Exploring the Applicability of SIEM Technology in IT Security

Chikonga Maimbo

A thesis submitted to the
Auckland University of Technology
in partial fulfilment of the requirements for the degree of
Master of Computer and Information Sciences (MCIS)

April 2014

Primary Supervisor: Krassie Petrova

Table to Contents

List of Tables

List of Figures

Abbreviations

CIA	Confidentiality, Integrity and Availability
IS	Information Systems
IT	Information Technology
PCI DSS	Payment Card Industry Data Security Standard
SIEM	Security Information and Event Management
WMI	Windows Management Instrumentation

Declaration

“I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.”

C. MAIMBO

Acknowledgements

First and foremost I would like to thank my supervisor Krassie Petrova for her timeous advice, guidance and support stretching way back to my time studying for a Postgraduate Diploma in Computer and Information Sciences with AUT. Krassie's supervisory experience was important in refining the thesis topic as well as working closely with me to completion of this research.

Special thanks to Colin Eagle, my manager at the Reserve Bank of New Zealand for providing much needed support and approving leave allowing me to make crucial progress at critical stages of writing this thesis.

Finally and most importantly I would to thank and acknowledge my awesome wife Bridget Maimbo for being supportive and patient throughout the course of my study, especially looking after our lovely daughter at the times I needed time to focus on my study. Special thanks to our daughter Alora Makatendeka Maimbo for giving daddy time to work on the thesis.

Abstract

The changing Information Security (IS) landscape and increased legal, regulatory and audit compliance requirements have driven organisations to collect, maintain, securely store and regularly analyse application, system and network event logs. For some organisations, collecting and storing event logs is primarily to satisfy national or industry specific compliance requirements and secondarily for event based information security monitoring. For other organisations, the collection, storage and analysis of event logs is to add an additional dimension to the increasingly multi-dimensional approach to information security. Monitoring the security of networks through the collection, filtering, aggregation, normalisation, correlation and analysis of archived or real-time event logs has indeed increasingly become one of the core activities in the day to day operations of information security professionals.

While event logs have in the past been primarily used for monitoring the health and operational status of networks, from purely a system and network administration perspective, they are today considered a critical source of deriving important information on the overall security status of applications, systems, networks, data and information. From a security perspective these event logs are mined or analysed to identify potential or real security threats, breaches, anomalies or any other suspicious behaviour. Beyond their storage, the presentation of event messages has also become important in order to enable timely response to any anomalies within the network, improve decision making and assist in the restoration of a network to normal operation after security breach.

Security Information and Event Management (SIEM) technology have in the recent decade seen an increase in adoption by organisations responding to a rapidly evolving information security landscape spurred by the complexity of information security threats currently being observed. Threats that today are in some instances appropriately name Advanced Persistent Threats (APTs). The need to equip information security professionals with a level of ease in proactively identifying security threats or reactively in forensic analysis post a security breach also stand as some of the enablers for SIEM adoption. Regulatory compliance has perhaps been the most significant driver for the adoption of SIEM by organisations.

SIEM capability includes collection, filtering, aggregation, normalisation and correlation of events messages, collected from a wide range of systems, network devices and applications. These capabilities represent a significant enhancement from the basic event log collection and storage of Log Management technologies from which SIEM was an outgrowth. Log management technologies have traditionally being lacking in similar capabilities to SIEM, and as mentioned mostly capable of basic collection and storage of event logs.

This research investigated the applicability of SIEM technology within the context of IT Security, essentially researching into the role and significance thereof if any that SIEM might have in IT security. Specifically the research sought to answer the questions of whether SIEM technology enhanced the ability to monitor and respond to application, system and network security events as in an environment comprising a high volume of security, network and device system logs.

1. Introduction

Connectivity to the Internet has expanded significantly since the early days of the Internet going public becoming more pervasive and ubiquitously available. As access to the Internet has grown, so has the range in the profiles of those connecting up to the Internet. This range of profiles has included: personal users, government and military institutions, commercial organisations and educational institutions among others (Daya, unknown). While, the development and growth of the Internet has led to a more interconnected society, enabling faster and easier information sharing, collaboration, data distribution and data consumption; the growth of the Internet and advances in technology have also in a significant way contributed to the changing information security landscape (Khelafa, 2007). For the most part this has seen a significant rise in information security threats and attacks (Khelafa, 2007). In a 2002 Computer Security Institute/Federal Bureau of Investigation survey on the then state of computer crime and security, 90% of the respondents comprising of business, state and academic institutions, acknowledged having experience a security breach (Computer Security Institute, 2002). As early as 2002, the survey by the CSI/FBI highlighted the increase in security threats as a result of Internet connectivity with up to 74% of respondents pointing to their Internet connectivity as a source of their majority security attacks (Computer Security Institute, 2002). A 2013 report on data breaches by Verizon, reported 92% of security breaches perpetrated by outsiders or external sources who would typically use the Internet as the medium of attack (Verizon, 2013).

Hosts with connectivity to the Internet have previously been the primary attack target in this widening of cyber-security threats (Hansman & Hunt, 2005). The increasing value and importance of information and data has however seen attacks more targeted towards the trusted secure internal network where most of this valuable information and data resides. External threats have not been the only cause of concern for information security professional. Insider threats have also been on the rise with more and more computer crime that is reported found to have been committed by the trusted insider (Ganapathi, Oliner & Xu, 2012). A 2013 Verizon reported noted that 14% of security breaches were traced to trusted insiders (Verizon, 2013).

Information security threats have over the last few years become more targeted, focused, sophisticated and complex (Barwinski, 2005). In the early years of computing however information security exploits were largely driven by a desire to prove and brag of technological skill (Khelafa, 2007). This changed over time as the drivers behind information security exploits shifted towards a highly criminal and profit making model (Blaich, Li, Liao & Striegel, A, 2010). This shift in exploit motivation has meant that security exploits have become more complex, possess faster propagation capabilities, include complex anti-detection invasive and evasive features (Hoefelmeyer, 2004), and have much more disruptive and damaging capabilities (Cichonski, Grance, Millar & Scarfone, 2012). These complex exploits are today created and executed within an almost commercial context far removed from the curiosity that was observed in the early computing days (Lozito, 2011).

With this background of evolving information security threats; threat monitoring though the analysis of event logs has grown to play a key role in information security beyond the original use cases of application debugging, network and system fault diagnostics (Pasquinucci, 2007). Through the collection, storage and analysis of event logs; information security professionals are able to gain a picture of the security state of applications, networks, systems, information and data (Grimaila, Myers & Mills, 2009). In addition; in the event of a security incidence event logs are now often the first place to look at when gathering information useful to investigating the security breach thus helping to answer important and relevant questions related to the security breach (Knight, 2010). In time, the ability to quickly gain knowledge of the inner workings of an attack through log event analysis has been a proven method for identifying and detecting suspicious activity on a network (Grimaila, Myers & Mills, 2011). This knowledge gained has also aided attack response, additionally assisting in the development of solutions that mitigate further attacks (Lambeth et.al., 1998).

