [Cyfor, 2012].

2.10.3 Digital forensically ready preparation and components

In order to be forensically ready any organisation will have to:

Implement policies and procedures: This will guide the organisation's digital forensic readiness. Policies are key due to the fact that the policy defines the rules that guide future decisions in order to make sure that the organization can collect usable digital evidence in a legally accepted manner when the need arises. This will also help minimise forensic cost during an incident [Tan, 2001].

Do data classification: This will assist the investigators in the way that the data is treated by the organization after an incident occurred. There is a significant value (both monetary and organizational) difference between confidential data and secret data. Depending on organisational policies, certain data will be encrypted, such as Intellectual property (IP) data or organizational secret, whereas other data will be treated as public knowledge. [Carol et al., 2007].

Enhance access control: The implementation of two-factor authentication is a valid example. This may be to enter into an office or to log on to a system. By implementing two-factor authentication (an access card with a biometric fingerprint) linked to a trusted time source it can be proved beyond reasonable doubt proof that the individual did enter the building and had access to a specific area where an incident occurred.

Security zones: Physically zone your organization with access control. The zoning of the organisation will restrict access in levels to employee areas. By limiting access to certain areas, it will be easier to identify what went wrong or who did what when a crisis arises [Grobler and Louwrens, 2007].

Physical security components: Implement physical security components such as Closed-circuit Television (CCTV) cameras. CCTV is a critical part of any security implementation.

By recording access to the server room, linking the video footage to a trusted time source and using two-factor authentication organizations can provide evidence for a strong case where the legal representative must prove that the employee was not in the location at the time of the incident [Rowlingson, 2004].

Enable logging on all critical systems: Linking logging to a trusted time source and trusted storage will make available data on what transactions were done, when and by whom. This is critical evidence during a cyber investigation [Tan, 2001, Rowlingson, 2004].

2.10.4 Digital Forensics and data logs

An important question is where data logs fit into digital forensics. The answer to this question is that data logs are a critical source for any forensic investigation [Sadowski, 2010]. Data logs can also be utilised as digital evidence if properly managed and if the chain of evidence is not compromised.

2.11 Summary

Several definitions was made available to the reader with an explanation of how they integrate and complement each other. Overviews was given of work performed by other researchers working in the same field of. Digital forensics was covered due to the fact that logs often function as legal evidence. The importance of collecting and analysing logs were discussed. It is clear that all organizations must implement solutions such as a SIEM in order to enhance security as well as comply of policies and best practices.

In Chapter 3 the reader will be presented with the ways in which data was collected from an organization's firewall, how that data was stored, and how it was then transferred to a central server. The database layout will be discussed, along with the method for importing the data into the database. Chapter 3 will also provide the reader with an analysis and summary of this process, along with relevant material such as statistics on data compression - and its advantages and disadvantages - with regards to data collection and storage.

3

Data Collection

IN this chapter we will be focusing on the data that was used to compile this thesis. This chapter begin by defining how the data was collected. After the explanation of data collection we will be looking at the data structure, from the raw logs up to the stage where data is stored within the database.

For the purposes of this project data was collected only on the DMZ connecting the organization network to the Internet, via the use of several technologies as discussed in this Chapter. This examination will consider only PF logs from one sensor, however, due to availability and the huge amount of logs received from the firewall as per Figure 4.1. In order to identify only relevant data it was decided to concentrate on all blocked events, as these events should not enter or exit the organization network. Full pcap logs were excluded from this study due to the amount of data received (denied initial packet events in the logs) as per Table: 3.3.

In Figure 3.1 the reader is presented with the flow diagram for the data collection, data analysis and data presentation. When IP packets are received by the PF firewall each connection is analysed and, depending on the configuration, the connections are dropped, denied, reset or allowed. As an added function, depending on the configuration, the connections are either

logged or not. In our case all connections not allowed are dropped and logged as per A in Figure 3.1.

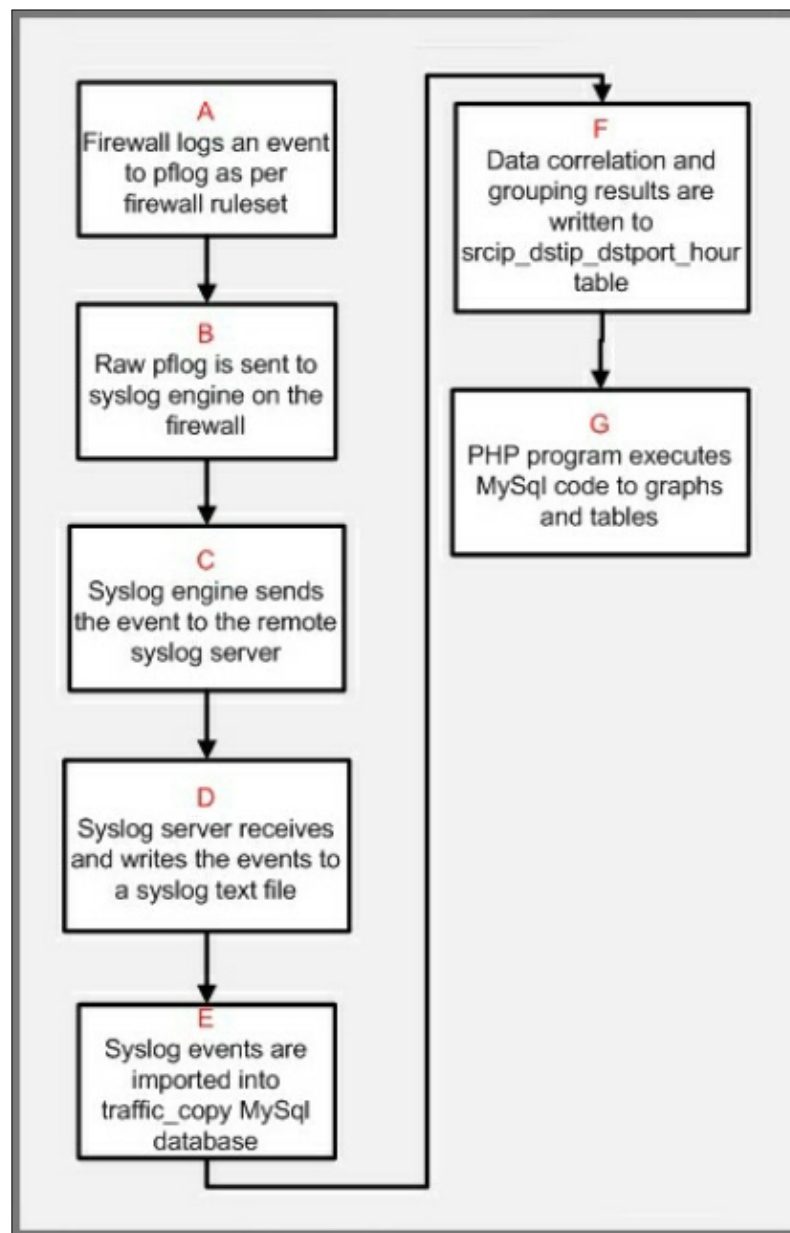


Figure 3.1: Data Collection Process

The next step is for the raw pflog to be sent to the syslog engine using logger as per B in Figure 3.1. In order for the syslog server to start during the boot process the configuration in the rc.conf file must be amended as per Code Listing 3.1.

The raw logs are processed and converted to text and sent to the syslog server. This forms part of tasks B and C in the process (Figure 3.1). The code used to accomplish the conversion of

raw logs to text to the syslog server is defined in Code Listing 3.2

Code Listing 3.1: BSD rc.conf configuration for Syslog

```

1 #/etc/rc.conf
2 syslogd_enable="YES"
3 syslogd_program="/usr/sbin/syslogd"
4 syslogd_flags=""           # Flags to syslogd (if enabled).
```

Code Listing 3.2: pflog to syslog

```

1 #!/bin/sh
2 #
3 /usr/sbin/tcpdump -n -e -i pflog0 | /usr/bin/logger -t dmz_fw -p local0.info -h 10.120.60.60↵
4 &
5 #
6 # tcpdump parameters:
7 # -n Do not convert addresses or port numbers to names
8 # -e Print the link level header on each dump file
9 # -i Stipulates the interface tcpdump must listen on
10 # logger parameters:
11 # -t Specifies that every line must be marked as a tag and not the default
12 # -p Specifies the priority and message
13 # -h Specifies the host the destination IP where the message must be sent
```

In Code Listing 3.3 a single syslog entry is displayed. Taking a closer look at the syslog entry the reader is made aware of the following important fields used within this thesis, which are displayed within this example syslog entry:

1. Date: Specifies the date the event was triggered.
2. Time: Specifies the time the event was triggered.
3. Firewall IP: The IP of the Firewall where the event was triggered is added to the log entry in order to identify firewall in question.
4. Action: Specifies the action the firewall enforced, defined by the PF rules when the event occurred.
5. Direction: Specifies the direction the entry was logged, in or out as an example.
6. Interface: The interface defines on which interface the event was triggered and logged.
7. Source IP: Specifies the IP of the source devices initiating the connection.

8. **Source Port:** Specifies the source port of the Source devices initiating the connection. The source port is also used to track a session.
9. **Destination IP:** Specifies the IP of the destination devices to which the Source IP is initiating a connection.
10. **Destination Port:** Specifies the destination port of the service to which the source IP is initiating the session, for example, a web server running 80/tcp.
11. **Protocol:** Specifies the system the service is running on the destination IP. An example of a well known protol running on the Internet is HTTP as a web server.

With the information available within the syslog entry as per Code Listing 3.3, the event can be tracked from where the connection was initiated through to the destination, where the event was logged, what protocol was used and at what date and time the event was logged or triggered.

Code Listing 3.3: BSD Syslog entry

```

1 # syslog entry
2 Aug  2 00:00:00 10.10.225.33 00:14:05.264637 rule 0/0(match): block in on em1: ←
    10.101.203.146.3994 > 218.244.158.205.4949:  tcp 20 [bad hdr length 8 - too short, < 20]

```

The next step is for the syslog server to receive the syslogs as per D in the Figure 3.1 and write the data to a text file. In this case the logs are rolled over every 24 hours after the logs are imported into the Visualisation database as per Code Listing 3.4. The configuration for the writing of data to a text file is displayed in Code Listing 3.5

Code Listing 3.4: BSD newsyslog.conf configuration

```

1 #/etc/newsyslog.conf
2 # logfilename [owner:group] mode count size when flags [/pid_file] [sig_num]
3 /data/syslog/dmz/GP-DMZ-fw.log 600 364 * @T00 JC

```

Code Listing 3.5: BSD syslog.conf configuration

```
1 #/etc/syslog.conf
2 # DMZ Firewall logs
3 +10.10.10.10
4 *.* /data/syslog/dmz/GP-DMZ-fw.log
```

3.1 Importing of logs into the database

There are two options that can be used to import the syslog data as per E in the flow Diagram 3.1 into the MySQL database.

1. Option 1: If only one syslog file is to be imported into the MySQL database, the shell script as per Code Listing 3.6 can be edited and executed
2. Option 2: Where multiple syslog files are to be imported into the MySQL database, the shell script as per Code Listing 3.7 can be executed in order to import all syslog files in the directory as per Code Listing.

Code Listing 3.6: Single log file import into DB

```

1 # Single log file import into DB
2 #!/bin/sh
3 cat /home/dirkc/logs/2014/07/GP-DMZ-fw.log.0 | /usr/local/www/apache22/data/import_logs/↵
   pfanalyse4.pl -y 2014 -d DMZ -v

```

Code Listing 3.7: Batch log files import into DB

```

1 # Batch log files import into DB
2 #!/bin/sh
3 for i in /home/logs/2014/07/GP-DMZ-fw.log.* ; do cat $i | /usr/local/www/apache22/data/↵
   import_logs/pfanalyse4.pl -y 2014 -d DMZ -v ; done

```

The Perl script and syslib file that was used to import the syslog file into the MySQL traffic_copy table within the traffic database are displayed in Code Listings 6.2 and 6.3

3.2 Summarization of data entries

After the logs are imported into the traffic_copy table within the traffic database, a SQL query as per Code Listing 3.8 is initiated in order to summarise the data entries. A new table is then created with the SQL query, namely srcip_dstip_dstport_hour, where all entries are summarised per hour.

Code Listing 3.8: Summarization of data entries

```

1 DROP TABLE IF EXISTS `srcip_dstip_dstport_hour`;
2 CREATE TABLE srcip_dstip_dstport_hour AS
3 SELECT STR_TO_DATE(CONCAT(packetdate, ' ', MAKETIME(hr,0,0)), '%Y-%m-%d %H:%i:%s') AS dttime↵
   ,src_ip, dst_ip, dst_port, action, SUM(src_ip_total) AS sessions, protocol, d
4 evice
5 FROM
6   (SELECT DAY(packetdate) AS dy, HOUR(packetdate) AS hr, src_ip, COUNT(src_ip) AS ↵
   src_ip_total, dst_ip, COUNT(dst_ip) AS count_dst_ip, dst_port, protocol, action, date(↵
   pa
7 cketdate)
8 AS packetdate, device
9 FROM
10  traffic_copy group by dy, hr, src_ip, dst_ip, dst_port, protocol, action ORDER BY dy, hr, ↵
   dst_port,device) ALIAS
11 GROUP BY hr, dy, src_ip, dst_ip, dst_port, protocol ORDER BY packetdate, hr, action, ↵
   src_ip, dst_ip, dst_port, sessions, protocol, device desc;

```

3.3 Compression Evaluation

Data compression is a critical part of the storing and/or archiving of data and logs. It is important to note that the hardware used has an influence on the compression and decompression time, as well as on additional processing time. Due to the fact that when text files are compressed with a compression ratio of, for example, 1:16, a text file of 600MB in size without compression may be shrunk to 37MB with compression. The advantage of compressing text files for log retention is the fact that less storage space is required for files that are stored in compressed format. Sending a 1GB log file (text only) file from point A to B with a communication medium speed of 10Mbps may take up to 15 minutes (assuming that no other data is simultaneously transmitted on the communication medium). By compressing the 1GB file to 55MB with a compression ratio of 1:15, the compressed file can be transmitted over the same 10Mbps communications medium (again, assuming that no other data is simultaneously transmitted on the communication medium) in under a minute. The disadvantage of storing compressed text data is that when the data must be accessed the file must first be decompressed to memory or storage, which will necessarily increase the process time, depending on the hardware used.

For the purposes of this thesis, when logs are rotated by the system every 24 hours they are compressed using a compression tool named bzip2¹². The compression ratio for 5 compressed files, as well as their uncompressed and compression ratio, are listed in Table 3.1, with an average compression ratio for the 5 files of 1:15.51. This means that for each 15MB of uncompressed syslog data the compressed syslog file will be 1MB in size using BZ2.

¹²<http://www.bzip.org/>

Table 3.1: Text log compression ratio

File Number	Compressed size (MB)	Uncompressed size (MB)	Compression Ratio	Compression Time (seconds)	Decompressing Time (seconds)
1	105	1600	1:15.23	364	35
2	102	1700	1:15.68	386	34
3	37	609	1:16.45	145	12
4	95	1400	1:14.73	326	32
5	100	1500	1:15	341	33

During this research a large text file, uncompressed, was transferred over an organizational network. In Table 3.2 the data captured for the transfer is made available to the reader. The 4.7GB file was transferred from the source to the destination in 39 minutes, or 2340 seconds, at 4am in the morning. By taking a average compression ratio of 1:15 from Table 3.1, the file size should be 314MB. If we could transfer the same file over the network at 16Mbps the transfer rate should be 150 seconds. The additional time for compression should be 295 seconds, with 104 seconds for decompression. By adding up the compression time, decompression time and transfer time the total time for compression, decompression and transfer time was 549 seconds, or 9 minutes.

Table 3.2: Transfer speed and time

File Number	Size (MB)	Average Transfer speed (Mb/s)	Transfer Time (Minutes)
1	4720	16	39

3.4 Data Collected

Table 3.3 displays the data collected. By performing estimates on the data within the table it is shown that the average data collected per month is 7.6GB, with 85,885,111 syslog entries or rows in the MySQL database. The estimate for 1 year's worth of logs for one Firewall is 91GB, with 1,030,621,332 syslog entries or rows in the MySQL database.

Table 3.3: Firewall log data Collected

Description	Value
Date Range	2014-03-01 to 2014-04-30 2014-07-01 to 2014-07-31
SENSOR Count	1
DATA Collected over 3 months (INPUT SIZE)	23GB
DATA Collected over 3 months (INPUT ROWS)	257,655,334
DATA imported in to MySQL table traffic_copy (ROWS)	257,655,334
DATA imported in to MySQL table srcip_dstip_dstport_hour (ROWS)	7,723,195

3.5 Database Layout

In this section the database layout will be presented on a high basic level as this layout were only used as a proof of concept. For this database layout four tables were used, namely:

- country_codes
- ip_lookup
- srcip_dstip_dstport_hour
- traffic_copy

3.5.1 Table traffic_copy

The country_codes table structure is made available as (A) in Figure 3.2. Within this table all syslog data that was received is imported before any summarization is done. The table is then

used to build the `srcip_dstip_dstport_hour_structure` table after the summarization program is executed. By adding additional functionality, such as storing the hash value for each data entry before the data is imported, this method adds some data integrity to the system. This table consumes the most amount of disk space as all events are stored.

3.5.2 Table `srcip_dstip_dstport_hour_structure`

The Table `ip_lookup` structure is made available as (B) in Figure 3.2. Within this Table all summarized data - data used for the analysis and presentation of findings in a visual system - is stored. Data is summarised by counting how many times the same event occurred within a time interval of one hour. Only one of these events is saved within the Table `srcip_dstip_dstport_hour_structure` table, which also provides a count of how many times the event occurred within the hour.