Security Information and Event Management (SIEM) technologies are gaining prominence and acceptance within the IT security environment as more information security professionals gradually transition from the more simplistic Log Management tools and techniques (Jenkins, 2011). Log management technology provides for basic centralised event log collection and storage but is lacking in advanced event log analysis capabilities. This lack of advanced capabilities in log management has been associated with security event analysis that was very manual with little inherent capability for automation. Analytical capabilities of traditional log management tools have had to be incorporated via scripts and other customised techniques. The increasing volume of events generated on networks has however now meant that manual methods of log analysis have become untenable (Makanju, Milios, & Zincir-Heywood, 2009).

Regulatory and legislative compliance demands have also played a key role in log management adoption, being attributed for the increase deployment of log management tools by organisations and establishing log management as a permanent feature in the enterprise security architecture (Pasquinucc, 2007). Growing industry and federal regulatory compliance has however led to the acceptance that Log Management tools are unable to meet compliance demands and hence the shifting focus to SIEM.

As with Log Management tools, SIEM technology also provide for centralised storage of collected event logs sourced from application, network and systems devices deployed within an IT infrastructure (Kavanagn & Nicolett, 2011). However SIEM come with more advanced capabilities. These advanced capabilities include advanced filtering, aggregation, normalisation and correlation, alerting and reporting (Shipley, 2008). SIEM is said to enable real-time or near real-time event analysis. This thesis focuses on using SIEM as a tool to enhance IT security management as explained next.

In this research, the term Log Management tools will be used as a reference for tools that simply collect and store event logs without the advanced functionality of correlation, aggregation, filtering, normalisation, alerting and real-time analysis (Shipley, 2008). The term SIEM will be used with respect to event logging, storage and analysis tools that have the advanced capabilities of filtering, aggregation, correlation, normalisation, alerting and real-time analysis (Shipley, 2008).

1.1 Intended Contributions and Research Objectives

Log management and event logging have recently gained vast attention in past research (Maruyama & Yamanishi, 2005) spurred by the realisation of the value of logs in detecting security threats and satisfying regulatory compliance (Grimaila, Myers, Mills & Peterson, 2011). Observations have been made of the increase in researchers investigating this area (Maruyama & Yamanishi, 2005). Ganapathi, Oliner & Xu (2012) consider research in logs and event analysis as having become a “rich” (p.1) and Stearly (2004) noted research in this area as an “open problem” (p.2). A range of Log Management research areas have been addressed by past research, all seeking to contribute to advancing log management capabilities. Grimaila, Myers & Mills (2009) investigates use cases in Log Management as applying to insider threats, Makanju, Milios & Zincir-Heywood (2009) and Kliger, Mozes, & Yemini (1996) look at automating log analysis in order to address the largely manual log management techniques prevalent at the time, whilst Nabil (2009) investigates the combination of continuous running queries and filtering on real time streams of events logs for the purpose of improving event correlation. Koike & Takada (2002) researches into a log visualisation tool that enables the automated management of an increasing high volume of log data and Nagappan (2010) investigates log analysis techniques and log management targeted a "very large and complex logs" (p.1).

In spite of the advances in Log management, it is widely accepted that log management technology falls short in the current complex IT security landscape (Levin, 2009). Cognisant of the shortcomings of Log Management, research is now gradually focusing on SIEM as the future for log management, event analysis and compliance. Research in SIEM technologies is therefore now becoming an important research area in IT security (Kotenko, Polubelova & Saenko, 2012). As research into SIEM is still a developing area, a gap was observed by this researcher with respect to the relevance SIEM as it applied to IT security. The undertaking of this research was to fill this gap through exploring the applicability of SIEM technology in information security. Putting context to this research, the question could be asked of whether SIEM for security monitoring and compliance is hype or is a critical component in IT (Moffatt, 2013).

The thesis sought to answer the following research questions:

1. Are SIEM technologies applicable to IT security?
2. Does SIEM technology enhance the ability to monitor and respond to IT security incidences in an environment comprising a significant high volume of application, network and device system logs?

The research contributes to the body of knowledge by adding to the understanding about how SIEM fits into the contemporary multi-dimensional approach to IT security, and by developing an approach towards using SIEM in a high volume log environment.

1.2 Research Design

The undertaking of this study was to achieve the primary objective of adding to the body of knowledge of SIEM in IT security. Also equally critical was for this researcher to obtain a deeper and practical understanding of SIEM, an understanding that would be expected to be foundational to future SIEM in IT security research. The need to obtain a deeper and practical understanding of SIEM in IT security while contributing to the current body of knowledge in the area was the driver for this research to be conducted using the experimental research method within a positivist paradigm. (Grimaila, Myers, Mills & Peterson, 2011) employed the experimental research method to evaluate a distributed event correlation SIEM architecture against a centralized architecture. Similarly Aguirre & Alonso (2012) adopted an experimental approach in also investigating use of multiple SIEM in detecting security threats comparing that to a single centralised SIEM architecture. Coppolino, D'Antonio, Formicola & Romano (2011) also employed the experimental approach in studying SIEM integration with Critical Infrastructure Protection.

Past research observed to use the experimental methodology in SIEM research, was noted to make use of either automatically generated event logs stored in a SIEM or event logs manually and purposely generated for the researcher to have a richer set of relevant event data for analysis. This was the case in the research by Grimaila, Myers, Mills & Peterson (2011) where the event logs for the investigation was generated in a logging lab infrastructure utilising a proprietary tool call SAST generating what the researcher describes as "realistic traffic" (Grimaila, Myers, Mills & Peterson, 2011). Aguirre & Alonso (2012), use the OSSIM SIEM to experimentally investigate the use of federated SIEM in conjunction with a tool called Nikto. Nikto was used to scan a webserver with the purpose of generating purpose event logs for analysis. The scans in the research by Aguirre & Alonso (2012) are monitored with a SNORT IDS engine which then logged the traffic to the OSSIM SIEM for analysis. In both researches highlighted, the use of the experimental methodology enabled the researchers to control the variables in a way that would not have been possible on a real business network due to the risk that such experimentation might have.

The difficulty of conducting such research involving potentially high risk traffic on a business network is apparent. Such traffic is likely to trigger notifications requiring information security personnel response. There is also the risk of traffic generated for the purpose of experimentation causing unexpected behaviour on the business network, likely resulting in a breach of confidentiality, integrity or availability. For these reasons and that of this researcher seeking to gain a deeper practical understanding of SIEM in IT security while answering the research questions, the author chose the experimental research methodology to conduct this research. Dobbins, Lane and Steiner (1988) describe the use of laboratory experiments as valuable and particularly useful in experimental examination that is conducted on an individual level, which was the case with this research.