3.5.3 Table `ip_lookup`

The Table `ip_lookup` structure is made available as (D) in Figure 3.2. Within this Table are several fields, including most significantly `ip_start`, `ip_end`, where country and city were stored. These fields within the Table, as well as fields in the Table `country_codes`, are used to identify the location of a identified IP as per Figure 4.10. The `ip_start` and `ip_end` are stored as integers by passing the IP address to a Mysql command, namely `inet_aton("IP Address")`, which then generates the numeric value. The IP address is displayed by sending the numeric value to the MySQL command `inet_ntoa(numeric value)`, which then generates the IP address.

3.5.4 Table `country_codes`

The `country_codes` Table structure is made available as (C) in Figure 3.2. Within this table the 2 letter international country code as well as the name of the country were stored. This table is mainly used to extract the country name from the table by providing the 2 letter country code in several of the SQL scripts used within the proof of concept system.

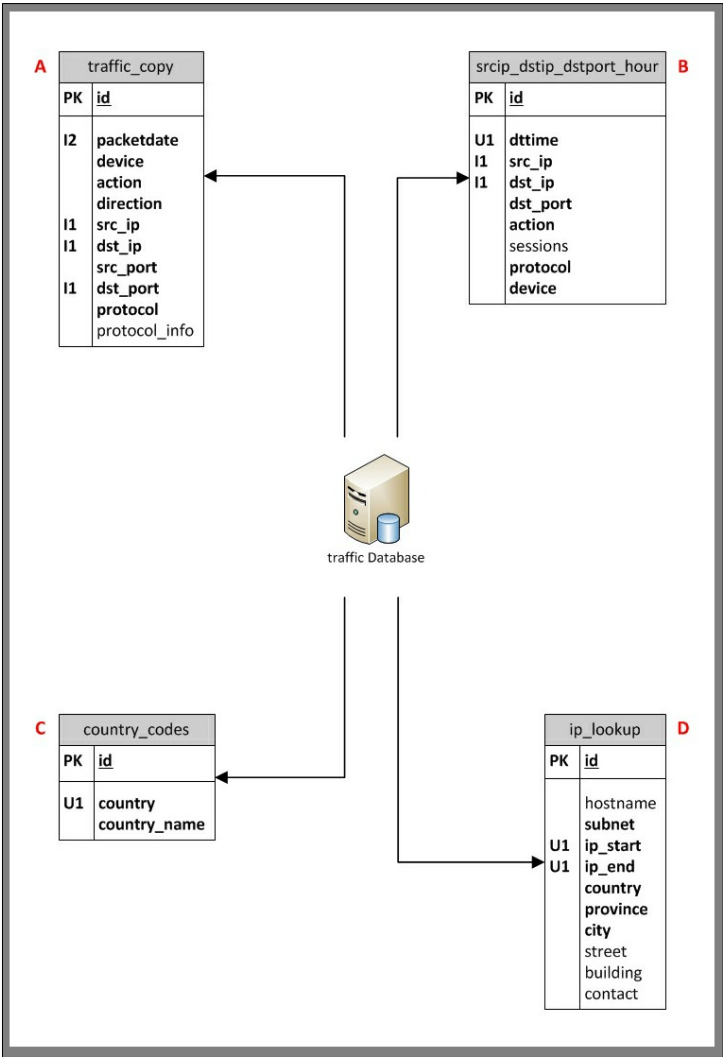


Figure 3.2: Database layout

3.6 Data Analysis

Several PHP programmes were developed to enable analysis of the enormous amounts of data. Figure 4.2 in Chapter 4 provides a visualization of this by means of graphs and tables generated during this project. MySQL is used as the database for storing FreeBSD pflogs running on FreeBSD as the operating system, and several tools are used such as jpgraph¹³ for graphing, Perl scripts, PHP, and SQL queries for data import and analysis.

¹³<http://jpgraph.net/>

3.7 Summary

Within this chapter the reader was presented with one of the several processes, software and programs that can be used to:

- collect logs
- transfer the logs to a central server
- import the logs into a database
- summarise data in order to improve data analysis, increase search time when searching for data, use less storage space and ultimately require less processing power

The use of compression was also discussed in Section 3.3, where the reader was presented with the statistics in Table 3.1, making the reader aware of the advantages and disadvantages of compressing logs. Compression was a crucial factor during the transfer of the logs and will be critical for any organization seeking to invest in a system such as the one presented within this proof of concept. By compressing the logs before sending the logs to the central server, transfer time was saved but additional time had to be spent in compressing and decompressing the logs at the source and destination.

In Section 3.3 the reader was made aware that the transfer time of a file from the source to the destination can be reduced by using compression before they are transferred. The disadvantage, as explained, is that there is additional time and processing power required for compressing and decompressing the file. Finally, this Chapter presented a clear proof of concept for collecting log data, transferring and importing it into a database.

In Chapter 4 the reader will be presented with the method for designing the Visualisation System, how the network architecture was implemented, and how the system functions.

4

Visualisation System

IN this chapter the reader is presented with the rationale behind the building of this system, the architecture of the proof of concept, as well an overview of its functionality. The researcher's own experience within Information Technology logs was used to both troubleshoot faults as well as complement the analysis of logs in order to determine how events such as, for example, a configuration change took place, or what data was accessed by whom and from where. The researcher spent on average between 290 and 410 hours analysing logs in order to find communication problems, identify unauthorised access attempts, and identify complex errors. Without logs it would be difficult, indeed near impossible to identify complex errors, unauthorised access attempts or traffic patterns running within a company network.

Companies install expensive server, network and security hardware and software with built-in features such as SNMP or syslog that can be used by network and security solutions to know what is going on within the network [Stallings, 1998]. As an Information Technology department you should at minimum be able to account for:

1. What physical assets and intangible assets you own.

2. What applications and services are operational on your network.
3. What information is communicated from your organization to the outside world.
4. What information is received into your organization.
5. What services / ports / information are blocked or allowed by your DMZ firewall.
6. When employees log on.
7. Alerts that must be raised when an employee authenticates to a system out of his or her approved working hours.
8. Alerts that must be raised when unsuccessful authentication takes place. A good example of when a alert should generate an event that must be investigated as a matter of urgency is where three unsuccessful attempts are experienced within sixty seconds for the same account.

Within the context of all of the above-mentioned aspects, logs are important as most information required will be extracted from logs, be it in text format, within a database or in a third party format.

The Graphing system was designed to store, analyse and visually present findings across several opensource¹⁴ technologies. Open source software is freely available and in some circumstances, their is a GNU license agreement applicable. Taking a look at the linux version named SUSE versus OpenSUSE as an example. New versions of SUSE are first made available as open source. After months of testing by the public, inputs and code changes the product will be released as OpenSUSE for the Open Source community and SUSE for the industry that requires maintenance and support in an enterprise environment. OpenSUSE is the foundation of SUSE [OpenSUSE, 2014].

4.1 Hardware and Software specifications

The specifications for the server used during the project are made available to the reader in Table 4.1

¹⁴<http://www.gnu.org/>

Table 4.1: Hardware and Software specifications

Item	Description
Motherboard	Intel DZ68BC
CPU	Intel(R) Core(TM) i5-2500 CPU @ 3.30GHz
Memory	14GB 1333MHz DDR3 synchronous memory
Storage	Seagate Barracuda, ST1000DM003-9YN162 1TB
Operating System	FreeBSD 10.0
Database	MySQL (amd64) 5.2
Web server	Apache 2.2.27

FreeBSD 10.0 was chosen as the operating system for the graphing system due to the low hardware specifications required and high performance [Larabel, 2012]. The operating system (kernel) can be compiled for the hardware used so that stability, availability and performance are increased. The choice of database for storing the logs for further processing was MySQL¹⁵ (amd64) [Larabel, 2013]. As a web server Apache¹⁶ was installed. This web server was used for hosting several PHP programs which were implemented to analyse and visually display the results.

4.2 Network and DMZ Architecture

In this section the network architecture of the case-study organization is discussed. An overview is shown in Figure 4.1. This architecture was chosen in order to have more control with regards to the traffic to and from the organisation VPN. A key factor of this architecture is its 'untrusted zone', which consists of the Internet as well as any Internet routers (R1 and R2) that are connected to the Firewall. Several FreeBSD firewalls with PF were installed in fail-over mode, forming the first line of defence into and out of the organisation's network. Due to the length and time constraints of this project, this thesis will not be considering other related security devices such as IPS, Anti Virus Gateways or mail content filterin. Finally, IPSec was chosen as encryption protocol for connectivity between offices and hosted server locations to protect confidentiality and integrity of data.

¹⁵<http://www.mysql.com/>

¹⁶<http://www.apache.org/>

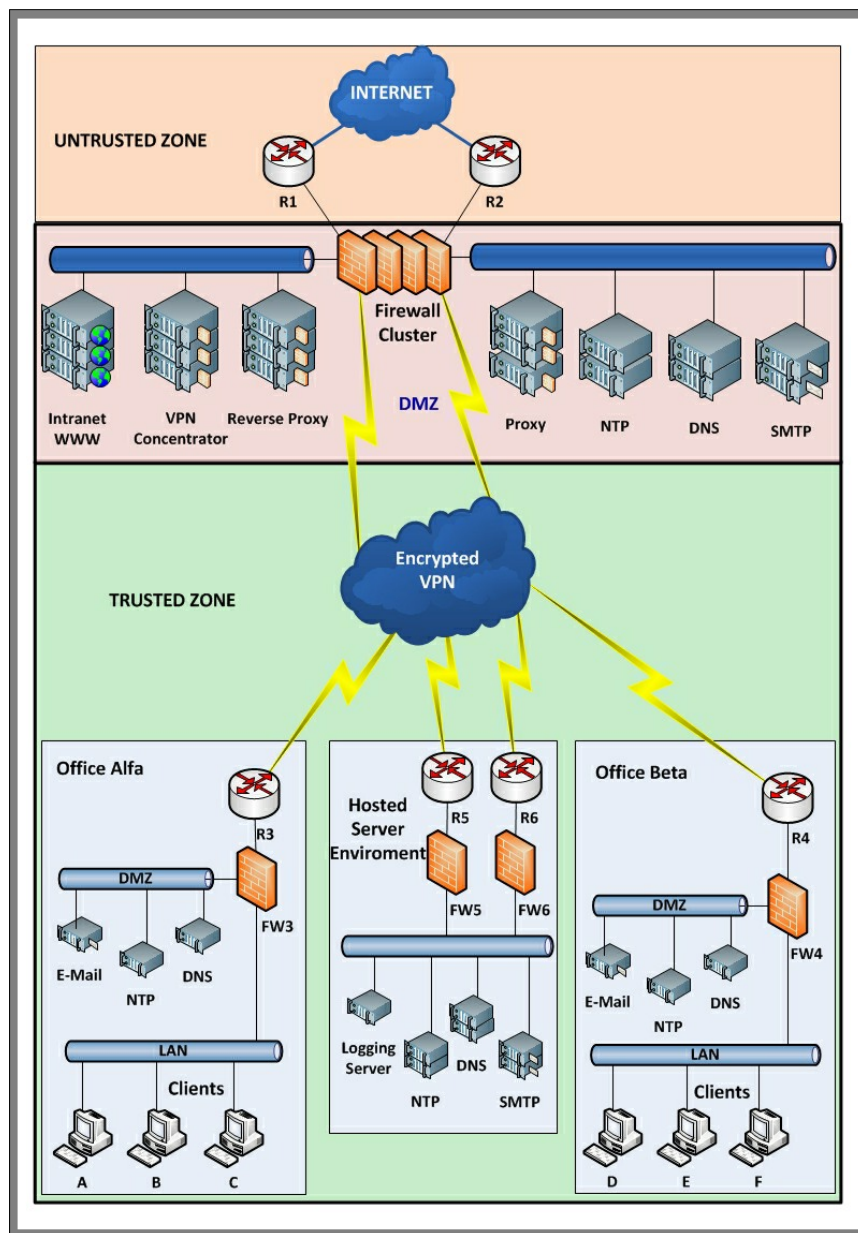


Figure 4.1: Network Architecture

4.2.1 Client connectivity to the Internet

By default companies allow DNS, HTTP, HTTPS and FTP to connect to the Internet due to the fact that most applications that are hosted on the Internet are Web-based. To translate a name to an IP, DNS is required. To have synchronised time NTP must be allowed. However, the implementation of these protocols and services varies from organisation to organisation; for example, some organisations will allow internet connectivity by implementing masquerading whereas other organisations will have authenticated proxy servers.

For the proof of concept, proxy servers were used as the only mechanism through which clients within the organisation network could access the Internet, using protocols such as HTTP, HTTPS, FTP. By using proxy servers such as Squid¹⁷, clients can be authenticated before they are allowed to access websites on the Internet, resulting in logs of who accessed what at a specified time. With the addition of URL redirector software such as squidguard¹⁸, content control is an added feature that allows the organisation to specify what websites clients may access.

4.2.2 Internet Connectivity to Web servers in DMZ

Reverse proxies were installed within the organisation's DMZ for all connections from the internet to web servers as a second line of defence for these web services. Several ACLs were used to allow only approved public IPs to access sensitive web servers behind the reverse proxies. As several web services were hosted behind the reverse proxies, they served as a major time-saver, logging all connections to these web services and caching data.

4.2.3 Host based firewalls

Web services that were to be accessed from the Internet were installed within the organisation's DMZ and protected by the above-mentioned reverse proxy servers within the DMZ. Each of the web servers were installed on a Linux¹⁹ version or FreeBSD, with either PF or iptables²⁰ running as the host firewall. Finally, the host firewalls were configured with "deny all" as the default rule and only allowed specified traffic.

4.2.4 DNS Traffic

DNS servers were installed within the organisation's DMZ in order for internal DNS servers and clients to be able to do DNS lookups. Without access to DNS servers clients would have to remember the IP addresses of all the services they need to access. DNS translates a friendly name - such as google-public-dns-a.google.com - to a IP address, namely 8.8.8.8, to which computer systems can then connect. This process takes into account the fact that users are much better at remembering names than IP addresses, and furthermore only DNS servers within the

¹⁷<http://www.squid-cache.org/>

¹⁸<http://www.squidguard.org/>

¹⁹<http://www.linux.com/>

²⁰<http://www.netfilter.org/projects/iptables/>

DMZ were allowed to query external DNS servers. As such port 53 was blocked for tcp and udp.

4.2.5 NTP Traffic

NTP servers were installed within the organisation's DMZ in order for internal NTP servers and clients to have their time in sync. Time is important both when using cryptography and also in forensic investigations as a means of proof that when an event occurred, it was at a specific date and time. Only the NTP server within the organisation's DMZ are allowed to communicate with external NTP servers.

4.2.6 SMTP Traffic

SMTP servers were installed within the organisation's DMZ. SMTP are mainly used for e-mail transmission, and only the SMTP servers within the organisation's DMZ were allowed to send and receive mail messages to and from SMTP servers on the Internet. Only internal e-mail servers were allowed to connect internally to the SMTP server for sending and receiving emails. Clients on the internal network would connect to Microsoft non-standard protocols in order to send and receive mails from their accounts on the organisation's internal Microsoft servers.

4.2.7 Remote management / Access

Remote Management from the Internet of the organisation's internal network was only allowed when using authenticated L2TP. The L2TP server is located within the DMZ. For each user a dedicated IP (static IP) was allocated from where the IP is firewalled, according to the approved change control application form and Remote access policy. Technicians that require remote access in order to work on the networks, or systems outside of the organisation's network is not allowed.

4.2.8 Firewall Rules

By default, all inbound and outbound is blocked. Traffic must be explicitly allowed in order to pass through the firewall. An example of a basic pf.conf file can be seen in Code Listing 4.1.

Code Listing 4.1: Sample PF firewall Rules

```

1 #REDIRECT RULE
2 rdr on $ext_if proto tcp from any to 164.151.0.0 port 3123 -> 172.123.188.165 port 3113
3
4 #LOG AND PASS TRAFFIC IN ON EXTERNAL INTERFACE
5 pass in log quick on $ext_if proto tcp from {196.1.1.1,41.1.1.} to ←
   {172.130.1.103,172.130.1.104} port 443 keep state
6
7 #LOG AND PASS TRAFFIC OUT ON INTERNAL INTERFACE
8 pass out quick on $int_if proto tcp from {196.15.211.170} to {172.130.1.103,172.130.1.104} ←
   port 443 keep state

```

4.2.9 PFLogs

The format of the log files generated by the pflog daemon is in binary format [Perrin, 2011], therefore reading these logs with a normal text editor is not possible. In order to convert the binary files to a human-readable format a tool such as tcpdump²¹ is used. The tcpdump tool prints out the content of packets as per a specified network interface. A shell script as per Code Listing 3.2 is running on the firewall as shown in Figure 4.1; using tcpdump and parameters this tool prints data from the pflog0 interface in human-readable format and then parses it to a tool called logger. Logger provides a shell interface for syslog. The parameters pflogs are sent to a specified host, see for example the MSC Server in Figure 4.1 running syslogd daemon.

Traffic originating from the Internet will pass through router R1 or R2, as per Figure 4.1, depending on the routing table, and will be inspected by the Firewall. Depending on the defined rule set the traffic will then be either allowed or dropped. All firewall actions are logged for deny rules as well as certain allowed rules.

4.3 Visualisation System Layout

Several menu items are made available in Figure 4.2. These items will execute SQL scripts on the server and display the results in a graph, table or text format. These menus and MySQL scripts was chosen as a proof of concept. The reader will now be presented with an overview of each of the menu items as per Figure 4.2.

²¹<http://www.tcpdump.org/>

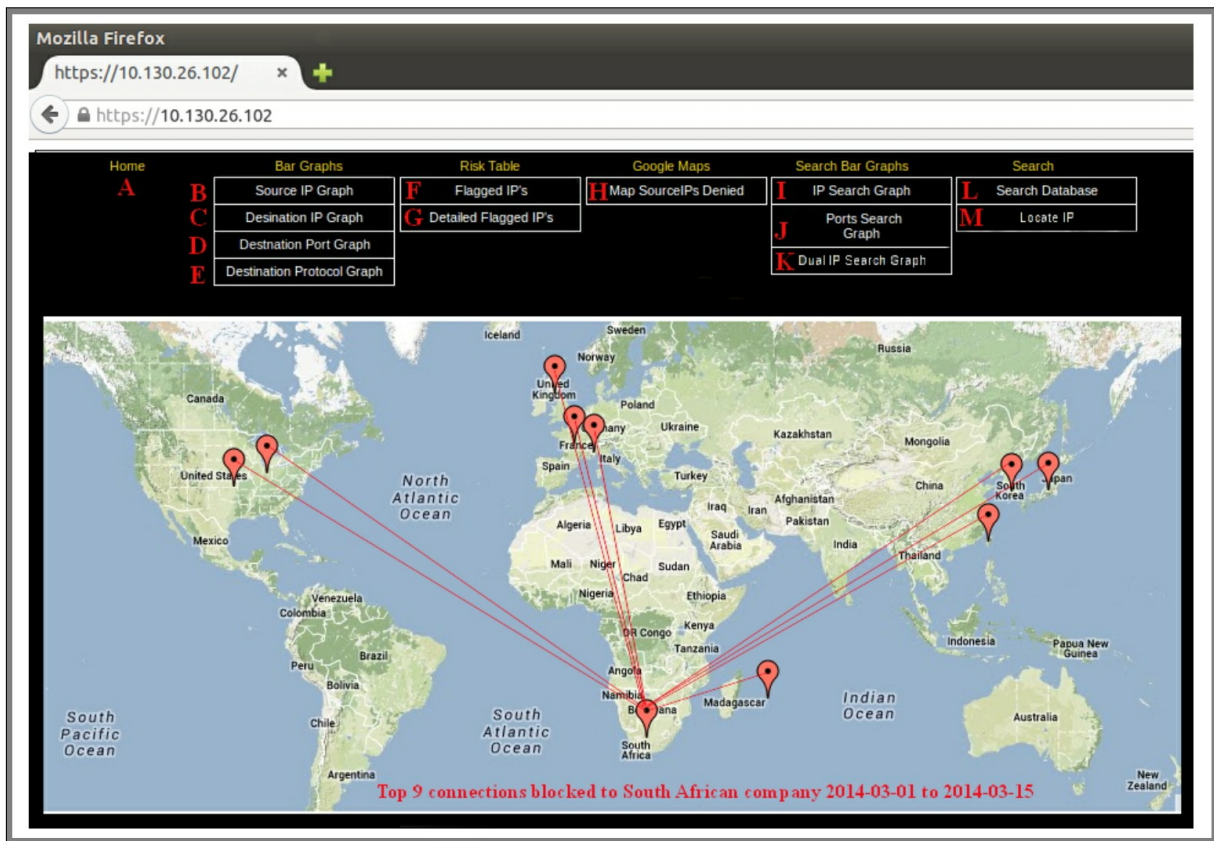


Figure 4.2: Visualisation System

4.3.1 Source IP Graph

Sub-menu item Source IP Graph, shown as (B) in Figure 4.2, will present a graph of the top 5 source IPs that were passed or blocked as per a selected date range. The SQL query in the back end will calculate the top 5, adding all events that were logged within the selected date ranges and actions.

The user will be presented with a form where the following can be selected:

1. Start Date: The year, month and date are made available in a drop-down menu.
2. End Date: The year, month and date are made available in a drop-down menu.
3. Firewall Action: There are only two choices available from the drop-down menu, namely **Pass** and **Block**.

4. Graph Filling: There are only two choices available from the drop-down menu, namely **Do not fill Graph** and **Fill Graph**. The first option will present the graph fields with lines and second will fill each graph field with the corresponding colour.

4.3.2 Destination IP Graph

Sub-menu item Destination IP Graph, shown as (C) in Figure 4.2, will present a graph of the top 5 destination IPs that were passed or blocked as per a selected date range. The SQL query in the back end will calculate the top 5, adding all events that were logged within the selected date ranges and actions.

The user will be presented with a form where the following can be selected:

1. Start Date: The year, month and date are made available in a drop-down menu.
2. End Date: The year, month and date are made available in a drop-down menu.
3. Firewall Action: There are only two choices available from the drop-down menu, namely **Pass** and **Block**
4. Graph Filling: There are only two choices available from the drop-down menu, namely **Do not fill Graph** and **Fill Graph**. The first option will present the graph fields with lines and the second will fill each graph field with the corresponding colour.

Destination Port Graph

Sub-menu item Destination Port Graph, shown as (D) in Figure 4.2, will present a graph of the top 5 destination ports that were passed or blocked as per a selected date range. The SQL query in the back end will calculate the top 5, adding all the events that were logged within the selected date ranges and actions.

The user will be presented with a form where the following can be selected:

1. Start Date: The year, month and date are made available in a drop-down menu.
2. End Date: The year, month and date are made available in a drop-down menu.

3. Firewall Action: There are only two choices available from the drop-down menu, namely **Pass** and **Block**
4. Graph Filling: There are only two choices available from the drop-down menu, namely **Do not fill Graph** and **Fill Graph**. The first option will present the graph fields with lines and the second will fill each graph field with the corresponding colour.

Destination Protocol

Sub-menu item Source IP Graph, shown as (E) in Figure 4.2, will present a graph of the top 5 destination protocols that were passed or blocked as per a selected date range. The SQL query in the back end will calculate the TOP 5 adding all the events that were logged between the selected date ranges and action.

The user will be presented with a form where the following can be selected

1. Start Date: The year, month and date are made available in a drop-down menu
2. End Date: The year, month and date are made available in a drop-down menu
3. Firewall Action: There are only two choices available from the drop-down menu namely **Pass** and **Block**
4. Graph Filling: There are only two choices available from the drop-down menu namely **Do not fill Graph** and **Fill Graph**. The first option will present the graph fields with lines in the graph and the second will fill each graph field with the corresponding colour.

Flagged IPs

Sub-menu item Flagged IPs, shown as (F) in Figure 4.2, will present a table of the selected top source IPs, destination IPs, destination ports or sessions that were **passed** or **blocked** and order the results **Asending** or **Desending** as per a selected date range.

The user will be presented with a form where the following can be selected:

1. Start Date: The year, month and date are made available in a drop-down menu.
2. End Date: The year, month and date are made available in a drop-down menu.

3. Firewall Action: There are only two choices available from the drop-down menu, namely **Pass** and **Block**
4. Order by: Select the field that must be used to calculate the top events. The choices are Source IPs, Destination IPs, Destination port or Sessions
5. Order: Display the top events by listing data Asending or Desending
6. Top: From the drop-down menu the variable for the amount of TOP events must be displayed can be chosen. The variable ranges from 5, 10, 20, 20, 10, 200, 500, and increments with 500 up to 2 000 000.

The reader is presented with an example of the **Flagged IP's** in Figure 4.3.

Pre-Defined: Custom Dates From: 2014 March 1 To: 2014 April 31 Firewall Action: Block Sessions Descending Top 20 > 100 000 Sessions GO					
Flagged IP's					
From 2014-3-1 to 2014-4-31, order by sessions, desc, list Top 20 records					
Counter	Action	src_ip	dst_ip	Dst_Port	Sessions
1	block	10.184.90.105	169.254.38.18	515	475929
2	block	10.132.128.134	69.43.161.172	33333	341538
3	block	10.201.217.8	109.74.196.143	447	338517
4	block	10.201.217.8	69.164.203.105	447	338420
5	block	10.201.217.8	91.233.244.106	447	338367
6	block	10.201.217.8	66.175.212.197	447	338216
7	block	10.95.188.167	69.43.161.172	33333	289583
8	block	10.99.45.22	65.55.56.206	123	289065
9	block	10.95.188.175	69.43.161.172	33333	260639
10	block	10.119.46.167	196.37.148.2	123	240125
11	block	10.119.46.167	196.21.187.2	123	239656
12	block	10.132.128.148	69.43.161.172	33333	236700
13	block	10.123.205.4	10.123.226.130	8087	217015
14	block	10.123.215.66	10.123.195.35	1070	215172
15	block	10.132.128.140	69.43.161.172	33333	208889
16	block	10.229.129.28	164.151.130.210	312	187824
17	block	10.123.215.66	10.123.195.35	1069	177922
18	block	10.123.215.66	10.123.195.35	1058	151078
19	block	10.229.129.4	105.228.146.209	22	144671
20	block	10.229.129.4	105.228.146.209	4111	144614

Figure 4.3: Flagged IPs

Detailed Flagged IPs

Sub-menu item Flagged IPs, shown as (G) in Figure 4.2 will present a table of the selected top source IPs, destination IPs, destination ports or sessions that were **passed** or **blocked** and order

the results **Asending** or **Desending** as per a selected date range. This menu item displays the details for the selected fields.

The user will be presented with a form where the following can be selected:

1. Start Date: The year, month and date are made available in a drop-down menu.
2. End Date: The year, month and date are made available in a drop-down menu.
3. Firewall Action: There are only two choices available from the drop-down menu, namely **Pass** and **Block**.
4. Order by: Select the field that must be used to calculate the top events. The choices are source IPs, destination IPs, Destination ports or Sessions
5. Order: Display the top events by listing data in ascending or descending order.
6. Top: From the drop-down menu the variable for the amount of TOP events that must be displayed can be chosen. The variable ranges from 5, 10, 20, 20, 10, 200, 500, and increments with 500 up to 2 000 000.

The reader is presented with an example of the **Flagged IP's** in Figure 4.7.

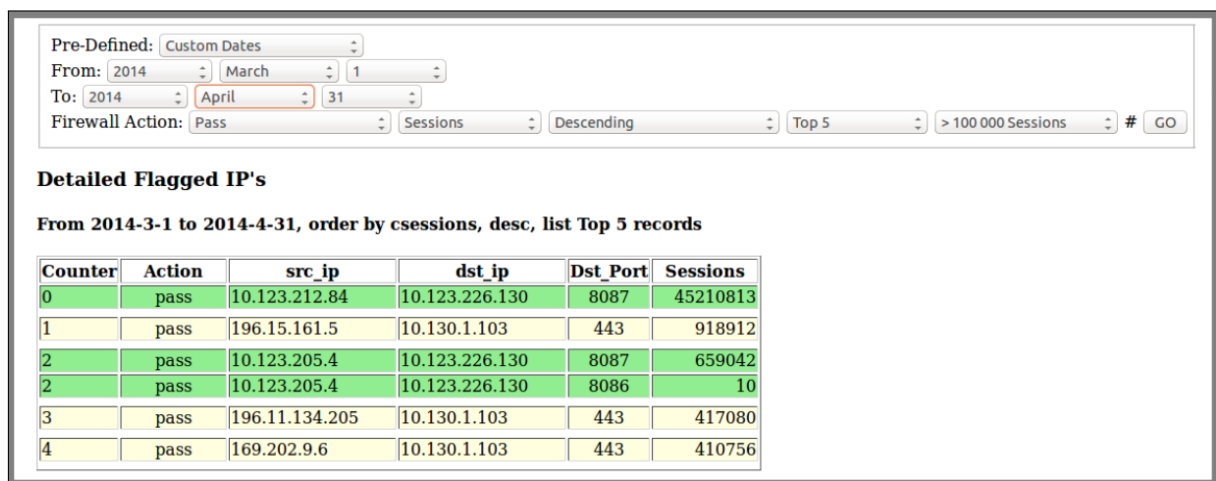


Figure 4.4: Detailed Flagged IPs

4.3.3 Google Maps

Sub-menu item Map Source IPs Denied, shown as (H) in Figure 4.2 presents a Google Map of the top source IPs, as per selection, that were **blocked** or **allowed** within either a selected

date range or in live data. This map displays the locations on a world map of blocked IPs. Please note, the map that is generated through this process is built using IPs logged by the DMZ firewall, although the original source may have 'jumped' from several other IPs. In the map displayed in Figure 4.2 the totals per country were removed due to sensitivity of the data.

The reader is presented with an example of the **Ports Search Graph** in Figure 4.5.



Figure 4.5: Google map

The Google Maps module can be used to even greater effect than its current utilisation by the proof of concept. If your private IPs are listed within the database and linked to cities, your internal communication can be displayed with the module as 'passed', 'blocked' or both along with the count for each city chosen. For the purposes of this proof of concept, however, a free Geoip database was downloaded and used, as discussed in Chapter 3.

The user will be presented with a form where the following can be selected:

1. Start Date: The year, month and date are made available in a drop-down menu
2. End Date: The year, month and date are made available in a drop-down menu
3. Top: From the drop-down menu the variable for the amount of top events to be displayed can be chosen. The variable ranges from 5, 10, 20, 20, 10, 200, 500, and increments with 500 up to 2 000 000.

4.3.4 IP Search Graph

Sub-menu item **IP Search Graph**, shown as (I) in Figure 4.2 presents a graph with the totals for ports that were **passed** or **blocked** for either the **Source IP** or **Destination IP** as per a selected date range. This graph will present the totals of ports were blocked.

The user will be presented with a form where the following can be selected:

1. Start Date: The year, month and date are made available in a drop-down menu.
2. End Date: The year, month and date are made available in a drop-down menu.
3. Source or Destination IP: From the drop-down menu there are two choices, namely **Source IP** and **Destination IP**. This field will be used to search within the database for the IP specified in the IP field.
4. Graph Filling: There are only two choices available from the drop-down menu, namely **Do not fill Graph** and **Fill Graph**. The first option will present the graph fields with lines, and second will fill each graph field with the corresponding colour.
5. IP: The Source or Destination IP must be typed into the IP field.

The reader is presented with an example of the **IP Search Graph** in Figure 4.6.

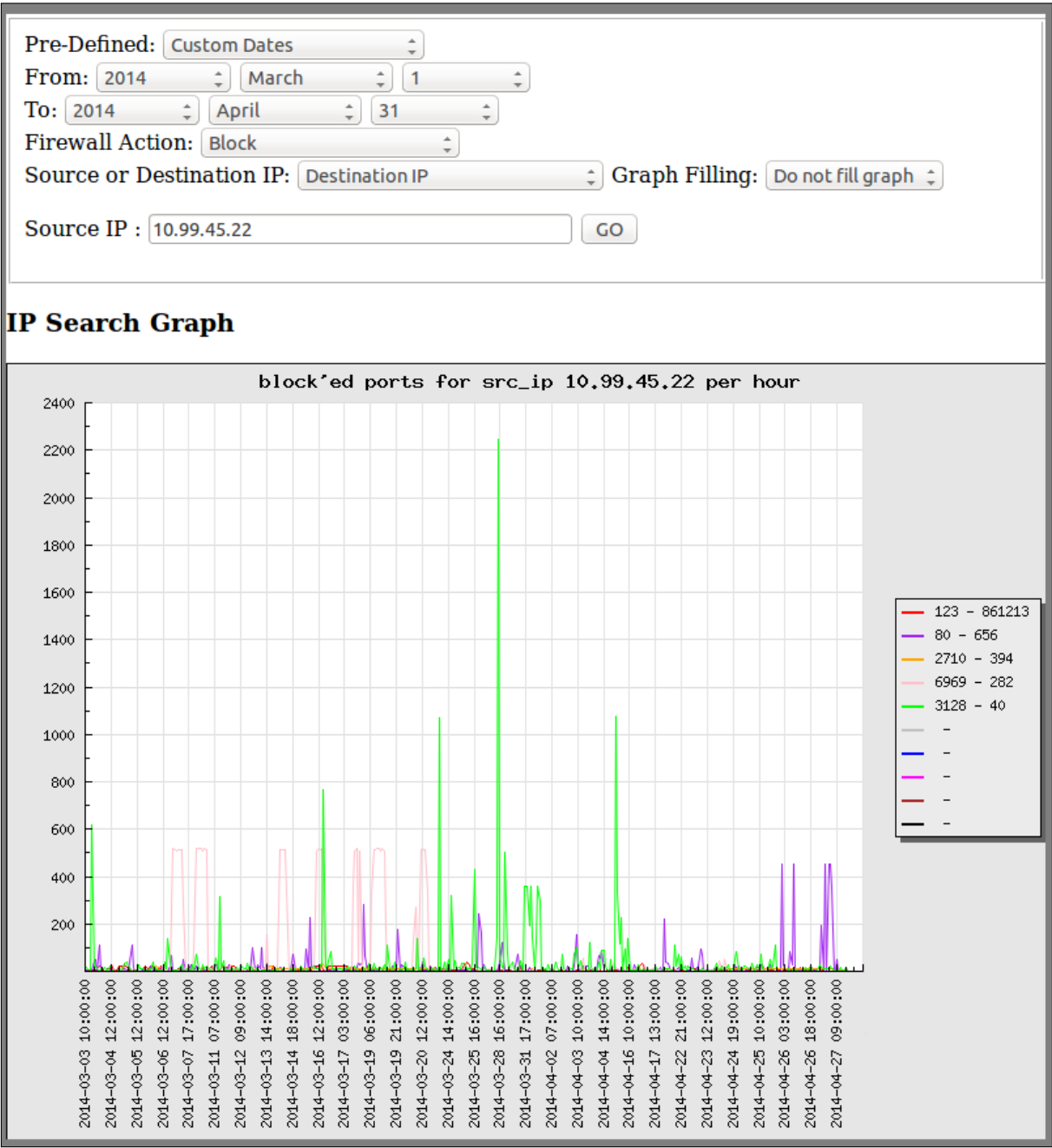


Figure 4.6: IP Search Graph

4.3.5 Ports Search Graph

Sub-menu item **Destination Ports Search Graph**, shown as (J) in Figure 4.2 presents a graph with the totals for IPs that were **passed** or **blocked** for the **Destination Port** as per a selected date range.