The components that were primary to the experiment in this research were A SIEM lab, relevant events logs- automatically and purposely generated and SIEM use cases. The importance and relevance of SIEM use cases is well addressed in literature as highlighted in Chapter 4. Past research mentioned in this Chapter all leverage SIEM use cases in their experimental research and hence the equal emphasis of SIEM use cases in this research, in particular in their use for exploring the applicability of SIEM in IT security.

1.3 Research Structure

The structure of this thesis follows a logical progression towards answering the research questions outlined. Each Chapter builds on the knowledge presented in the Chapter preceding it. Chapter 2 explores the evolution of IT security and is intended to provide a context of the environment in which SIEM are deployed while highlighting some of the key forces in the adoption of SIEM in IT security. Chapter 3 presents a review of log management, addressing log management within the context of information security. The architecture of log management and Log Management related challenges are also presented in Chapter 3. Chapter 3 lays the foundation for the discussion in of SIEM in Chapter 4. Chapter 4 presents a brief historical background to SIEM. The architecture of SIEM is also addressed in Chapter 4 allowing for comparison with the log management architecture. The Chapter provides a context of SIEM within information security. An important section in Chapter 4 is the examination of SIEM use case and use cases in General as this provides context and background relevant for the experimental analysis presented in Chapter 5. The presentation on SIEM use cases also lays the foundation for

Chapter 6. Chapter 5 presents details on setting up the experiment lab used in this research. Building on the overview of use case presented in Chapter 4, SIEM use cases are identified for exploration in the experimental lab. In Chapter 6 the discussion focuses on exploring SIEM while leveraging the use cases identified in Chapter 5. A discussion on the observations follows as well as conclusive arguments on the applicability of SIEM to IT security. Finally Chapter 7 is a summary of this research paper.

2. IT Security: A Historical Perspective

2.1 Defining IT Security

Anderson (2003) describes most IT security definitions as broad and lacking precision. Anderson (2003) further argues that there is no clear agreement on a definition of IT Security and that the several definitions that exist focus more on what IT Security does not what it is. Anderson (2003) then proceeds to propose a definition of IT security as "A well-informed sense of assurance that information risks and controls are in balance" (p.310). A similarly simplistic definition of information security is given by Hoppe, Pastwa & Sowa (2009) as "the protection of information from a wide range of threats in order to ensure business continuity, minimize business risk, and maximize return on investments and business opportunities" (p.204). Neumann (2003) infers computer security as representing a freedom from and resistance to danger, which danger in the context of IT security represents suspicious or malicious threats or attacks. In defining IT security, Peltier (2001) addresses physical and logical controls stating that IT security represents the prohibition of "unauthorized or accidental modification, destruction, disclosure, loss or access to automated or manual records and files as well as loss, damage or misuse of information assets" (p.266).

While the definitions just stated address the protection of information in general, the modern widely accepted definition of IT security focuses on the specifics of the protection of Confidentiality, Integrity and Availability (CIA) of information as the key tenets of IT security (Hawkins & Oblinger, 2006). The concept of Confidentiality, Integrity and Availability is commonly referred to as the CIA triad (Chapple, Stewart & Tittel, 2008). This protection of the confidentiality, integrity and availability of information is said to be the ultimate goal or primary objective of IT security (Chapple, Stewart & Tittel, 2008). IT security is therefore defined in terms of the CIA traid according to Parkin (1998). Confidentiality in the CIA triad, represents the assurance that only authorised users have access to protected "data, objects or resources" (p.180); Integrity represents the verification that objects have been "modified only by authorised subjects" (p.181) and Availability is ensuring that "timely and uninterrupted access to objects" (p.183) occurs for authorised users (Chapple, Stewart & Tittel, 2008).

With respect to the CIA triad, it can therefore be derived that any security protections including SIEM, put in place by organisations is designed to protect the confidentiality, integrity and availability of information and data.

2.2 IT Security – A Historical Perspective

As highlighted in the introduction, the growth and development of the Internet from a resource used mainly by the academic community to a key communication enabler (Avolio, 1998), has had a significant impact on computer crime driving its increase and rapidly transforming the IT security landscape. The growth of the Internet has been "fundamentally changing the computing environment" as Wiebschuch (2000) states. Other factors that have also contributed to a rapidly evolving information security landscape include, a largely unregulated Internet, the increasing number of interacting application instances, the increasing number of interconnected devices both at intranet and Internet level, the daily increase in the number of users of various profiles connecting up to the Internet (Botta et.al., 2007), (Doug & Kevin, 2011) and poorly secured systems which increase the exposure surface to information security exploits (Voloudakis, 2006).

The IT security landscape, IT security threats, information and its value have evolved and become more complex making the securing of information a much more demanding task (Lampson, 2004). What makes information security complex as stated is its dynamic nature. Even as the demand for information security to evolve at either the same rate as the emerging threats or at a faster rate is apparent, threats to information, data and computer infrastructure are emerging at a rate that information security professional and vendors are struggling to keep up with.

IT Security has a history that goes back more than 30 years (Lampson, 2004). In the early days of computing, access to computers was generally limited to large organisations that had the financial resources to own the costly and not generally available technology. As a result of this restricted access, computer crime was small in number and complexity compared to today. Limited forms of financial crime existed and computer crime was targeted more towards attempts to access "proprietary data" (D' Ovidio, 2007). At that time, IT security was considered an afterthought, mostly reactive (Voloudakis, 2006), "unimportant, unnecessary and already sufficient" (Avolio,

1998) rather than a critical issue. Physical security was the primary means of securing information through physical access restrictions to the then mainframes (Solms, 1999). Information security was also seen as a purely technical problem. By 1995, IT security was seen to have dropped down the priority list by IT executives (Whitman, 2003).

Interestingly, in spite of the low priority placed on IT security in the early days of computing, the first IT security models are said to have actually been developed in the early 1970s driven by a realisation of the need to put in place data and information protection mechanism (Groom, 2003). As information security was mostly technology focused, these security models were applied to mostly the IT security technology of the time, which then were focused on securing the network perimeter. The security technologies through which the perimeter was secured included firewalls, Intrusion Detection Systems (IDS) and routers. The more successful security models initially included access control lists, password based controls, subject/object access matrix models, "multi-level security using information flow", Public Key Infrastructure (PKI) and cryptography-based infrastructure (Lampson, 2004).

The development of IT security models was followed by consideration of the architecture of networks with concepts such of those of Demilitarized Security Zones (DMZs) being incorporated into network designs (Groom, 2003). Organisations at the time were realising that the security protecting their information and data had to go beyond just the securing the perimeter (Lozito, 2011). IT security protection mechanism continued to enhance as Virtual Private Network (VPN) technology, multi-layered authentication mechanisms, web proxies with malware, antivirus and URL reputation scanning capabilities (Lozito, 2011), advanced encryption and log management were incorporated into the IT security architecture. Other security technologies and concepts organisations are considering in the present for securing their networks include penetration testing and vulnerability scanning for the purpose of identifying vulnerabilities in both public facing and internal applications.