The user will be presented with a form where the following can be selected:

1. Start Date: The year, month and date are made available in a drop-down menu.
2. End Date: The year, month and date are made available in a drop-down menu.
3. Destination Port: The Destination Port must be typed into the Destination field.
4. Graph Filling: There are only two choices available from the drop-down menu, namely **Do not fill Graph** and **Fill Graph**. The first option will present the graph fields with lines and the second will fill each graph field with the corresponding colour.

The reader is presented with an example of the **Ports Search Graph** in Figure 4.7.

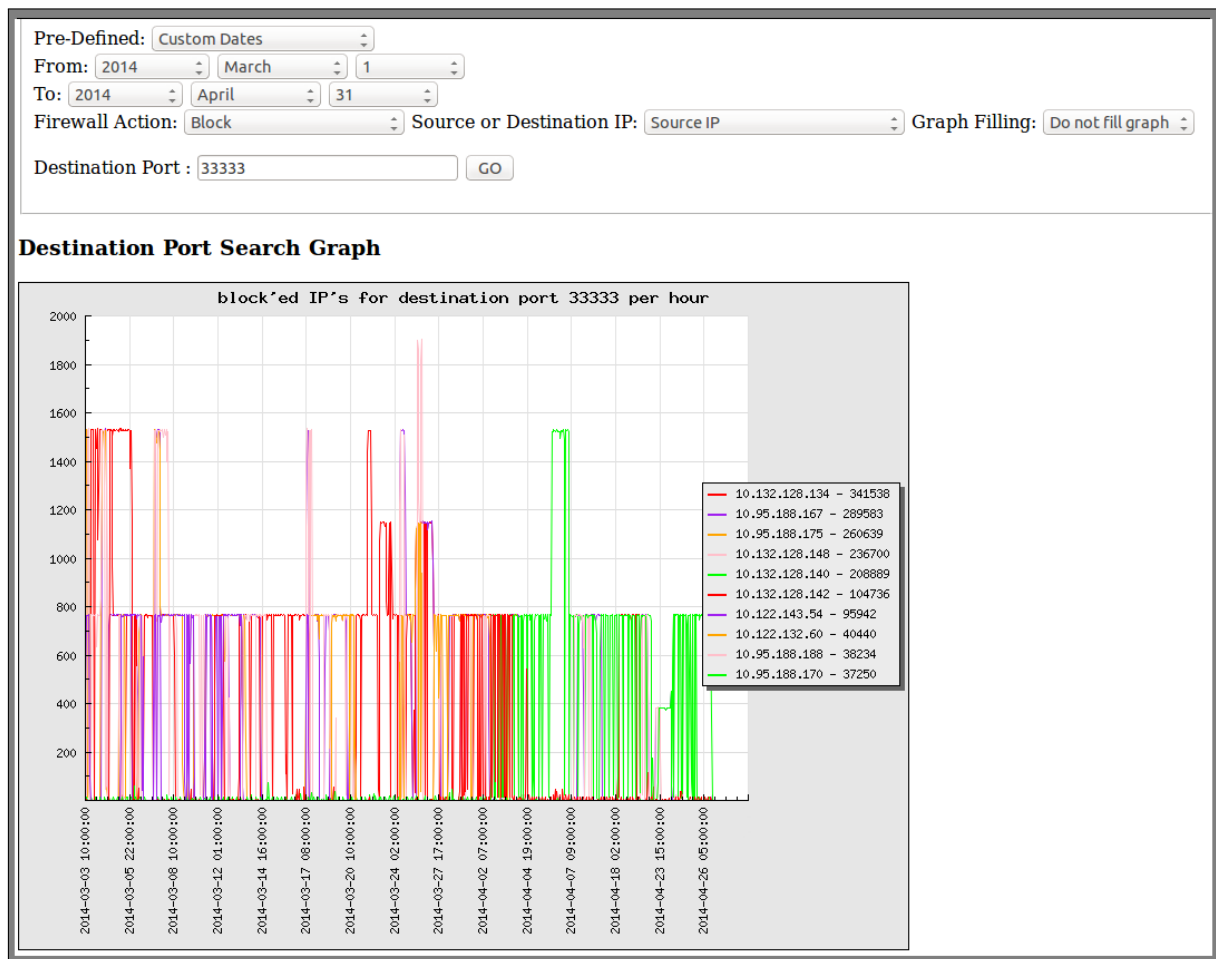


Figure 4.7: Destination Ports Search Graph

4.3.6 Dual IP Search Graph

Sub-menu item **Dual IP Search Graph**, shown as (K) in Figure 4.2 presents a graph with the totals for IPs that were **passed** or **blocked** for the **Destination Port** as per a selected date range.

The user will be presented with a form where the following can be selected:

1. Start Date: The year, month and date are made available in a drop-down menu.
2. End Date: The year, month and date are made available in a drop-down menu.
3. Destination Port: The Destination Port must be typed into the Destination field.
4. Graph Filling: There are only two choices available from the drop-down menu, namely **Do not fill Graph** and **Fill Graph**. The first option will present the graph fields with and the second will fill each graph field with the corresponding colour.

4.3.7 Search Database

Sub-menu item **Search database**, shown as (L) in Figure 4.2 presents a table for one of the following selection criteria:

1. Source IP
2. Destination IP
3. Destination Port
4. Source IP and Destination IP
5. Source IP and Destination IP and Destination Port
6. Source IP and Destination Port
7. Destination IP and Destination Port

The user will be presented with a form where the following can be selected:

1. Start Date: The year, month and date are made available in a drop-down menu.

2. End Date: The year, month and date are made available in a drop-down menu.
3. Graph Filling: There are only two choices available from the drop-down menu. namely **Do not fill Graph** and **Fill Graph**. The first option will present the graph fields, and the second will fill each graph field with the corresponding colour.
4. Order by: Select the field that must be used to calculate the top events. The choices are Source IPs, Destination IPs, Destination port or Sessions.
5. Order: Display the top events by listing data in ascending or descending order.
6. Top: From the drop-down menu the variable for the amount of top events that must be displayed can be chosen. The variable ranges from 5, 10, 20, 20, 10, 200, 500, and increments with 500 up to 2 000 000.

The reader is presented with an example of the **Search Database** in Figure 4.8.

The screenshot shows a web-based search interface. At the top, there's a 'Pre-Defined:' dropdown set to 'Custom Dates'. Below it, 'From:' is set to '2014' (year), 'March' (month), and '1' (day). 'To:' is set to '2014', 'April', and '31'. 'Firewall Action:' is set to 'Block', and 'Sessions' is also a dropdown. 'Order by:' is set to 'Descending', and 'Top' is set to '20'. A dropdown menu is open on the right, showing options: 'Src IP', 'Dst IP', 'Dst Port' (highlighted), 'Src IP & Dst Port', 'Dst IP & Dst Port', 'Src and Dst IP', and 'Src IP & Dst IP & Dst Port'. Below these are input fields for 'Source IP:', 'Destination IP:', and 'Destination Port:', followed by a 'GO' button.

Figure 4.8: Search Database

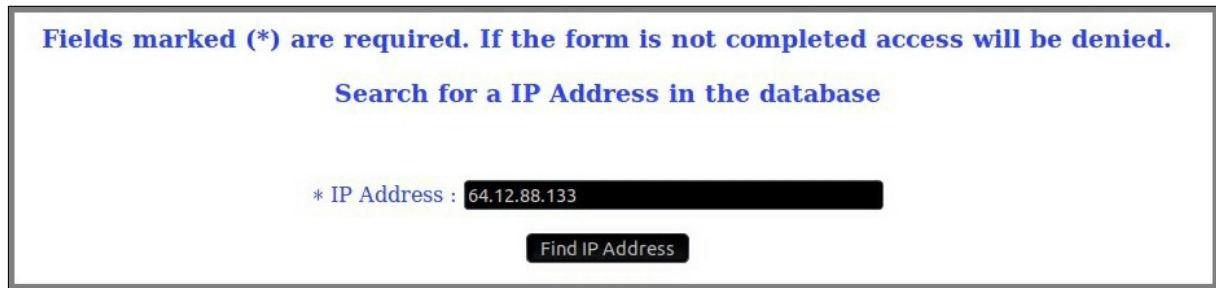
4.3.8 Locate IP

Sub-menu item **Locate IP**, shown as (L) in Figure 4.2 will present Country name, Country Code, Province and City.

The user will be presented with a form where the following can be selected:

1. IP Address (IP address is a mandatory field)
2. Find IP Address

The reader is presented with the search screen shown in Figure 4.9 and the results are presented as shown in Figure 4.10.



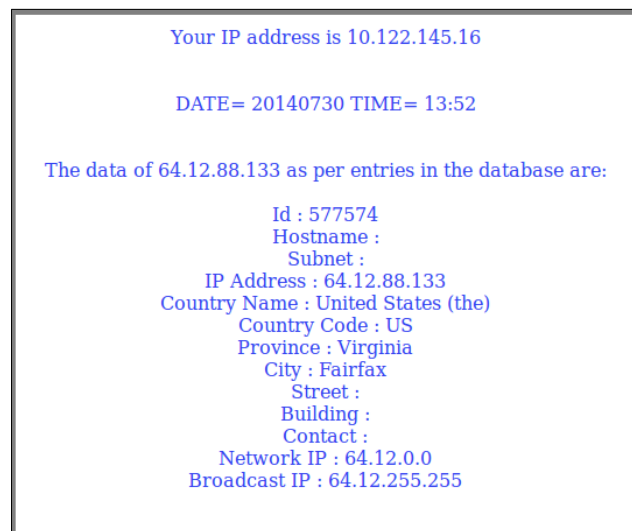
Fields marked (*) are required. If the form is not completed access will be denied.

Search for a IP Address in the database

* IP Address : 64.12.88.133

Find IP Address

Figure 4.9: Locate IP Query Screen



Your IP address is 10.122.145.16

DATE= 20140730 TIME= 13:52

The data of 64.12.88.133 as per entries in the database are:

Id : 577574
Hostname :
Subnet :
IP Address : 64.12.88.133
Country Name : United States (the)
Country Code : US
Province : Virginia
City : Fairfax
Street :
Building :
Contact :
Network IP : 64.12.0.0
Broadcast IP : 64.12.255.255

Figure 4.10: Locate IP Results Screen

4.4 Summary

This Chapter presented the reader with the proof of system that was developed by the researcher. The hardware that was available to the researcher was described, as well as the software that was used to develop the system. It was shown that the visualisation system was web based, allowing the system to be accessed from any IP network depending on network connectivity, firewall rules and organizational policies. Several drop-down menus were used to make navigation as easy as possible. Each of the menus were discussed in detail and examples were also made available.

The Google Maps menu, as displayed in Figure 4.5, was made known to the reader but not discussed in detail or used within the case study due to an agreement with both the researcher's supervisor and the owner of the organization from which data was collected. Due to the sensitivity of the logged data, the findings will not be visually displayed using the Google Map functionality as the map could be used to accuse a country of hosting, for example, a command and control server, and would therefore be excised from this thesis. However, it is important to note that Google Maps' functionality makes it possible to have a near real-time visual, mapping country names with a count of allowed or denied events, and even how many times any given country triggered a log rule on the firewall.

This chapter clearly presented that the proof of concept visually displays log data in a easy-to-use web-based application, making valuable data available and, crucially, accessible to the organization. This system is able translate the million lines of logs that are almost impossible for a human to use, presenting the data as graphs, maps and tables within seconds of any request for information.

Chapter 5 will present a case study (Case Study - PC infected with Malware) in detail, providing readers with an account of the process described by the researcher. Readers will be shown the process by which the Visualisation System was used to identify a computer infected with malware, worms and viruses.

5

Case Study - PC infected with Malware

THIS chapter demonstrates how the proof of concept visualisation tool was applied to real-world data and used to identify computers within the organization's network infected by malware. The steps taken to identify the source computers will be shown, and the destination IPs and ports that were used to identify incidents will also be discussed. Screen prints of actual findings are made available to the reader with an explanation of the significance on each.

The hosts were identified on the basis of the amount of denied and allowed attempts logged on the PF firewall, anomalies and observed ports. Criminals are getting smarter by the day and tend to use day-to-day encrypted protocols, namely HTTPS, to gain access to organisation networks or to steal data. As per the network architecture explained in Section 4.2, SMTP, HTTP, HTTPS, NTP and DNS traffic should not originate from the internal network. By monitoring all traffic logged by the DMZ firewall that originated from the internal network and connected directly to the Internet, the proof of concept identifies incorrect internal device configurations or applications.

The services and protocols listed in Table 5.6 are used for remote management, name resolution, file transfer or sending e-mails. Most of these ports, excluding 33333/tcp, are valid ports, commonly observed as part of legitimate Internet-facing traffic on a daily basis.

The method TCP utilises to set up a connection over IP involves a three way handshake, as shown in Table: 5.1 [Postel, 1981]. For any successful TCP connection the three way handshake must be complete; if not, the connection will not be established. It is important to understand how a successful tcp connection is established in order to understand tcpdumps as well as firewall logs.

Step No	Description
1	Host A sends a TCP SYN to Host B
2	Host B receives the SYN from Host A
3	Host B sends a SYN-ACK to Host A
4	Host A receives the SYN-ACK from Host B
5	Host A sends a ACK to Host B
6	Host B receives the ACK from Host A
7	Host A established TCP socket connection to Host B

Table 5.1: TCP three way handshake

As can be seen in Table 5.1, the successful implementation of a complete three-way handshake means very little chance of spoofing.

In this case study the reader is presented with the application of the system. The start of the system would be to request a list of the top 20 flagged IPs in order to identify the high risk sources and destinations. In Figure 5.1 the IP highlighted in red, namely 10.133.2.71, was chosen for the case study due to the fact that the source IP was online and the researcher could gain remote access to gather screen prints.

The next step would be to request a detailed report as made available in Figure 5.2. The detailed report calculates the amount of times, displayed as sessions, an event was logged for the same action taken, source IP, destination IP and destination port. The session count is displayed in this instance from the most to the least number of sessions.

From the detailed report three top ports were chosen to be further analysed within this case study. The description for the ports that were chosen available in Internet Assigned Numbers Authority (IANA)²²

The ports chosen for further analysis are:

²²<http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.txt>

- 25/tcp - Simple Mail Transfer
- 447/tcp - DDM-Distributed File
- 44416/tcp - Unknown

The next step was to generate graphs of the top 5 source and destination IPs for the three chosen ports. These graphs display the overall trend of events. With visualized data provided by graphs, the start of an event can be identified, along with the dates and times when events for the relevant ports are most frequent. As a result of looking at the trends, events and baselines, triggers can be developed that will raise an alert.

Lastly, analysis was performed via a remote session with the chosen IP; these findings are presented.

From here on the details of the findings will be made available with screen prints as evidence.

Traffic blocked on the DMZ firewall to and from the Internet was used for this case study. The internal IP, namely 10.133.2.71, was used for the case study of the top 20 identified flagged IPs list, as per Table 5.1 due to availability.

For this case study dates between 2014-03-01 and 2014-04-29 were used to gather information due to the availability of equipment and logs.

5.1 Top 20 Flagged IPs

The **Flagged IP** web application (F in Figure 4.2) was run, searching for the top 20 flagged IPs. The results that were found are displayed in Figure 5.1. Within this figure the chosen IP - 10.133.2.71 - is marked with the colour red. This figure clearly indicates that the IP chosen for this case study has software initiating 808989 sessions directly to the Internet that were blocked as per firewall policy and architecture design.

Flagged IP's					
From 2014-3-1 to 2014-4-31					
Counter	Action	src_ip	dst_ip	Dst_Port	Sessions
1	block	10.184.90.105	169.254.38.18	515	482402
2	block	10.201.217.8	109.74.196.143	447	419499
3	block	10.201.217.8	69.164.203.105	447	419317
4	block	10.201.217.8	91.233.244.106	447	419235
5	block	10.201.217.8	66.175.212.197	447	419048
6	block	10.132.128.134	69.43.161.172	33333	416110
7	block	10.99.45.22	65.55.56.206	123	388199
8	block	10.95.188.167	69.43.161.172	33333	349917
9	block	10.123.205.4	10.123.226.130	8087	306970
10	block	10.132.128.148	69.43.161.172	33333	287014
11	block	10.119.46.167	196.37.148.2	123	277431
12	block	10.119.46.167	196.21.187.2	123	276940
13	block	10.95.188.175	69.43.161.172	33333	260639
14	block	10.123.215.66	10.123.195.35	1070	215172
15	block	10.132.128.140	69.43.161.172	33333	208889
16	block	10.133.2.71	66.175.212.197	447	202319
17	block	10.133.2.71	91.233.244.106	447	202289
18	block	10.133.2.71	109.74.196.143	447	202233
19	block	10.133.2.71	69.164.203.105	447	202148
20	block	10.229.129.28	xxx.yyy.130.210	312	188522

Figure 5.1: Top 20 Flagged IPs

5.2 IP 10.133.2.71

The **Search Database** web application (L in Figure 4.2) searched for the top 20 ports that were blocked for source IP 10.133.2.71. The results for this search are displayed in Figure 5.2. It can clearly be seen in Figure 5.2 that IP 10.133.2.71 is trying to connect to several Internet IPs on different destination ports. These connections raised red flags due to the fact that the organization has no services hosted on any of the Public destination IPs.