As these security technologies were being incorporated to maintain the confidentiality, integrity and availability of data and information; the role of event logs and log management as an IT security forensic tool was gaining traction (Koike & Takada, 2002). Consideration was being given of the importance that the collection and storage of event logs could have in post security breach forensic analysis (Koike & Takada, 2002). In addition to post security breach forensic analysis, the availability of event logs began to be appreciated in enabling information security professional to confirm and report on the effectiveness of deployed security measures and adherence to policy. Essentially event log collection and analysis allowed for validation that the technology in place was meeting security threat protection expectations (Casey, 2008).

Today, there is a realisation of the need for a significant shift in the IT security paradigm from a single-dimensional information security approach to a multi-dimensional approach (Avolio, 1998). The "sophistication and frequency" of what are today termed Advanced Persistent Threats (APTs), Trojans, worms, bots, targeted phishing attacks and exploitation of zero day vulnerability attacks (Lozito, 2011), is now forcing organisations to constantly review their perception and attitude towards information security (Brewer, 2012). Information security is now seen as more than just a technical problem, but rather a business problem (Parkin, 1998). With the consideration of IT security as more than just a technical problem the scope of security protection has been extended to the entrenchment of policies, procedures, risk management practices, regulatory compliance and log management (Solms, 1999).

The role of governments in regulating IT security is today a trend that is becoming more apparent and pervasive as evidenced by federal legislative regulatory requirements such a HIPAA, Gramm-Leach Bliley Act and to Sarbanes-Oxley (Moore, 2004). The demands for accountability and traceability from consumers, government, shareholders and the pressure from industry compliance and regulations such as the Payment Card Industry requirements highlighted the growing focus on log management (Casey, 2008), resulting in more organisations implementing some form of logging, analysis and secure storage capability (Pasquinucci, 2007). According to Levin (2009), a key requirement of such regulation is that organisations "collect, analyse, report on and archive all logs to monitor activities inside their IT infrastructure" (p.21), putting pressure on organisations "to collect and retain logs from systems that would otherwise not have been considered" (Casey, 2008).

2.3 Taxonomy of IT Security Threats

The principle of "Know the enemy, and know yourself, and in a hundred battles you will never be in peril" (Sun Tzu, unknown) is one that is commonly employed in IT security to illustrate the importance of information security professional understanding and appreciating IT security threat taxonomies (Kessler, 2004). According to

(Chuvakin, 2008) an understanding of the various types to threats is critical in the successful deployment of SIEM beyond satisfying compliance requirements. Various IT security threat taxonomies have been presented in past research. Early IT security taxonomies by while focusing on vulnerabilities and seen as “ambiguous” provided a background for later IT security threat taxonomies (Hansman & Hunt, 2005). An understanding of IT security threat taxonomies enables effective communication, ranking, prioritisation and response to computer threats and incidences (Hansman & Hunt, 2005) and enables the deployment of appropriate protection mechanisms (Whitman, 2003).

According to Laurie (2004), Internet based security threats can be classified into two broad categories of Protocol attacks and Malware attacks. Protocol attacks are said to threaten vulnerabilities in the implementation of protocols while malware threats are generally “executed” in order to be employed in a security breach. Protocol attack examples include cross-site scripting and SQL injection attacks whereas malware attack examples are viruses, Trojans, worms etc. (Laurie, 2004). Stevens (2006) classifies information security threats into four major categories: Network intrusions, viruses, worms, rootkits and Domain Name Service threat. Gordon, Loeb & Zhou’s (2011) taxonomy of IT security threats classifies IT security threats in terms of their attack on the confidentiality, integrity or availability of information and data. According to Gordon, Loeb & Zhou (2011), confidentiality threats target unauthorised access, availability threats attempt to prevent access by authorised user to systems, and integrity threats, attack the validity of data and information. Jariwala & Jinwala (2009) present taxonomy on attacks targeted at AES encryption. Álvareza & Petrovic´ (2003) presents a staged taxonomy of web attacks that classifies IT security threats on the basis of the attack life cycle looking at 9 life cycle stages of Entry Point, Vulnerability, Service, Action, Input Length, Target, Scope and Privilges. Through interviews and of review of past research, Whitman (2003) identifies 12 generic threat categories which are:

- Act of Human Error or Failure
- Compromises to Intellectual Property,
- Deliberate Acts of Espionage or Trespass,
- Deliberate Acts of Information Extortion,
- Deliberate Acts of Sabotage or Vandalism,
- Deliberate Acts of Theft,

- Deliberate Software Attacks,
- Forces of Nature,
- Quality of Service Deviations from Service Providers,
- Technical Hardware Failures or Errors,
- Technical Software Failures or Errors and
- Technological Obsolescence.

The OWASP (OWASP, 2014) top 10 project is today an important web application security threats taxonomy reference used by information security professional in improving the security status of their web applications. An appreciation of the OWASP top 10 web application security threats classification illustrates the importance IT security professional gaining and maintain a knowledge of IT security threat taxonomies.

3. Log Management in IT Security

3.1 Log Management in IT Security

Enabling logging in applications has been a long standing practice (Park, Yuan & Zhou, 2012). Today, enabling logging is such a widely accepted practice that most devices, applications and systems are inherently designed with the ability to generate some form event logs (Makanju, Milios & Zincir-Heywood, 2009). Up until the 1990s, logs were generally used for network, application and system debugging, performance monitoring, troubleshooting and fault finding and rarely used for security monitoring (Grimaila, Myers, Mills & Peterson, 2011). In addition logs were mostly considered important when a problem arose and not so much for proactive or real-time monitoring (Lobo, 2003). With the growth and expansion of the Internet, log use cases began to be extended to for instance statistical reporting and business intelligence for marketing purposes (Casey, 2008). Log management use cases were further extended into IT security for security breach identification, monitoring for policy violations, fraudulent activity monitoring and discovery, post security breach forensic analysis and regulatory compliance among other IT security related log use cases (Kent & Souppaya, 2006).

As more networked servers, desktops, mobile devices were deployed and the number of threats against networks and systems continued to rise; the volume and range of computer security event log equally greatly increased creating the need for IT security log management (Kent & Souppaya, 2006). The value of log management in security threat detection and compliance has continued on the upward trend since the early (Grimaila, Myers, Mills & Peterson, 2011). In a SANS 2010 survey, 63% of respondents placed the need to collect logs for "Detect/prevent unauthorized access and insider abuse" as critical (Shenk, 2010). As Figure 3.1 illustrates, an increase of about 50% was observed in log management adoption between 2006 and 2010. Event logs when used for IT security forensics help to answer the questions of what happened, when it happened, who triggered the event, and why it happened enabling IT security professionals to reconstruct events during or after an incident (Marty, 2011). With this growing recognition of the importance of logs, the concept of Log Management as a wider process for managing security related event became an integral process in IT security (Casey, 2008).

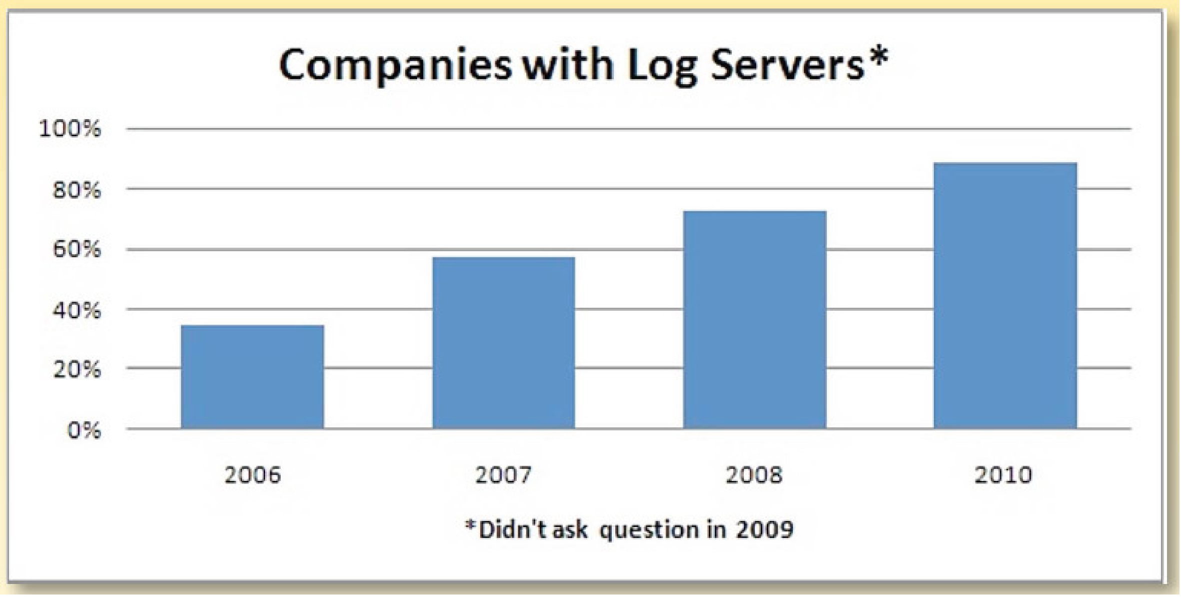


Figure 3.1. Companies with Log Servers (Shenk,2010)

Log management is the process of "generating, collecting, storing, retrieving and disposing of logs" (Grigorescu & Becker, unknown). Log management tools have traditionally been leveraged to collect and store raw event log from disparate sources such as operating systems, devices and applications (Gharaee, Madani & Rezayi, 2011). As already highlighted, the initial use of log management tools focused on the detection of faults in applications and devices (Grimaila, Myers & Mills, 2011). This has slowly changed over time with log management tools being leveraged more in identifying and detecting suspicious security related behaviour on networks (Grimaila, Myers, Mills & Peterson, 2011).

A Log can be defined as "a record of the events occurring within an organization's systems and networks" where each entry or event within the log contains a record of specific information related to the event (Kent and Souppaya, 2006). Logs and associated log entries or log messages contained in the logs is information that applications and other information technology devices use to report on the systems status, activities that would have occurred within the system and related actions taken (Pasquinucci, 2007). By configuring application and systems to generate logs security professionals are able to gain a picture of the existing state of applications, network and systems (Grimaila, Myers & Mills, 2011).

An event on the other hand can be defined as "is a single occurrence within an environment, usually involving an attempted state change." (CEE, 2010). The environment could be any of the log sources within an IT infrastructure (Chuvakin, Phillips & Schmidt, 2013). According to Marty (2011), event logs are generated at multiple layers of the network infrastructure stack including the application layer, the transport layer, the network layer and the physical layer in response to either stimuli or defined occurrences within the log source. Events logs can be classified into two broad categories, operational event log and security event logs (Gorge, 2007). Operational event logs as expected are useful for the monitoring of system status such as whether a system is online or offline or system crashes. Security logs on the other hand are useful in threat monitoring. Security logs are also useful for monitoring activity such as unauthorised access or change management related activity such as what new accounts have been created or deleted (Gorge, 2007).

Events logs can also be said to "Fat" or "Lean" depending on how much context is available within the events. Fat logs contain adequate context within each event, meaning the events themselves are a lot more descriptive. Lean logs on the other hand don't have much context around the event messages (Grigorescu & Becker, unknown). Event logs are normally assigned severity levels in order to assist in prioritisation. The most common severity levels including informational, debug, warning, error and alert (Chuvakin, Phillips & Schmidt, 2013).

Log analysis and log management are a "well-studied but open problem" (Stearly, 2004). Log management has little value where there is no associated log analysis (Knight, 2010). While log analysis has been considered an important component of IT security it has also been appreciated as a difficult and sometimes complex process in particular as the volume of event logs and the complexity of IT security threats increases (Ganapathi, Oliner & Xu, 2012). As a result of this sometimes difficulty in analysing and interpreting event logs, event log analysis is said to require some level of expertise in order to correctly interpret the information contained within the logs (Koike & Takada, 2002). An important part of this expertise is an understanding of IT security threat taxonomies as presented in Chapter 2.

Traditionally log analysis has been manual, requiring much time and resources; expensive and prone to inaccuracies (Nabil, 2009). Analysing the event logs without using specially developed and custom tools has been described by Dostal, Javornik, Ledvinka & Slavicek (2008) as "out of human capability". This challenge has had to be addressed in order to better manage events in today's larger sized networks (Kliger, Mozes, & Yemini, 1996) and hence the trend in moving towards systems with more advanced capabilities aimed at making the log management and analysis process more efficient (Nabil, 2009). In this regard Log management today, is transitioning from the simple collection and storage of log data to more complex functions of log normalization, classification, prioritization, correlation and complex analysis (Gharaee, Madani & Rezayi, 2011). These capabilities as will be presented in Chapter 4 are being found SIEM. The emergence of SIEM technology is therefore seeing as a transition from log management (lobo, 2003).

3.2 Log Management Architecture

The Log management architecture is a simple architecture composed of the three components namely: Log generation or acquisition; Log transportation or transmission and Log storage. Figure 2 illustrates a simplistic representation of the log management architecture where acquisition represents log generation and transmission/relay represents log transportation. A brief outline of these components follows.