Search Database					
From 2014-3-1 to 2014-4-31					
Counter	Action	src_ip	dst_ip	Dst_Port	Sessions
1	block	10.133.2.71	66.175.212.197	447	202319
2	block	10.133.2.71	91.233.244.106	447	202289
3	block	10.133.2.71	109.74.196.143	447	202233
4	block	10.133.2.71	69.164.203.105	447	202148
5	block	10.133.2.71	64.12.88.133	25	87949
6	block	10.133.2.71	64.12.88.165	25	87749
7	block	10.133.2.71	64.12.91.197	25	86419
8	block	10.133.2.71	141.8.225.62	44416	62346
9	block	10.133.2.71	195.22.26.252	44416	13970
10	block	10.133.2.71	195.22.26.254	44416	13896
11	block	10.133.2.71	195.22.26.253	44416	13865
12	block	10.133.2.71	195.22.26.237	44416	13832
13	block	10.133.2.71	195.22.26.231	44416	13804
14	block	10.133.2.71	209.222.14.3	44416	6984
15	block	10.133.2.71	xxx.yyy.130.210	3128	163
16	block	10.133.2.71	64.4.10.33	123	19
17	block	10.133.2.71	85.17.167.196	9832	9
18	block	10.133.2.71	192.155.89.148	8000	3
19	block	10.133.2.71	65.55.56.206	123	2

Figure 5.2: Detailed table of Case Study

The **IP Search Graph** web application (I in Figure 4.2) was executed, searching for the top 5 ports that were blocked for source IP 10.133.2.71, as per Figure 5.3. The top 3 destination ports were 447/tcp with 808 989 total sessions, 25/tcp with 262 117 total sessions and 44416/tcp with 138 697 total sessions during the date ranges specified.

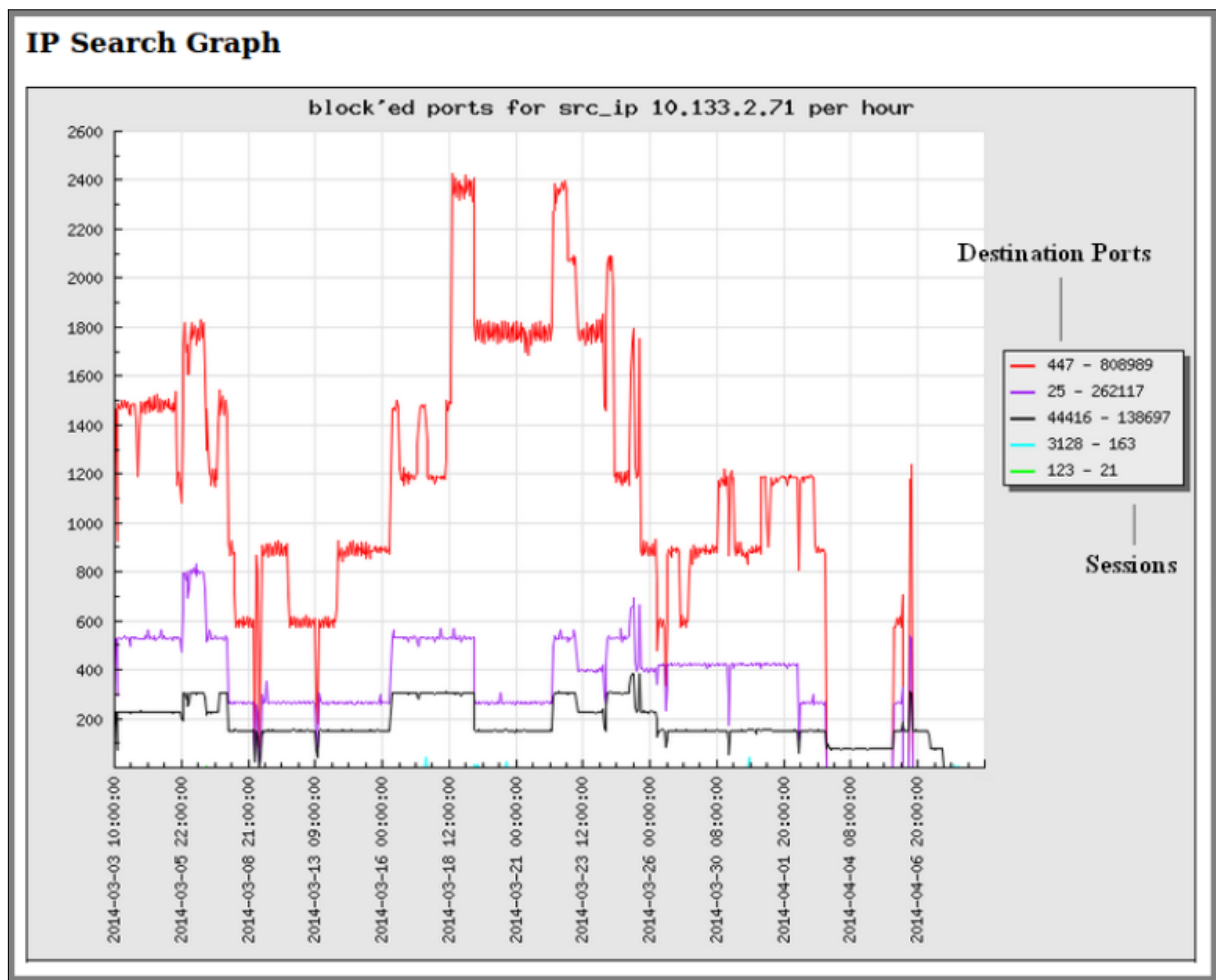


Figure 5.3: Summary graph for Case Study

5.3 Destination Port 25/tcp

A **Port Search Graph** web application (J in Figure 4.2) was run, searching for destination port 25 from 2014-03-01 till 2014-04-31 and graphing all source IPs as per Figure 5.4. The graphs lists the top 5 clients that were blocked initiating sessions to destination port 25 directly to the Internet. The top source IP was identified as a internal IP - 10.217.31.45 - with 310442 events that were blocked during the date ranges. The second IP in the top 5 was IP 10.133.2.71, with 287904 blocked events. Port 25/tcp is used to send emails, as discussed in paragraph 4.2.6. As per architecture design in Figure 4.1, all SMTP traffic must be sent to a internal mail server, which then sends the SMTP traffic to the recipient. Many viruses utilise SMTP to send collected information out of the internal network to a server, a public SMTP server from whence the information is sent to the recipient. Hence, Figure 5.4 shows that there are several IP devices

trying to connect to public SMTP servers and must be investigated.

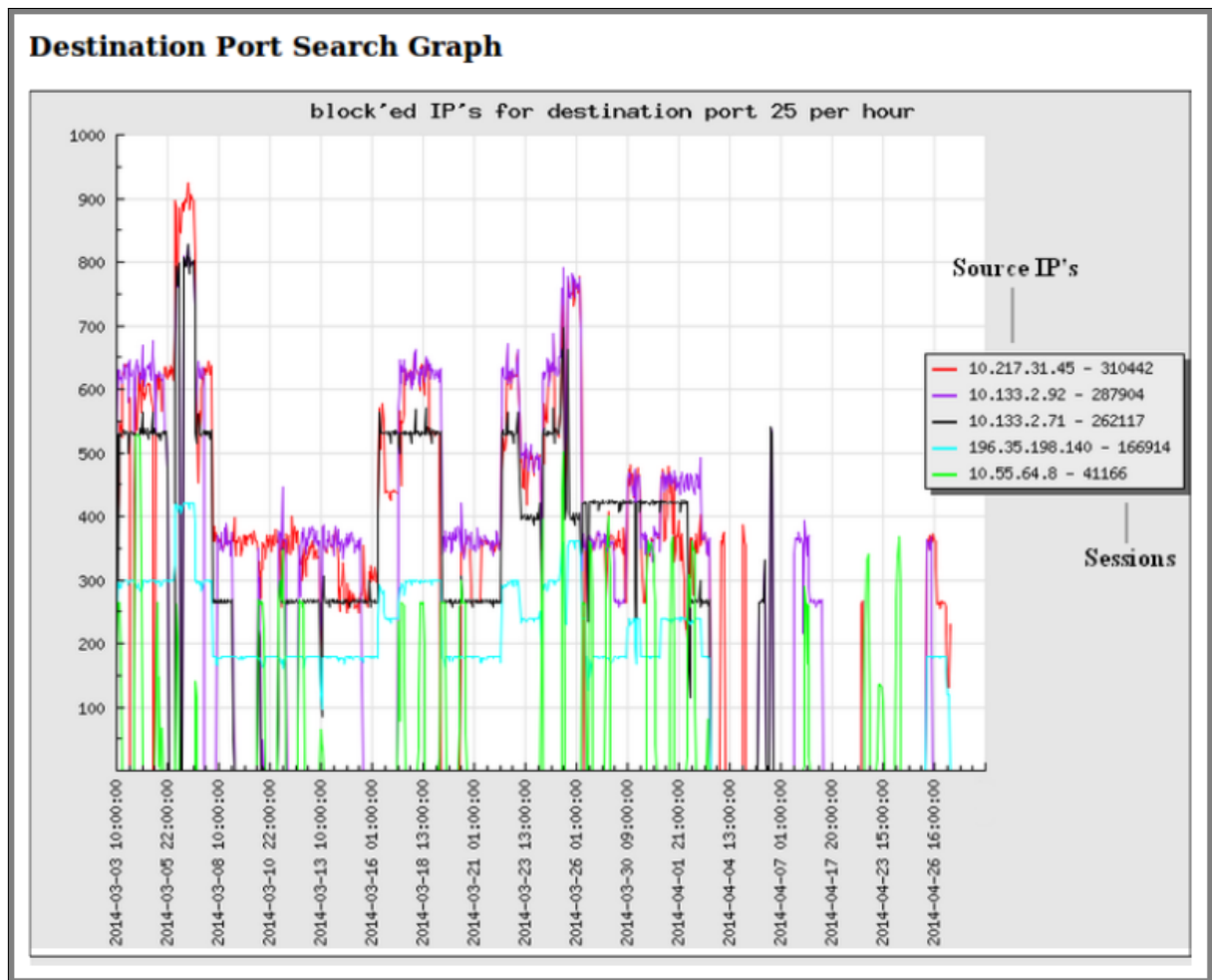


Figure 5.4: Destination port 25/tcp search graph - Source IPs

The **Port Search Graph** web application (J in Figure 4.2) searched for destination port 25/tcp from 2014-03-01 till 2014-04-31, graphing all destination IPs as per Figure 5.5. The number one IP was identified as 64.12.88.133, with 306 521 sessions blocked during the date ranges. All destination IPs are public IPs as per Figure 5.5. Looking at the first three IPs it can be seen that there is a possibility of the source IP being infected with Malware. The destination IPs listed in Table 5.2 were identified as possible Malware hosts by Totalhash²³.

²³<http://totalhash.com/>

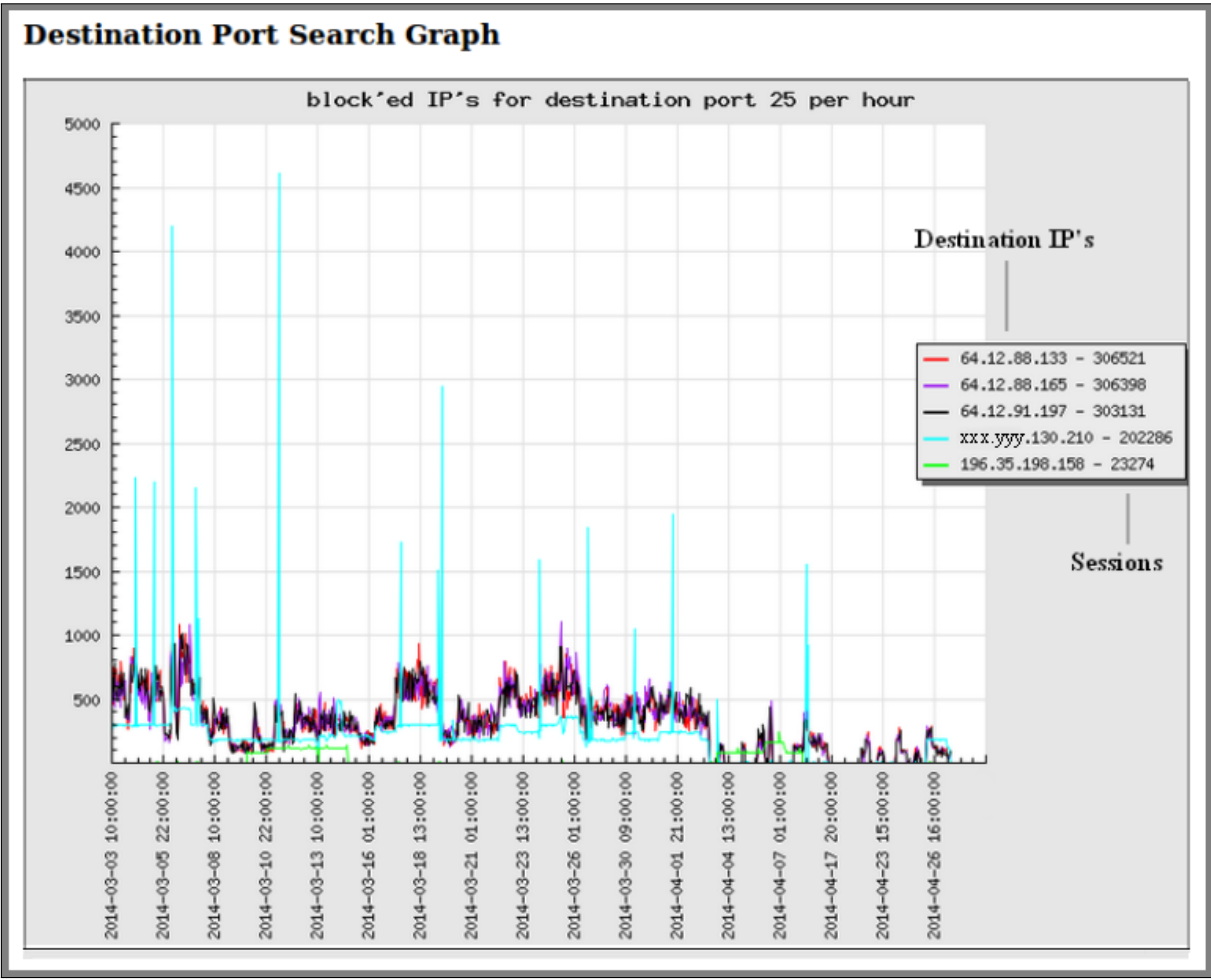


Figure 5.5: Destination port 25/tcp search graph - Destination IPs

Table 5.2: Destination Port 25/tcp Top 10 Destinations

Rank	Destination IP:port	Finding	SHA1 Hash
1	64.12.88.133:25	IP used by Malware	f90199796965685f319e77fe2d622e126ac2c51f
2	64.12.88.165:25	IP used by Malware	b1936dcb0008fdd5925f2c4076a2aadacacb3508
3	64.12.91.197:25	IP used by Malware	639bd3c59a3a2191647ab33210c1402822494add

An example of information found in regards to the Destination IP's as well as code are listed in Figure 5.6 from totalhash website. Information in regards to malware samples are made available by totalhash such as

- analysis date
- hash values

- malware names associated with malware
- network IP's and DNS names used by the malware
- infected files
- registry entries associated with the malware
- Raw Pcap data

The information can be used to identify malware on hosts as well as to understand how the flavour of malware works. In the case where a possible unlisted malware is found samples of the files be uploaded to totalhash and analysed.

ANALYSIS DATE	2014-08-03 11:39:14
MD5	a466e5c81ff38a19516323fe810eb065
SHA1	7eb8a53711f4da8d2139e033a5f2559ea128c16c
Static Details:	
FILE TYPE	PE32 executable for MS Windows (GUI) Intel 80386 32-bit
LANGUAGE	040904B0 
SECTION	.text md5: 36c6d5d4d7f4ee67306bfa301229ec49 sha1: a5066b772889451001a7c84a9b002c4e926259ca size: 40960
SECTION	.data md5: b73962902652b9081e5b45fdee43f056 sha1: 5adbcccc6effde1743823Eb9138b3a737759ab26e size: 4096
SECTION	.rsrc md5: a46acc64c2274c5b4e187c71d8efcd79 sha1: 1fd80c7e6614598b29458edf1b64202045dba25b size: 40960
SECTION	+& md5: 8c7197d865c3883bc4575be510fa9172 sha1: 78056fa1d0f25c44dabc3b288ce7a6148d3ddc5e size: 241664
TIMESTAMP	2001-07-19 22:01:47
VERSION	LegalCopyright: Copyright (C) Microsoft Corp. 1981-2000 InternalName: msn FileVersion: 6.10.0016.1624 CompanyName: Microsoft Corporation Built by: msnbld ProductName: Microsoft(R) MSN (R) Communications System ProductVersion: 6.10.0016.1624 FileDescription: msn OriginalFilename: msn.exe LegalCopyright: Copyright (C) Microsoft Corp. 1981-2000 InternalName: msn FileVersion: 6.10.0016.1624 CompanyName: Microsoft Corporation Built by: msnbld ProductName: Microsoft(R) MSN (R) Communications System ProductVersion: 6.10.0016.1624 FileDescription: msn OriginalFilename: msn.exe
PEHASH	64fc80f1bfec3a526065f0e5eda79e60d44166ef
IMPHASH	5002bceb823d3d7321ac4b2e8ee9f66d
AV	360 Safe Virus.Win32.Downloader.AL
AV	Ad-Aware Win32.Vladtre.3
AV	Alwil (avast) AutoRun-BSV [Wrm]:ladtre-A [Drp]
AV	Arcabit (arcavir) W32.Ramnit.i
AV	Authentium W32/Nimnul.A
AV	Avira (antivir) W32/Nimnul.C
AV	CA (E-Trust Ino) Win32/Wapomni.H5
AV	CAT (quickheal) W32.Nimnul.C
Network Details:	
DNS	www.a.shifen.com Type: A 180.76.3.151
DNS	www.a.shifen.com Type: A 180.76.3.151
DNS	175.ns768.com Type: A 192.42.116.41
DNS	175.nsvin987.com Type: A 192.155.89.148
DNS	175.nsvin987.com Type: A 195.22.26.231

Figure 5.6: Totalhash Output Example

5.4 Destination Port 447/tcp

A **Port Search Graph** web application (J in Figure 4.2) searched for destination port 447/tcp from 2014-03-01 till 2014-04-31 and graphing all source IPs as per Figure 5.7. The graphs list the top 5 clients that were blocked initiating sessions via destination port 447/tcp directly to the Internet. The top source IP was identified as a internal IP, namely 10.201.217.8, with 1 677 099 events that were blocked during the date ranges. The second IP in the top 5 was 10.133.2.71, with 808 989 blocked events. Figure 5.7 shows that there are several IP devices trying to connect to public IPs and must be investigated.