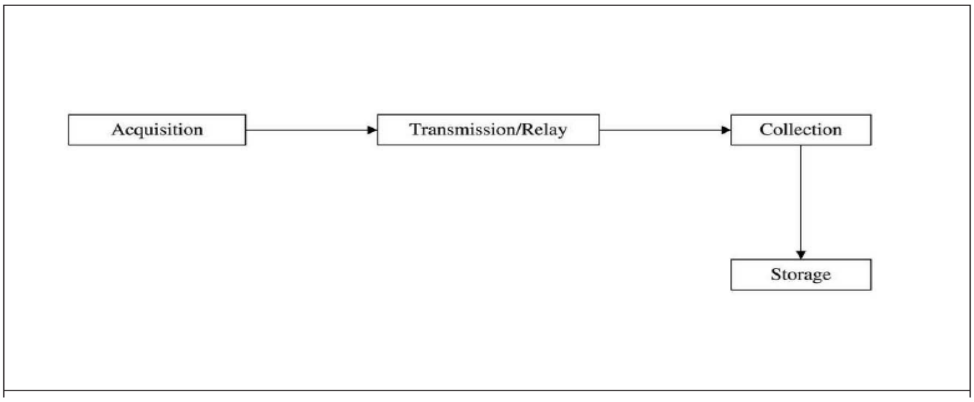


Figure 3.2. Log Flow (Forte, 2004)

3.2.1 Log Generation

When configured appropriately, the majority of IT applications, network and security devices have the capability of generating some form of event messages (Forte, 2004). These event messages when generated and sent to centralised log management storage enable the Log management environment. When configured to generate event messages, applications, network and system devices generally first store these events locally and also forward the events to centralised storage if available. The majority of systems have stored these logs locally in simple flat or binary log files, compressed or uncompressed (Chuvakin, Phillips & Schmidt, 2013). Local storage is usually limited by the disk space size of these systems as this space is mostly shared with the operating system. To avoid filling up the disk space with event log messages applications, network devices and systems in log management have been configured to send logs to either a centralised or distributed log management facility (Marty, 2011). In log management, the applications, systems or network devices that generate events logs are referred to as Log Sources. The log management architecture is therefore composed of log sources from which event logs are generated in response to triggers or stimuli as defined by the application developer, system or network device vendor.

Within a single log source, there could potentially be multiple components generating events. For instance on a log source running the Windows operating system, components events could be generated by components such as web servers, Active Directory, the local firewall or installed third party applications (Pasquinucci, 2007).

A log management environment could be composed of either push based log sources or Pull-based log sources or both (Chuvakin, Phillips & Schmidt, 2013). With Push-based log sources, the log sources send logs from the device, system or application to a centralised or distributed log host. With pull based log sources the centralised log host pulls the event from the log sources (Chuvakin, Phillips & Schmidt, 2013).

3.2.2 Log Transport

A method of transportation is required for an event message to be delivered to centralised or distributed log storage. Log transportation therefore defines the movement of logs from a log source to a log storage host. There currently exist no standards with regards to event message transportation even with the current range of transport protocols used by various vendors (Lozito, 2011). The syslog protocol has however for many years been one of the most widely used log transport protocols to the extent of almost becoming the de facto event log transportation standard (Casey, 2008). Other log transmission methods some proprietary and other open also exist but are not as widely used as the syslog protocol. These other log transmission protocols include IDMEF3, the Simple Network Management Protocol (SNMP), Windows Management Instrumentation (WMI), Webservices management (WS-management), Simple Object

Access Protocol (SOAP) over HTTP (Chuvakin, Phillips & Schmidt, 2013), LEA checkpoint, SDEE cisco and E-streamer proprietary by Sourcefire. Of the available communication protocols, IDMEF3 and syslog are said to have the most detail and comprehensiveness with regards to log format (Gharaee, Madani & Rezayi, 2011).

3.2.3 Log Collection and Storage

Storage of event logs relates to the collection of events for the purpose of short or long-term archival. Log storage, normally residing on what is referred to as the Log Host can take various forms for instance a physical or virtual host running any of the available operating systems (Chuvakin, Phillips & Schmidt, 2013). Log storage can either be centralised or distributed. While traditionally log storage has been centralised there recently has been research into comparison of the benefits of distributed log collection and storage environments versus centralised log storage environments (Grimaila, Myers & Mills, 2011). In a centralised log storage model, event logs are sent to a single log host, on which log analysis is then done. Because of the volume of events generated, a centralised logging model could place significant memory and CPU resource demands on the central host (Grimaila, Myers & Mills, 2011). The Distributed log storage architecture on the other hand allows for event logs to be sent and stored on multiple log hosts located at strategic points within the network. With both the centralised and distributed logging architectures, the purpose is to have key central points within the log management architecture for event log analysis.

In the early days of computing, the cost of log storage significantly limited logging capability with respect to the volume of logs that could be generated, stored and retained (Casey, 2008). The result then was that mostly critical events were stored on the central log host (Casey, 2008). Then however, log retention was not an important consideration. The continuously falling cost of storage has how however meant that today, the volume of event logs that can be stored and retained is now significantly higher. More data is therefore being logged and made available for analysis, in fact according to Casey (2008) there is now the problem of “too much data”. The availability of large storage capacity how worked in favour of organisation who have to meet strict log retention compliance requirements such as from the PCI DSS standard (Kent & Souppaya, 2006).

Available logs storage options in log management have varied including databases, single or multiple files or data streams (Nagappan, 2010). Event logs have traditionally been stored in flat files; however databases are now becoming a common storage format for logs providing a structured method of event log storage (Chuvakin, Phillips & Schmidt, 2013). Propriety log storage formats are also common place, examples of which include the windows event log storage format (Chuvakin, Phillips & Schmidt, 2013). Other log storage formats include flat text-based log files, indexed flat text files, binary files, databases or Hadoop log storage (Chuvakin, Phillips & Schmidt, 2013).

3.3 Event Log Format

The PCI DSS standard (PCI Security Standards Council, 2013) expects that event log message should at a minimum contain the following data elements:

- User identification
- The type of event
- The date and time
- Whether the action that trigger the event was successful or not
- The identity of the network component that produce the event

Marty (2011) also suggests that at a minimum, the following data elements should be present in event message: the timestamp, the application that triggered the event, the user associated with the event, the session ID, the severity of the event and categorisation detail to allow for ease of event classification. This is however not always the case where event message will contain all the expected elements whether as stated by PCI DSS or as suggested by Marty (2011). There exists no industry standard that defines the format of event logs, with log formats varying across the many devices and applications from the various vendors (Ganapathi, Oliner & Xu, 2012). In addition to the varying formats, the event logs themselves are presented in a very unstructured manner (Nabil, 2009). Mostly at a basic level however, a typical log entry contains a log header, a timestamp and a description of the event (Stearly, 2004). The varying event log formats have been observed to add to log analysis challenges in particular with respect to log management tools (Marty, 2011). In addition to the varying event formats, insufficient contextual information in event logs has been seen to contribute to log analysis challenges (Oliner & Stearley, 2007).

3.4 Log Management Challenges

As observed in the previous section, for a long time and stretching into the present, there has been a lack of standards in relation to event log transport protocols as well log formats (Nagappan, 2010). This was also observed by this researcher while analysing events in the Splunk SIEM lab setup for this research. The unstructured nature of event logs has therefore been a well-known challenge in log management (Chen, Fan, Li & Zhang, 2008). Kent & Souppaya (2006) also observes log content inconsistencies as known challenge in log management. While Grimaila, Myers, Mills & Peterson (2011) note that the ability to detect insider threats increases as the number of log sources increases, the high volume of events collected and stored has been accepted as making log analysis more difficult and challenging (Lobo, 2003). As previous stated the lack of context and useful information in log messages also add to log management challenges (Oliner & Stearley, 2007). Manual analysis of logs inherent in log management is known to be tedious and prone to errors, largely due to again format of the event logs and the volume of events generated (Koike & Takada, 2002).

4. SIEM in Information Security

4.1 Background to SIEM

As presented in Chapters 2 and 3, the role of a centralised or distributed event logging framework has become established as the IT security landscape has evolved. Log management initially filled this gap, providing a centralised storage option but allowing for mostly manual and limited analysis capabilities. The lack of advanced capabilities and the inability for log management to scale when working with large volume security event log datasets have however been key drivers in the transition to SIEM (Lozito, 2011). The growing reliance on event logs in incident forensic investigations (Bishop, 2003) and the need to readily have relevant information available for use in the forensic investigations have also been recognisable contributors in the shift from log management to SIEM (Aguirre & Alonso, 2012).

SIEM whose history goes as far back as 1997 refers to the “collection, logging, and analysis of system and application events to identify potentially malicious activities and system errors” (Grimaila, Myers, Mills & Peterson, 2011). The collection of these events can either be in real or near real-time. SIEMs emergence in IT security was to provide an advanced tool for managing the large volumes of events coming especially from high volume event generating devices such as firewalls, IPS and IDS. The term SIEM came about with the merging of the two different technologies of Security Information Management (SIM) and Security Event Management (SEM) (Hoppe, Pastwa & Sowa, 2009). At the core of their functionality, SEM and SIM technologies both involved the collection of events logs (Gabriel, Hoppe, Pastwa & Sowa, 2009). SEM however as a technology, was focused on real-time monitoring, correlation normalisations and incident management (Kavanagn & Nicolett, 2011) while SIM was more inclined toward the log management capabilities which were presented in Chapter 3, emphasizing on event log collection, storage and enabling analysis (Gabriel, Hoppe, Pastwa & Sowa, 2009). Figure 4.1, shows the high level concept of the merging of SEM and SEM in SIEM.

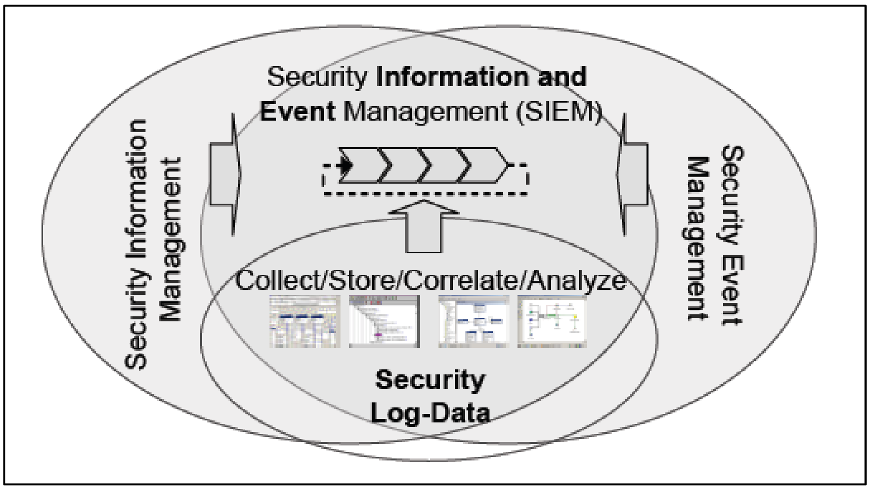
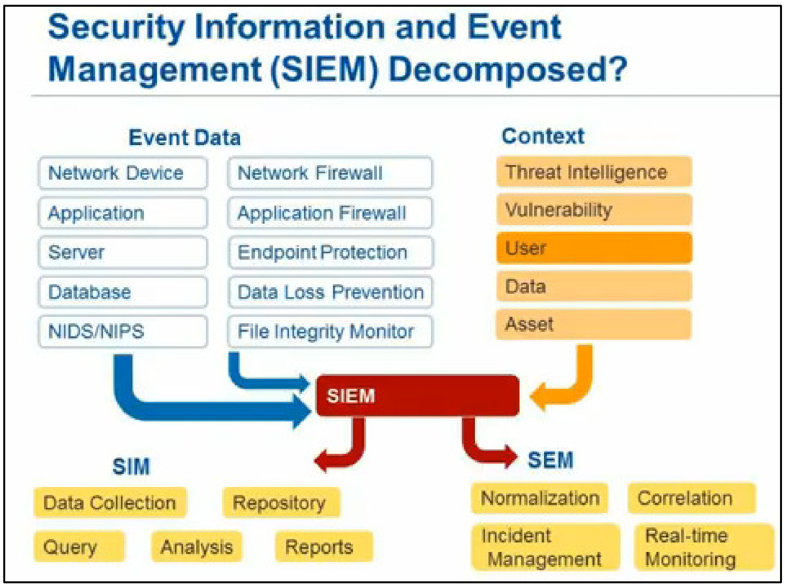


Figure 4.1. Comptual architecture of SIEM. (Hoppe, Pastwa & Sowa, 2009)

The diagram in Figure 4.2 provides more detail on the merging of SIM and SEM illustrated in Figure 4.1. Figure 4.2 illustrates the individual SIM and SEM capabilities, demonstrating how the merging of these technologies resulted in a more capable single technology of SIEM. SIEM as with log management collects security events logs generated by a growing variety of logging enabled heterogeneous log sources. These log sources include end-user operating systems, server operating systems, routers, firewalls, Active Directory, Anti-Virus systems, IPS, VPNs and Web application firewalls among other log sources.



Security Information and Event Management (SIEM) Decomposed?