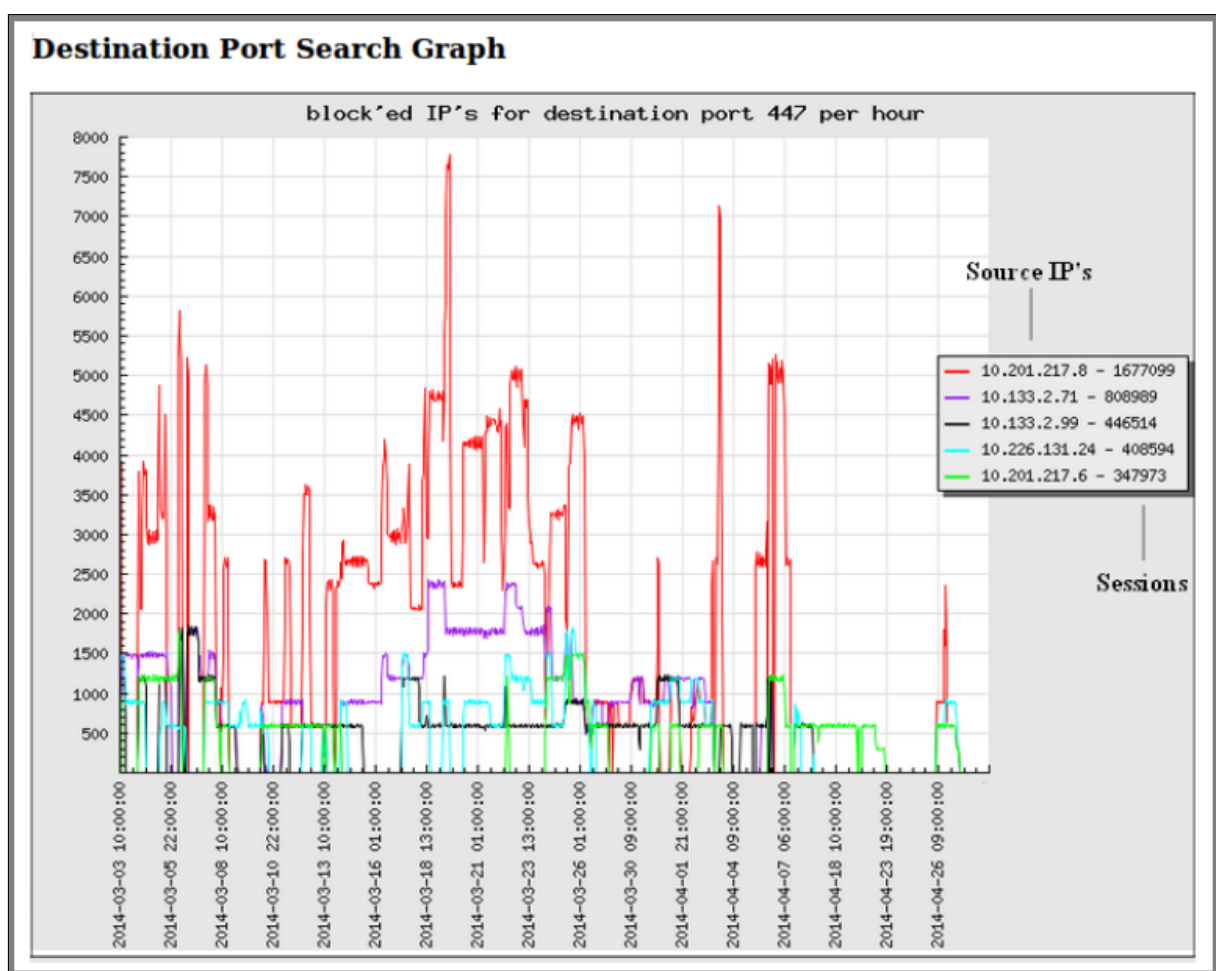


Figure 5.7: Destination port 447/tcp search graph - Source IPs

A **Port Search Graph** web application (J in Figure 4.2) was run searching for destination port 447/tcp from 2014-03-01 till 2014-04-31 and graphing all destination IPs as per Figure 5.8. These graphs list the top 5 clients that were blocked initiating sessions via destination port 447/tcp directly to the Internet. Top destination IP was identified as a public IP - 91.223.244.106

- with 1 094 017 events that were blocked during the date ranges. Please note that all destination IPs were public IPs. Looking at the first three IPs it can be seen that there is a possibility of the source IP being infected with Malware. The destination IPs listed in Table 5.3 were identified as possible Malware hosts by virustotal²⁴.

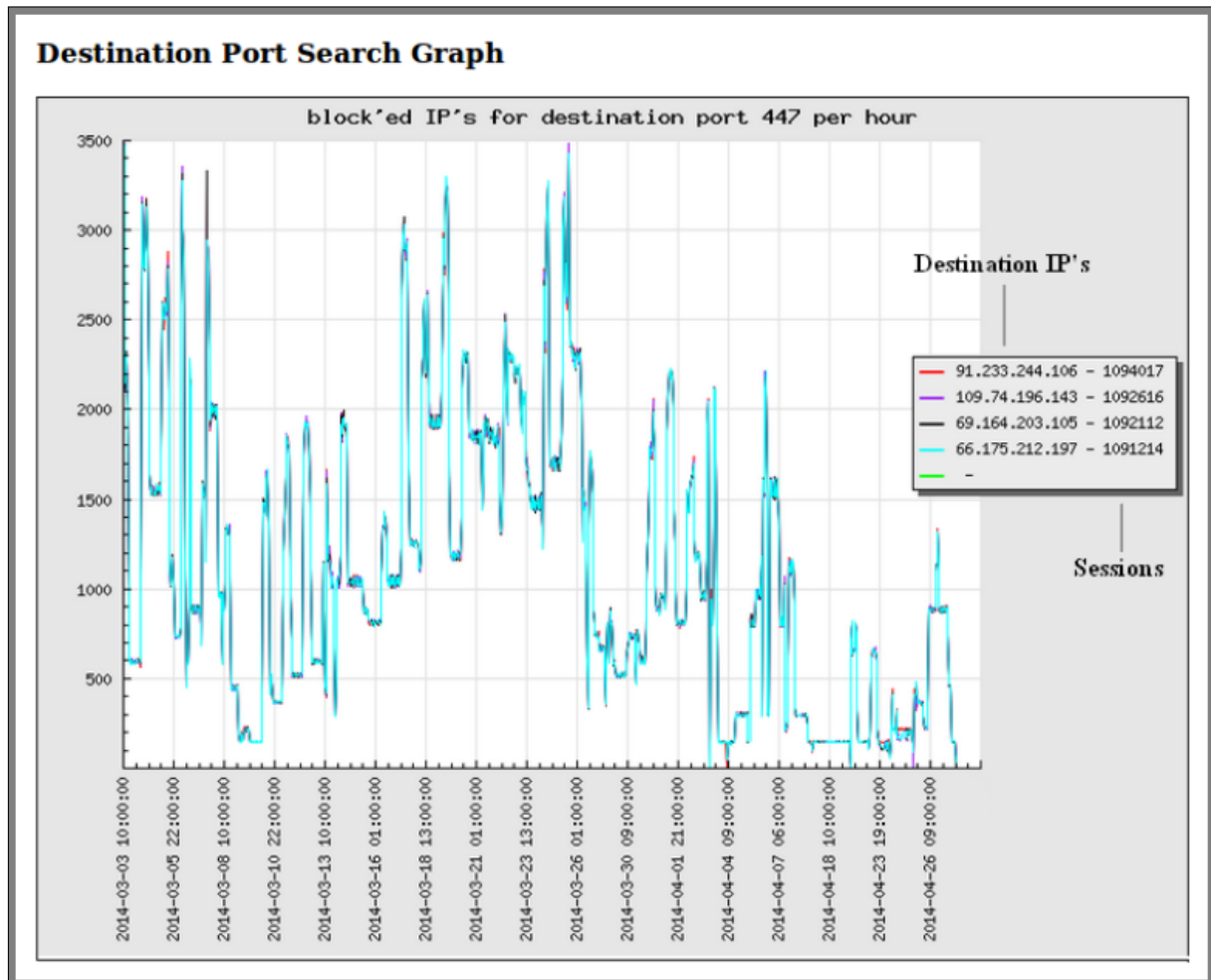


Figure 5.8: Destination port 447/tcp search graph - Destination IPs

Table 5.3: Destination Port 447/tcp Top 10 Destinations

Item	Destination IP:port	Finding
1	91.233.244.106:447	Used by Malware
2	109.74.196.143:447	Used by Malware
3	69.164.203.105:447	Used By Malware.

²⁴<https://www.virustotal.com/>

5.5 Destination Port 44416/tcp

A **Port Search Graph** web application (J in Figure 4.2) was executed, searching for destination port 44416 from 2014-03-01 till 2014-04-31 and graphing all source IPs as per Figure 5.9. These graphs list the top 5 clients that were blocked initiating sessions to destination port 44416 directly to the Internet. Top source IP was identified as a internal IP namely 10.154.106.16 with 192 062 events that were blocked during the date ranges. The second IP in the Top 5 is 10.130.44.91 with 189 432 blocked events. Figure 5.9 shows that there are several IP devices trying to connect to public IPs and must be investigated.

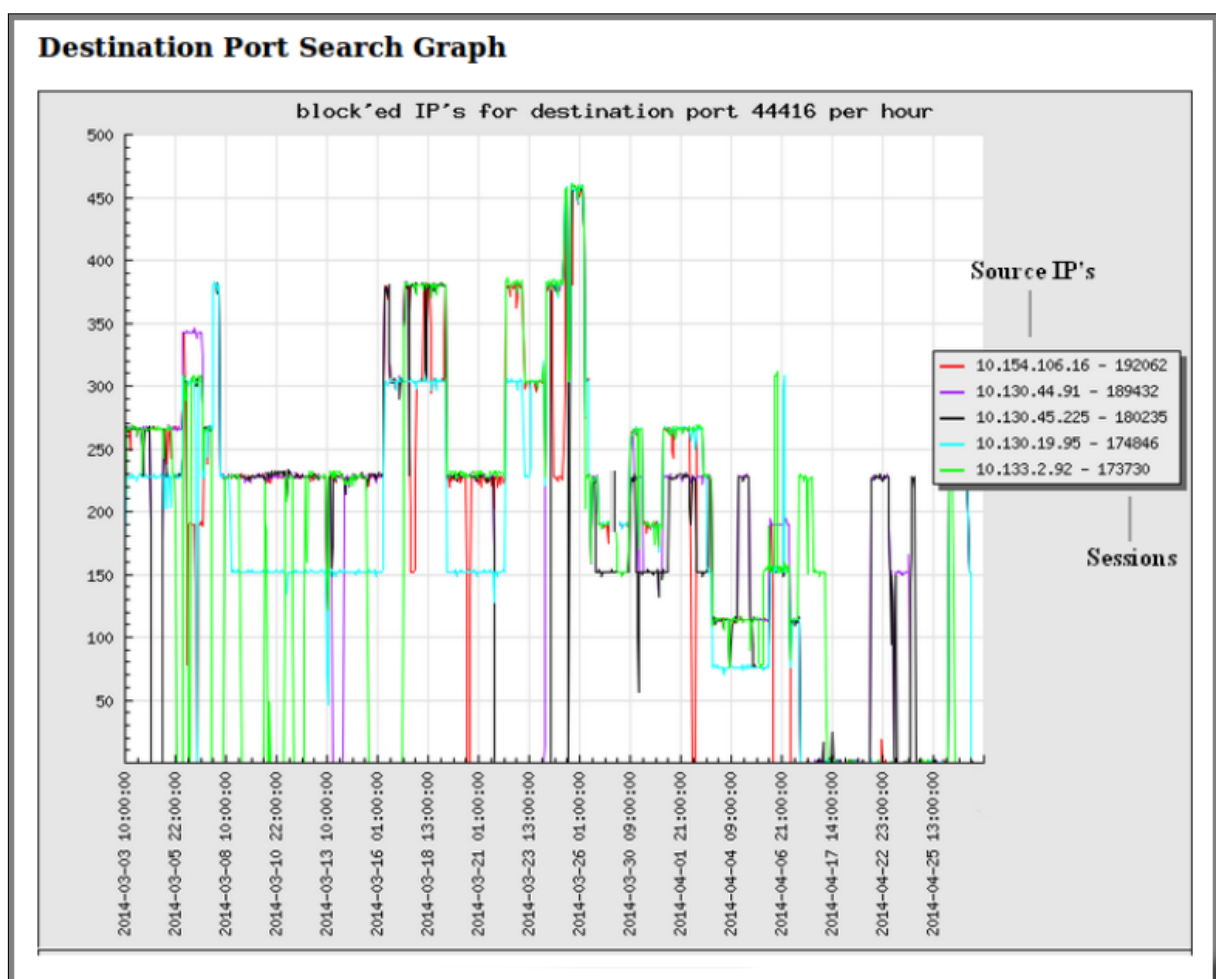


Figure 5.9: Destination port 44416/tcp search graph - Source IPs

A Port Search Graph web application (J in Figure 4.2) was executed, searching for destination port 44416 from 2014-03-01 till 2014-04-31 and graphing all destination IPs as per Figure 5.10. These graphs list the top 5 clients that were blocked initiating sessions via destination port 44416/tcp directly to the Internet. The top destination IP was identified as a public IP

- 141.8.225.62 - with 1 929 519 events that were blocked during the date ranges. Again, all destination IPs were public IPs. Looking at the first three IPs it can be seen that there is a possibility of the source IP being infected with Malware, as per Table 5.4.

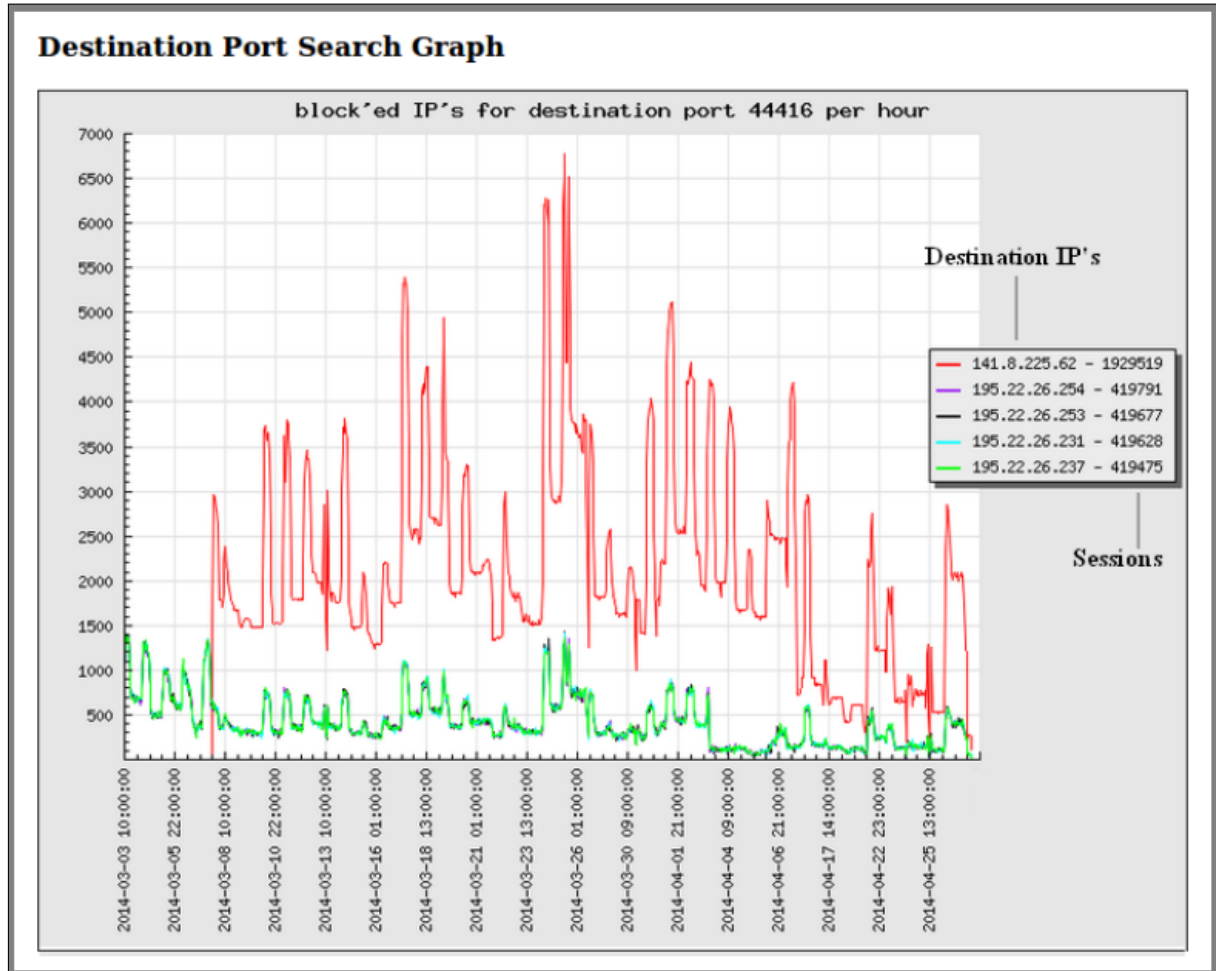


Figure 5.10: Destination port 44416 search graph - Destination IPs

Table 5.4: Destination Port 44416/tcp Top 10 Destinations

Counter	Destination IP	Finding
1	141.8.225.62:44416	IP used by Malware
2	195.22.26.254:44416	IP used by Malware
3	195.22.26.231:44416	IP used by Malware

5.6 RDP connection to IP 10.133.2.71

An RDP session was established from the researcher's computer, with the target identified as 10.133.2.71. The first step after establishing authentication with the target was to verify if Anti-virus was installed, however it was discovered that no running Anti-virus was either actively running or even installed. A netstat command was issued three times, with a 3 minute interval on the command prompt, to display network connections.

The command netstat was executed with a parameter to display the addresses and port numbers in numerical form, as per the results shown in Figure 5.11, numbers 1 and 3. For case number Two netstat was executed with the condition that addresses and ports not be displayed in numerical format, as per the results in Figure 5.11, number 2.

1	TCP	10.133.2.71:1206	69.164.203.105:447	SYN_SENT	
	TCP	10.133.2.71:1207	148.81.111.111:80	SYN_SENT	
	TCP	10.133.2.71:1208	62.141.94.83:21	SYN_SENT	
2	TCP	userat-user:3170	sinkhole.cert.pl:http	SYN_SENT	
	TCP	userat-user:3172	91.233.244.106:447	SYN_SENT	
	TCP	userat-user:1290	mtaout-a-mtc-a.mx.aol.com:smtp	SYN_SENT	
	TCP	userat-user:1291	148.81.111.111:http	SYN_SENT	
	TCP	userat-user:1292	li117-105.members.linode.com:447	SYN_SENT	
	TCP	userat-user:3081	ftp.smtp.ru:ftp	SYN_SENT	
	TCP	userat-user:3083	sinkhole.cert.pl:http	SYN_SENT	
	TCP	userat-user:3084	196.23.168.181:http	SYN_SENT	
3	TCP	userat-user:4571	sinkhole.cert.pl:http	SYN_SENT	
	TCP	userat-user:4573	91.233.244.106:447	SYN_SENT	

Figure 5.11: Combined netstat output for Case Study

In the results of the netstat the states for all the destination public Internet addresses is SYN_SENT, as shown in Figure 5.11. These SYN_SENT states are the result of the architecture's design, which does not allow any internal host to connect to the Internet directly, as discussed in Chapter 4. However, we notice several attempts to connect to the Internet directly, utilising TCP and destination ports 21/tcp, 25/tcp, 80/tcp, and 447/tcp. Due to the firewall configuration and logging these attempts were blocked and logged.

As per Item 2 in Table 5.5, it is clear that malware is behind the compromised state of the command-and-control. It is also interesting to note a connection being made to a sinkhole, operated by cert.pl. This indicates that the domain used to connect has been seized by the cert.

Table 5.5: netstat output

Item	Reverse DNS Address	Remote IP	port
1	li117-105.members.linode.com	69.164.203.105	447/tcp
2	sinkhole.cert.pl	148.81.111.111	80/tcp
3	ftp.smtp.ru	62.141.94.83	21/tcp
4		91.233.244.106	447/tcp
5	mtaout-a-mtc-a.mx.aol.com	64.12.88.133	25/tcp
6		196.23.168.181	80/tcp

An anti virus package with updated definition files was installed on the host, which was then rebooted remotely. A new RDP session was then established from the researcher's computer in order to identify the host, namely 10.133.2.71.. Using the installed Antivirus package, a virus scan was issued on the host: several viruses were found, for example trojan horses and rootkits.

Name	In Folder	Detected As	Detection Type	Status
Memory\NtCreateFile		Generic.dxlrootkit	Trojan	Deleted
Memory\NtQueryInformationProcess		Generic.dxlrootkit	Trojan	Deleted
Memory\ZwCreateFile		Generic.dxlrootkit	Trojan	Deleted
Memory\ZwQueryInformationProcess		Generic.dxlrootkit	Trojan	Deleted
Repost4.htm		W32/Ramnit.alhtm	Virus	Cleaned
SetupVSE.Exe		W32/Sality.gen.z	Virus	Cleaned
Readme_ZH_CN.html		W32/Ramnit.alhtm	Virus	Cleaned
Readme_RU.html		W32/Ramnit.alhtm	Virus	Cleaned
Readme_NL.html		W32/Ramnit.alhtm	Virus	Cleaned
UnInst.exe		W32/Sality.gen.z	Virus	Cleaned
BrotherSoft_ExtremeToolbarHelper.exe		W32/Sality.gen.z	Virus	Cleaned
BrotherSoft_ExtremeToolbarHelper1.exe		W32/Sality.gen.z	Virus	Cleaned
WindowsInstaller-KB893803-v2-x86.exe		W32/Sality.gen.z	Virus	Cleaned
Stinger_22062014_140754.html		W32/Ramnit.alhtm	Virus	Deleted
Stinger_22062014_143445.html		W32/Ramnit.alhtm	Virus	Deleted
FramePkg.exe		W32/Sality.gen.z	Virus	Cleaned
BRKBHOOK.DLL		W32/Ramnit.a	Virus	Cleaned
brif03a.dll		W32/Ramnit.a	Virus	Cleaned
BRNIPMON.exe		W32/Sality.gen.z	Virus	Cleaned
BROWNIE.EXE		W32/Ramnit.a	Virus	Cleaned
BROWNIE.EXE		W32/Sality.gen.z	Virus	Cleaned
BrstsWdmgr.exe		W32/Ramnit.k	Virus	Deleted
brpip04amgr.exe		W32/Ramnit.k	Virus	Deleted
RCHANGE.EXE		W32/Ramnit.a	Virus	Cleaned
Update Hosts and Services Files Version 2.exe		W32/Ramnit.a	Virus	Cleaned
Capture Station Update.exe		W32/Ramnit.a	Virus	Cleaned
Copy of Shortcut to (1).lnk		Exploit-CVE-2010-2568	Trojan	Deleted
hh.exe		W32/Virut.n.gen	Virus	Cleaned
regedit.exe		W32/Virut.n.gen	Virus	Cleaned
regsvr.exe		W32/YahLover.worm.gen	Virus	Deleted
Capture Station Update.exe		W32/Sality.gen.z	Virus	Cleaned
RCHANGE.EXE		W32/Sality.gen.z	Virus	Cleaned
Update Hosts and Services Files Version 2.exe		W32/Virut.n.gen	Virus	Cleaned
BROWNIE.EXE		W32/Virut.n.gen	Virus	Cleaned
winhlp32.exe		W32/Virut.n.gen	Virus	Cleaned
InetFTPServermgr.exe		W32/Ramnit.k	Virus	Deleted
zjk.lnk		Exploit-CVE-2010-2568	Trojan	Deleted
zgF.lnk		Exploit-CVE-2010-2568	Trojan	Deleted
zKn.lnk		Exploit-CVE-2010-2568	Trojan	Deleted
zlj.lnk		Exploit-CVE-2010-2568	Trojan	Deleted
x.exe		W32/Virut.n.gen	Virus	Cleaned
teogijy.scr		W32/Virut.n.gen	Virus	Cleaned
teogijyx.exe		W32/Virut.n.gen	Virus	Cleaned
zmO.lnk		Exploit-CVE-2010-2568	Trojan	Deleted
zxo.lnk		Exploit-CVE-2010-2568	Trojan	Deleted
zyl.lnk		Exploit-CVE-2010-2568	Trojan	Deleted
zvG.lnk		Exploit-CVE-2010-2568	Trojan	Deleted
zyl.lnk		Exploit-CVE-2010-2568	Trojan	Deleted
x.exe		Downloader-CJX.gen.g	Trojan	Deleted
teogijy.scr		Downloader-CJX.gen.g	Trojan	Deleted
teogijyx.exe		Downloader-CJX.gen.g	Trojan	Deleted
Patch4.htm		W32/Ramnit.alhtm	Virus	Cleaned

Figure 5.12: Virus report for Case Study

Table 5.6: Observed Ports

Rank	Port Number	Description
1	33333/tcp	Well known virus port
2	21/tcp	FTP
3	25/tcp	SMTP
4	447/tcp	DDM-Distributed File Management
5	22/tcp	SSH
6	53/udp	DNS
7	23/tcp	SSH
8	3123/tcp	Mainframe 3270 emulator
9	8086/tcp	Mainframe Web service
10	8087/tcp	Mainframe Web service
11	3389/tcp	RDP

5.7 Summary

It should be clear that from a systematic deep packet analysis of the PF logs and the visualizations thereof, a security vulnerability in the organisation was eliminated. The significance of this case study is not simply that a problem was solved; the methodology used, as well as the fact that such a problem would never have been identified or solved via normal network or security management techniques, is even more compelling. The ability to consistently prevent this type of vulnerability will undoubtedly prevent events such as data leakage in the long term.

6

Conclusion

THIS document has presented a proof of concept visualisation system that can correlate, analyse and present logs visually in tables, graphs and maps. The software used during this research, for example Freebsd, was selected primarily due to availability. This thesis presents useful information regarding the FreeBSD firewall; system administrators that have a need to understand how FreeBSD logs can be used to their advantage will find this research compelling and useful in their daily work. The reader is also presented with a network architecture in order to demonstrate the ways in which network architecture can be used to force technology to respond in a required manner. By forcing SMTP to only be allowed to communicate with the Internet via the internal SMTP servers and through the DMZ firewall, for example, it is possible to identify internal systems that initiate SMTP sessions directly through the organization's DMZ firewall.

Chapter 2 provided a Literature Review, covering the concepts that were used during this thesis. An overview is also given as to why logs are essential in this day and age.

Chapter 3 describes how the source data was collected and imported into a database and also indicates the structure of the logs and database.

Chapter 4 deals with how the proof of concept graphing system was designed and how the system works. This chapter also covers the network architecture that was implemented during the data gathering period. The configuration of the firewall and network is also discussed.

Chapter 5 constitutes the major part of this document. One case study is presented and explained, showing how firewall logs can be used to identify security events and how these logs can be used to assist a firewall administrator to build a firewall rules template.

6.1 Research evaluation

This section assesses the degree to which the stated research goals were met. The aim of this research was to build a proof of concept system that could store, analyse, and visually present graphs as well as statistics of FreeBSD PF firewall logs by using an easy to use, open source solution that can be used within environments where FreeBSD and PF are installed. The researcher is thus able to present the reader with the outcome of the following goals, as established in Chapter 1:

- Store and forward the PF logs using a technology such as syslog. In Chapter 3 the researcher described how FreeBSD was configured to rotate, store pflogs and transfer the logs to a syslog server. Code Listing 3.4 presents the configuration file for rotating and storing the pflog file on a daily basis. In Code Listing 3.2 the reader is presented with the script used for transferring the pflogs to a syslog server utilising the logger program.
- Store syslog events in a MySQL database. The reader is presented with the scripts used for analysing the syslog files and importing the events into the MySQL database, as per Code Listing 3.6, store syslog events in a MySQL database. The syslog file is written into memory from the script, as per Code Listing 6.2, and 6.3 writes the contents to a comma delimited file, finally using the batch MySQL import functionality to import the comma delimited file into the database.
- Correlate log entries (events per hour) . The researcher wrote a MySQL query to correlate the logs; this correlation query grouped all identical events per hour together using the date, time, source IP, destination IP, destination port and action. It also adds a column called session, which states how many times the events were written per that date time and hour. The SQL query for the correlation is presented in Code Listing 3.8.

- Present summarised blocked logs in the format of graphs, tables and maps. The reader is presented with the proof of concept Visualisation System in Chapter 4. The System menus are made available in Figure 4.2 . Within this chapter the researcher presents to the reader each of the menu items with an example and explanation of each. The data is presented to the reader as tables, text, graphs and google maps. Only one high level image is displayed within this thesis due to sensitivity, privacy and confidentiality of the organisation, taking into consideration that live data was used for this thesis. Google maps (geomap) functionality, combined with live internet data is presented in Figure 4.2 and explained on a high level in Section 4.3.3
- Identify blocked connections into or out of the organization's network. The Visualisation System is able to present passed or blocked events as per menu items under Section 4.3. The data is presented in tables, text, graphs and google maps.
- Identify alerts by analysing the PF logs with the proof of concept Visualisation System. It was possible to identify alerts by analysing the PF logs with the proof of concept Visualisation System as per the findings in Figure 5.12. Within the case study in Chapter 5, the Visualisation System was utilised to identify a security vulnerability, or malware. This kind of problem would never have been identified or solved via normal network or security management techniques, proving that a system such as the proof of concept Visualisation System will add positively to any organisation.

Within the research that was conducted and presented in this thesis it has been shown that PF firewall logs can be visually displayed and can add positively to any organisation that is using PF as their firewall.

6.2 Expectations during the start of the research

During the start of the research it was expected that most denied events would be initiated from the Internet to the organization subnets. After the first months, PF syslog data was imported into the database, summarization completed and analysis run it became apparent that most denied events was initiated from the internal network of the organization. This shows that most threats to organizations are not in fact external but internal.

6.3 Lessons learned

Organizations should decide what data they see as relevant for logging and analysis, then plan and implement a solution that can collect, summarise, and display findings; this solution must also alert the relevant people depending on the classification of the event raised. Technology is available and can be used to 'see' what is running on organization's network, enabling organizations to have early warning systems and properly manage data that is flowing on the network.

6.4 Future Work

Within the final section of this thesis the reader is presented with possible future research that can be done with regards to log analysis, extending the present work:

- Future research in correlating large volumes of PF log files in a decentralised approach, in which the correlated logs are sent to a central database, over limited amounts of bandwidth, for analysis, as well as presenting the events, reports, and alerts in a web based solution.
- Collecting, analysing and presenting findings within network traffic packet capture (pcap) files. The data within a pcap file can be used to replay an unauthorised attack against, for example, a web server in order to identify how an attacker breached or tried to breach security mechanisms. Pcap files can also be used for forensic evidence, having the capability to replay transactions or to be used as evidence in a court of law. Due to the size of pcap files the storing, analysing and presenting of usable data from these files - in corporate environments as well as in small businesses - is still a challenge.
- Research the use of netflow data within the context of Information Security. Netflow is a well known technology used by many, both open source and commercial, hardware and software. The collection of netflow data from several devices, baseline and analysis of the data can present valuable information with regards to how data flows on a network.

6.5 Conclusion

This thesis provides evidence, through case studies, that by analysing logs threats can be identified and dealt with swiftly. For an individual to analyse millions of lines of logs per day is impractical and many threats will slip by due to the fact that a human can not correlate, analyse and make informed decisions as quickly as a well-designed analysis system. By not only storing logs to comply with auditors or legislation, logs can add positively to any organization, be it a corporate organization or a small business.

References

- Taylor Armerding. The 15 worst data security breaches of the 21st Century, 2012. URL <http://www.csoononline.com/article/700263/the-15-worst-data-security-breaches-of-the-21st-century>. (Date last accessed: 30-December-2014).
- Shabtai Asaf, Elovici Yuval, and Rokach Lior. *A Survey of Data Leakage Detection and Prevention Solutions*. Springer, 2012. ISBN 978-1-4614-2052-1.
- Daniel Barrett, J. *SSH, The Secure Shell: The Definitive Guide*. O'Reilly Media, 2005. ISBN 978-0-596-00011-0.
- Bruce Brown, C. *How to Stop E-mail Spam, Spyware, Malware, Computer Viruses, and Hackers*. Atlantic Publishing Group, 2011. ISBN 978-1601383037.
- Jonathan S Burgess. *The Tradition of the Trojan War in Homer and the Epic Cycle*. JHU Press, 2001. ISBN 978-0801878909.
- Tracy Burrows. SA fails on forensic readiness, April 2013. URL http://www.itweb.co.za/index.php?option=com_content&view=article&id=62939. (Date last accessed: 30-December-2014).
- Taylor Carol, Barbara Endicott-Popovsky, and Deborah A Frincke. Specifying digital forensics: A forensics policy approach. *Digital Investigation*, 4:101–104, 2007.
- Eric Chabrow. Struggling to Make Sense of Log Data, May 2012. URL <http://www.govinfosecurity.com/struggling-to-make-sense-log-data-a-4739>. (Date last accessed: 30-December-2014).
- Anton Chuvakin. Five Best and Five Worst Practices for SIEM by Dr. Anton Chuvakin, July 2011. URL http://www.slideshare.net/anton_chuvakin/

- five-best-and-five-worst-practices-for-siem-by-dr-anton-chuvakin. (Date last accessed: 30-December-2014).
- Bradley Cowie. An Exploratory Study of Techniques in Passive Network Telescope Data Analysis. Master's thesis, Rhodes University, November 2012.
- Cyfor. How can you justify a forensic readiness plan?, February 2012. URL <http://cy4or.co.uk/news/digital-forensics/how-can-you-justify-a-forensic-readiness-plan/>. (Date last accessed: 30-December-2014).
- Sarang Dharmapurikar, Praveen Krishnamurthy, Todd Sproull, and John Lockwood. Deep packet inspection using parallel bloom filters. In *High Performance Interconnects, 2003. Proceedings. 11th Symposium on*, pages 44–51. IEEE, 2003.
- Nick Gifford. *Information Security: Managing the Legal Risks*. CCH Australia Limited, 2009. ISBN 1921593296.
- Deb Goodkin. FreeBSD Foundation 2014 Year-End Fundraising Appeal, December 2014. URL <http://freebsdoundation.blogspot.com/2014/12/freebsd-foundation-2014-year-end.html>. (Date last accessed: 30-December-2014).
- Anna Granova and JHP Eloff. Online banking and identity theft: who carries the risk? *Computer Fraud & Security*, (November):7–11, 2004.
- CP Grobler and CP Louwrens. Digital forensic readiness as a component of information security best practice. In *New Approaches for Security, Privacy and Trust in Complex Environments*, pages 13–24. Springer, 2007.
- Peter N.M Hansteen. *The Book of PF*. No Starch Press, 3rd edition edition, 2014. ISBN 9781593275891.
- IBM. Proventia Network Intrusion Prevention System (IPS), April 2011. URL <http://www-935.ibm.com/services/th/en/it-services/proventia-network-intrusion-prevention-system-ips.html>. (Date last accessed: 30-December-2014).
- Barry Vivian William Irwin. *A Framework For the application of network telescope sensors in a global IP network*. PhD thesis, Rhodes University, January 2011.
- Rita Judith. *Computer Virus - Unabridged Guide*. Tebbo; Unabridged edition, 2012. ISBN 978-1486198986.

- Henrik Karlzén. An Analysis of Security Information and Event Management Systems-The Use of SIEMs for Log Collection, Management and Analysis. 2009. (Date last accessed: 30-December-2014).
- Michael Kende. Internet global growth: lessons for the future, September 2012. URL <http://www.analysysmason.com/Research/Content/Reports/Internet-global-growth-lessons-for-the-future/Internet-global-growth-lessons-for-the-future/>. (Date last accessed: 30-December-2014).
- Karen Kent, Suzanne Chevalier, Tim Grance, and Hung Dang. Guide to integrating forensic techniques into incident response. *NIST Special Publication*, -:800–86, 2006.
- Jay Kreps. The Log: What every software engineer should know about real-time data’s unifying abstraction, December 2013. URL <http://engineering.linkedin.com/distributed-systems/log-what-every-software-engineer-should-know-about-real-time-datas-unifying>. (Date last accessed: 30-December-2014).
- Michael Larabel. Why Should You Use FreeBSD? Here’s Some Reasons, May 2012. URL http://www.phoronix.com/scan.php?page=news_item&px=MTExMDg. (Date last accessed: 30-December-2014).
- Michael Larabel. Oracle Releases MySQL 5.6 To Improve NoSQL, Performance, February 2013. URL http://www.phoronix.com/scan.php?page=news_item&px=MTI5MzI. (Date last accessed: 30-December-2014).
- Dru Lavigne. *The best of FreeBSD Basics*. Reed Media Services, 2007. ISBN 978-0979034220.
- Greg Lehey. *The Complete FreeBSD*. O’Reilly Community Press, 4th edition edition, 2003. ISBN 978-0596005160.
- James A. Lewis. Raising the Bar for Cybersecurity, February 2013. URL http://csis.org/files/publication/130212_Lewis_RaisingBarCybersecurity.pdf. (Date last accessed: 30-December-2014).
- Michael W. Lucas. *Absolute FreeBSD*, volume 2nd Edition. No Starch Press, 2007. ISBN 978-1-59327-151-0.
- Sandra Mariana Maat. Cyber crime: a comparative law analysis. Master’s thesis, Unisa, 2009. URL <http://uir.unisa.ac.za/handle/10500/2056>. (Date last accessed: 30-December-2014).

- Trisha Marczak. Anonymous Exposure Of Taboo Rape Issue Challenges Public Perception Of Hacktivists, January 2013. URL <http://www.mintpress.net/anonymous-exposure-of-taboo-rape-issues-challenges-public-perception-of-hacktivists> (Date last accessed: 30-December-2014).
- Elinor Mills. FTC sues Wyndham Hotels over data breaches, June 2012. URL http://news.cnet.com/8301-1009_3-57460551-83/ftc-sues-wyndham-hotels-over-data-breaches/. (Date last accessed: 30-December-2014).
- David Moore, Colleen Shannon, Geoffrey M Voelker, and Stefan Savage. Network telescopes: Technical report. Technical report, 2004. URL <http://ants.iis.sinica.edu.tw/3bkmj9ltewxtsrrvnoknfdxrm3zfwrr/17/tr-2004-04.pdf>. (Date last accessed: 30-December-2014).
- NDTV. Hackers target US security think tank, December 2011. URL <http://www.ndtv.com/article/world/hackers-target-us-security-think-tank-160686>. (Date last accessed: 30-December-2014).
- NetworkComputing. How the Top Eight SIEM Vendors Stack Up, July 2012. URL <http://www.networkcomputing.com/careers-and-certifications/how-the-top-eight-siem-vendors-stack-up/d/d-id/1233787?> (Date last accessed: 30-December-2014).
- Mark Nicolett and Kelly M Kavanagh. Magic quadrant for security information and event management. *Gartner RAS Core Research Note*, 2011. URL http://www.novell.com/docrep/2011/05/gartner_magic_quadrant_siem_report_may2011.pdf. (Date last accessed: 30-December-2014).
- Parmy Olson. *We are anonymous: Inside the hacker world of LulzSec, Anonymous, and the global cyber insurgency*. Back Bay Books, 2012. ISBN 978-0316213523.
- OpenSUSE. What is the difference between openSUSE and SUSE's enterprise offerings?, 2014. URL http://en.opensuse.org/openSUSE:FAQ#What_is_the_difference_between_openSUSE_and_SUSE.27s_enterprise_offerings.3F. (Date last accessed: 30-December-2014).
- Oxford-Dictionaries. Definition of cyber in English, 2013. URL <http://oxforddictionaries.com/definition/english/cyber?q=cyber>. (Date last accessed: 30-December-2014).

- Oxford-Dictionaries. Definition of crime in English, 2014. URL <http://www.oxforddictionaries.com/definition/english/crime>. (Date last accessed: 30-December-2014).
- Francois Paget. Hacktivism Cyberspace has become the new medium for political voices, 2012. URL <http://www.mcafee.com/hk/resources/white-papers/wp-hacktivism.pdf>. (Date last accessed: 30-December-2014).
- Lei Pan and Lynn M Batten. Reproducibility of digital evidence in forensic investigations. In *Digital Forensic Research Workshop*, 2005.
- Chad Perrin. Filtering PF firewall logs, February 2011. URL <http://www.techrepublic.com/blog/it-security/filtering-pf-firewall-logs/>. (Date last accessed: 30-December-2014).
- Pingdom. Amazing facts and figures about the evolution of hard disk drives, February 2010. URL <http://royal.pingdom.com/2010/02/18/amazing-facts-and-figures-about-the-evolution-of-hard-disk-drives/>. (Date last accessed: 30-December-2014).
- J. Postel. Transmission Control Protocol. RFC 793 (INTERNET STANDARD), September 1981. URL <http://www.ietf.org/rfc/rfc793.txt>. Updated by RFCs 1122, 3168, 6093, 6528.
- Margaret Rouse. Rootkit, January 2008. URL <http://searchmidmarketsecurity.techtarget.com/definition/rootkit>. (Date last accessed: 30-December-2014).
- Robert Rowlingson. A ten step process for forensic readiness. *International Journal of Digital Evidence*, 2(3):1–28, 2004.
- Gorka Sadowski. Using logs for forensics after a data breach, November 2010. URL <http://www.networkworld.com/article/2193990/tech-primers/using-logs-for-forensics-after-a-data-breach.html>. (Date last accessed: 30-December-2014).
- Shah Sahil. Rising Concerns on Cyber Crime & Scams in India. Technical report, Tech in Asia, January 2011. URL <http://www.techinasia.com/online-scams-in-india/>. (Date last accessed: 30-December-2014).
- Tim Sammut. Understanding SQL Injection. Technical report, CISCO, 2012. URL http://www.cisco.com/web/about/security/intelligence/sql_injection.html. (Date last accessed: 30-December-2014).

- SAPS. Common Law Offences. Technical report, South African Police Service: Department of Police, 2013. URL http://www.saps.gov.za/crimestop/common_offence_definitions.htm. (Date last accessed: 30-December-2014).
- William Stallings. *SNMP, SNMPv2, SNMPv3, and RMON 1 and 2*. Addison-Wesley Longman Publishing Co., Inc., 1998. ISBN 9788131702307.
- T Steyn, HA Kruger, and L Drevin. Identity theft Empirical evidence from a Phishing exercise. In *New Approaches for Security, Privacy and Trust in Complex Environments*, pages 193–203. Springer, 2007.
- John Tan. Forensic Readiness, 2001. URL http://isis.poly.edu/kulesh/forensics/forensic_readiness.pdf. (Date last accessed: 30-December-2014).
- TechExpert. Most notorious viruses in PC history, September 2007. URL <http://immenseknowledge.blogspot.com/2007/09/most-notorious-viruses-in-pc-history.html>. (Date last accessed: 30-December-2014).
- Rahul Thadani. The first computer virus was designed for an Apple computer, by a 15 year old, February 2013. URL <http://blogs.quickheal.com/wp/the-first-pc-virus-was-designed-for-an-apple-computer-by-a-15-year-old/>. (Date last accessed: 30-December-2014).
- Aleksandar Valjarevic and Hein A. Venter. Harmonised Digital Forensic Investigation Process Model. 2012.
- David Wall. *Cybercrime: The transformation of crime in the information age*, volume 4. Polity, 2007. ISBN 978-0745627366.
- Ashford Warwick. 2011 the year of the hacktivist, Verizon data breach report reveals, March 2012. URL <http://www.computerweekly.com/news/2240147270/2011-the-year-of-the-hacktivist-Verizon-data-breach-report-reveals>. (Date last accessed: 30-December-2014).
- Samuel Alexandra Whitney. *Hacktivism and the future of political participation*. PhD thesis, Harvard University Cambridge, Massachusetts, 2004. (Date last accessed: 30-December-2014).
- Edward Wilding. *Information risk and security: preventing and investigating workplace computer crime*. Gower Publishing Co., 2006. ISBN 0566086859.

- Victoria Woollaston. Hacking Wi-Fi is childs play! 7-year-old shows how easy it is to break into a public network in less than 11 MINUTES. Online News Paper, January 2015. URL <http://www.dailymail.co.uk/sciencetech/article-2919762/Hacking-Wi-Fi-s-child-s-play-Seven-year-old-shows-easy-break-public-network-11-minutes.html>. (Date last accessed: 30-December-2014).
- Majid Yar. The Novelty of CybercrimeAn Assessment in Light of Routine Activity Theory. *European Journal of Criminology*, 2(4):407–427, 2005.
- Fang Yu, Zhifeng Chen, Yanlei Diao, TV Lakshman, and Randy H Katz. Fast and memory-efficient regular expression matching for deep packet inspection. In *Architecture for Networking and Communications systems, 2006. ANCS 2006. ACM/IEEE Symposium on*, pages 93–102. IEEE, 2006.

Appendix

THE reader is presented with the scripts used for analyzing the syslog files, for importing events into the MySQL database (see Code Listing 6.1), and for storing syslog events in the MySQL database. The syslog file is written into memory from the script (see Code Listing 6.2 and Code Listing 6.3) and writes the contents to a comma-delimited file, finally using the batch MySQL import functionality to import the comma-delimited file into the database.

Code Listing 6.1: Single log file import into DB

```
1 # Single log file import into DB
2 #!/bin/sh
3 cat /home/dirkc/logs/2014/07/GP-DHA-DMZ-fw.log.0 | /usr/local/www/apache22/data/import_logs/↵
    pfanalyse4.pl -y 2014 -d DMZ -v
```

Code Listing 6.2: pfanalyse4.pl

```
1 # pfanalyse4.pl
2 #!/usr/bin/perl
3 # Usage Notes
4 # cat /GP-DMZ-fw.log.0 | /usr/local/www/apache22/data/import_logs/pfanalyse4.pl -y 2014 -d ↵
    DMZ -v
5
6 require "syslib.pm";
7
8 $year = "2014";
9 $device = "DMZ";
10
11 use Getopt::Long;
12 $outfile = "\tmp\OUTFILE.CSV";
13 $year = "2014";
14 $verbose;
15 GetOptions ("outfile=s" => \$outfile,    # numeric
16             "year=s"    => \$yr,        # string
17             "device=s"  => \$device,    # string
```

```

18         "verbose" => \$verbose) # flag
19 or die("Error in command line arguments\n");
20
21 %mnt = ("Jan" => "01", "Feb" => "02", "Mar" => "03", "Apr" => "04", "May" => "05", "Jun" => "06", "Jul" => "07", "Aug" => "08", "Sep" => "09", "Oct" => "10", "Nov" => "11", "Dec" => "12");
22 # Process each line as <stdin> to prevent having to accomodate entire file in RAM
23 while(<>){
24     my($year,$month,$day,$time,$action,$direction,$srcip,$srcport,$dstip,$dstport,$proto) = process_line($_);
25     if($verbose){
26         $counter++;
27         if($counter % 100000 == 0){
28             print "$counter records processed -> output length ",length($sqlfile),"\n";
29         }
30     }
31     $sqlfile .= "\"$year-$month{$month}-$day $time\", \"$device\", \"$action\", \"$direction\", \"$srcip\", \"$dstip\", $srcport, $dstport, \"$proto\", \"\"\"\\n\"";
32 }
33
34 open(OUTFILE, ">$outfile");
35 print OUTFILE $sqlfile;
36 close(OUTFILE);
37
38 if ($verbose){
39     print "Starting connection to database\n";
40 }
41
42 use DBI;
43 $dbh = DBI->connect('dbi:mysql:traffic','root','toor') or die "Connection Error: $DBI::errstr\n";
44 # $dbh = DBI->connect('dbi:mysql:traffic;host=127.0.0.1','root','toor') or die "Connection Error: $DBI::errstr\n";
45 my $start = time;
46
47 if($verbose){
48     print "Disabling Table Indexes\n";
49 }
50 $sql = "ALTER TABLE traffic_copy DISABLE KEYS;\n";
51 $sth = $dbh->prepare($sql);
52 $sth->execute or die "SQL Error: $DBI::errstr\n";
53 $sql = "LOAD DATA INFILE '$outfile.' INTO TABLE traffic_copy FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '\"' LINES TERMINATED BY '\\n' (packetdate,device,action,direction,@src_ip,@dst_ip,src_port,dst_port,protocol,protocol-info) SET src_ip = inet_aton(@src_ip), dst_ip = inet_aton(@dst_ip);\n";
54 $sth = $dbh->prepare($sql);
55 $sth->execute or die "SQL Error: $DBI::errstr\n";
56 if($verbose){
57     print "Re-Enabling Table Indexes\n";
58 }
59 $sql = "ALTER TABLE traffic_copy ENABLE KEYS;\n";
60 $sth = $dbh->prepare($sql);
61 $sth->execute or die "SQL Error: $DBI::errstr\n";
62 my $duration = time - $start;
63 if($verbose){
64     print "disconnecting from database : Processing time in seconds - $duration\n";

```

```
print "Device name is = $device \n";
}
```

Code Listing 6.3: syslib.pm

```

1 # syslib.pm
2 #!/usr/bin/perl
3
4 sub process_line($line){
5     $line = $_;
6     my($prot) = "";
7     chomp($line);      # Remove \n \r Newline characters off the end of each line
8     if ($line =~ /^(...)\s+(\d+)\s+(\d+:\d+:\d+)\s+(\d+\.\d+\.\d+\.\d+)\s+(\d+:\d+:\d+\.\d+)\s+(\d+↵
9         rule\s+(\d+)/\d+(\match):\s+(.*)\s+(.*)\s+on\s+.*?:\s+(.*)\s+.*?\s+(.*?):\s+(.*)\s+(.*)$<-
10         /){
11         ($month,$day,$time,$action,$direction,$sip,$dip,$rest) = ($1,$2,$3,$4,$5,$6,$7,$8); # ↵
12             Split the line into its component parts
13         # Deal with lines which have ports (TCP + UDP) and lines which don't (ICMP)
14         if($sip =~ /^(\d+\.\d+\.\d+\.\d+)\.(\d+)$/){
15             ($srcip,$srcport) = ($1,$2);
16         }
17         elsif($sip =~ /^(\d+\.\d+\.\d+\.\d+)$/){
18             ($srcip,$srcport) = ($1,0);
19         }
20         if($dip =~ /^(\d+\.\d+\.\d+\.\d+)\.(\d+)$/){
21             ($dstip,$dstport) = ($1,$2);
22         }
23         elsif($dip =~ /^(\d+\.\d+\.\d+\.\d+)$/){
24             ($dstip,$dstport) = ($1,0);
25         }
26         $rest = uc($rest);
27         if($rest =~ /^(.*)\s+/){
28             $prot = $1;
29         }
30         else {          # Set the protocol to upper-case
31             $prot = uc($rest);
32         }
33     ##### NORMALIZE DATA: Fix anomalies
34     $prot =~ s/\./g;
35     $prot =~ s/\s/g;
36     $prot =~ s/[\\[\]\|]/g;
37     if($prot =~ /^L2TP/){$prot = "L2TP"}
38     if($prot =~ /^L2F/){$prot = "L2F"}
39     if($prot =~ /^NTP/){$prot = "UDP"}
40     if($prot =~ /^NBT/){$prot = "UDP"}
41     if($prot =~ /^SIP/){$prot = "UDP"}
42     if($prot =~ /^SMB/){$prot = "UDP"}
43     if($prot =~ /^SNMP/){$prot = "UDP"}
44     if($prot =~ /^RX/){$prot = "UDP"}
45     if($prot =~ /^RADIUS/){$prot = "UDP"}
46     if($prot =~ /^DOMAIN/){$prot = "UDP"}
47     if($prot =~ /^SYSLOG/){$prot = "UDP"}
48     ##### NORMALIZE DATA Upper-case
49 }

```

```
47
48     elseif(( $\$line$  =~ /CRON/) || ( $\$line$  =~ /sm-mt/) || ( $\$line$  =~ /sendmail/) || ( $\$line$  =~ /↵
        kernel/) || ( $\$line$  =~ /#011#011/) || ( $\$line$  =~ /iss-monitor/)){
49     }
50     else {
51         print "INVESTIGATE\n";
52         print "***** $\$line$ \n";
53     }
54     if( $\$prot$  =~ /^$/){ $\$prot$  = "UNKNOWN"}
55     return ( $\$year$ , $\$month$ , $\$day$ , $\$time$ , $\$action$ , $\$direction$ , $\$srcip$ , $\$srcport$ , $\$dstip$ , $\$dstport$ , $\$prot$ )↵
        ;
56 }
57
58 1;
```