Event Data

Network Device
Application
Server
Database
NIDS/NIPS

Network Firewall
Application Firewall
Endpoint Protection
Data Loss Prevention
File Integrity Monitor

Context

Threat Intelligence
Vulnerability
User
Data
Asset

SIEM

SIM

Data Collection
Repository
Query
Analysis
Reports

SEM

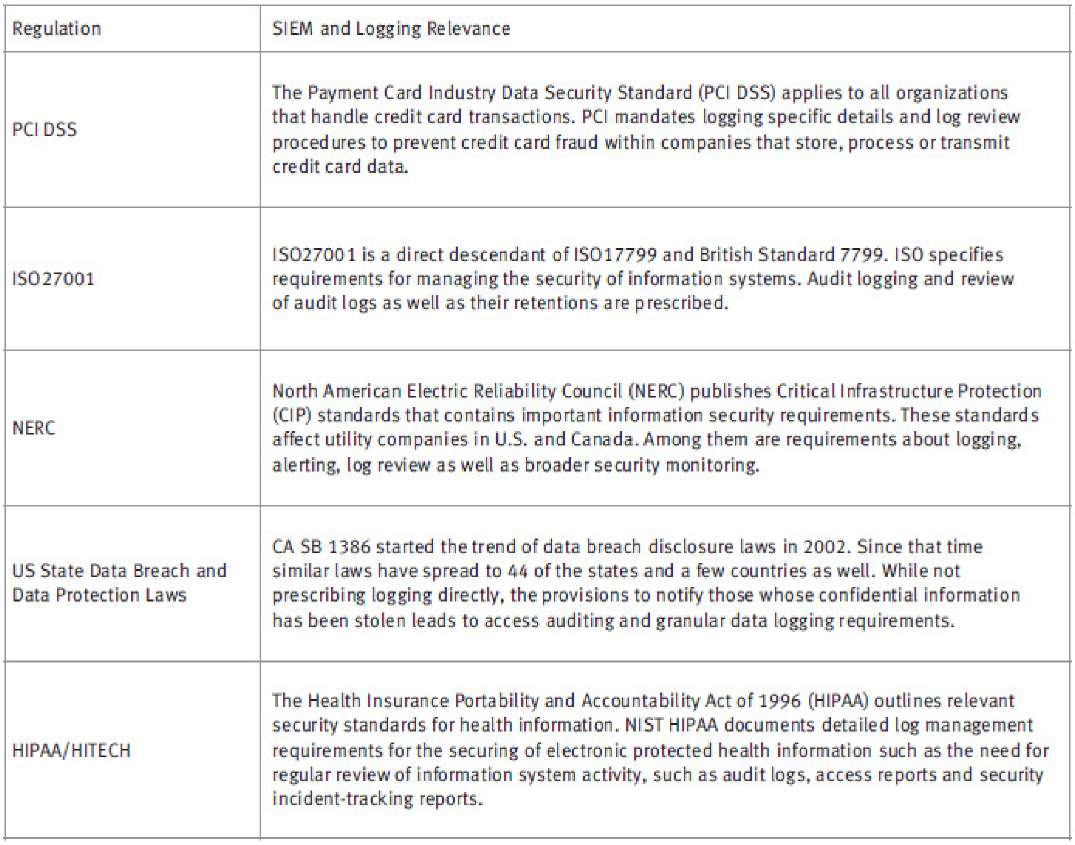
Normalization
Correlation
Incident Management
Real-time Monitoring

Figure 4.2. Merging of SIM and SEM

As with log management and many of the security technologies available today, SIEMs presence in the IT security architecture is an additional dimension to the multi-dimensional approach required in handling complex information security threats (Verizon report, 2013). As already stated, the value of SIEM lies in its capabilities which include normalization, correlation, event log parsing, filtering, alerting, reporting and support for a wider range of event log formats compared to log management (Aguirre & Alonso, 2012). Some SIEM implementations have had these capabilities further extended through capabilities such as in-built security knowledge bases, incident and case management capabilities and asset reporting enabling a consolidated view of the security status of an IT security environment (Kent & Souppaya, 2006). Capabilities such as fast searches even when searching through large volume data sets and contextualisation through data enrichment have also enhanced the value of SIEM in information security (Misnomer, 2012a).

SIEM capabilities have empowered security professional with an extended list of use cases. The promises of having this extended list of use cases being: to enable proactive threat monitoring and quicker response to potential threats, the reduction of false positives observed in IDSs systems (Coppolino, D'Antonio, Formicola & Romano, 2011) allowing for reporting on information security threats to be as accurate as can be achieved (Hutchison, 2009) and the improvement and potentially shortening of the security incident lifecycle through enabling more advanced post security breach forensic capability. Hoppe, Pastwa & Sowa (2009) notes that, compared to log management, a "critical success factor" in the deployment of SIEM relates to the quality for information it provides in assisting IT security related decision making.

The role of regulatory compliance in IT security was presented in Chapter 2 and is a continuing theme and area of interest in this research. From a regulatory compliance perspective, affected organisations are required to collect, store and archive, retain for a specified minimum period, report on and regularly analyse the collected event logs (Kavanagn & Nicolett, 2011). Figure 4.3 highlights the some regulations and their SIEM related logging relevance.



Regulation	SIEM and Logging Relevance
PCI DSS	The Payment Card Industry Data Security Standard (PCI DSS) applies to all organizations that handle credit card transactions. PCI mandates logging specific details and log review procedures to prevent credit card fraud within companies that store, process or transmit credit card data.
ISO27001	ISO27001 is a direct descendant of ISO17799 and British Standard 7799. ISO specifies requirements for managing the security of information systems. Audit logging and review of audit logs as well as their retentions are prescribed.
NERC	North American Electric Reliability Council (NERC) publishes Critical Infrastructure Protection (CIP) standards that contains important information security requirements. These standards affect utility companies in U.S. and Canada. Among them are requirements about logging, alerting, log review as well as broader security monitoring.
US State Data Breach and Data Protection Laws	CA SB 1386 started the trend of data breach disclosure laws in 2002. Since that time similar laws have spread to 44 of the states and a few countries as well. While not prescribing logging directly, the provisions to notify those whose confidential information has been stolen leads to access auditing and granular data logging requirements.
HIPAA/HITECH	The Health Insurance Portability and Accountability Act of 1996 (HIPAA) outlines relevant security standards for health information. NIST HIPAA documents detailed log management requirements for the securing of electronic protected health information such as the need for regular review of information system activity, such as audit logs, access reports and security incident-tracking reports.

Figure 4.3. SIEM and logging relevance (Chuvakin, 2010)

Log management was deployed to satisfy regulatory compliance prior to the emergence of SIEM. Available SIEM capabilities of normalisation and correlation have however seen a growing number of organisations replacing log management with SIEM for regulatory compliance, (Jenkins, 2011). (Chuvakin, 2008) points to regulatory compliance as actually becoming the primary reason organisations adopt and deploy SIEM rather than the original use case of SIEM as a platform which enables the detection of IT security threats. According to Gartner (2010), over 80% of SIEM deployment in 2010 were initiated to close a compliance gap.

4.2 SIEM Architecture

This section presents an overview of the SIEM architecture. The SIEM architecture is an integrated system of a number of components whose advances have built on the foundation laid by the log management architecture. As has been stated frequently in this paper, the core architectural SIEM components are:

- Log collection and high capacity storage
- Filtering
- Parsing
- Normalisation
- Rules
- Data enrichment and contextualisation
- Correlation
- Event Aggregation
- Real-time analysis, Alerting and Reporting

Figure 4.4 illustrates the relationships and data flow between SIEM components. The diagram in Figure 4.4 highlights the role of filtering, normalisation and correlation in the processing of event log data ultimately allowing for analysis, alerting and reporting on potential or real IT security threats.

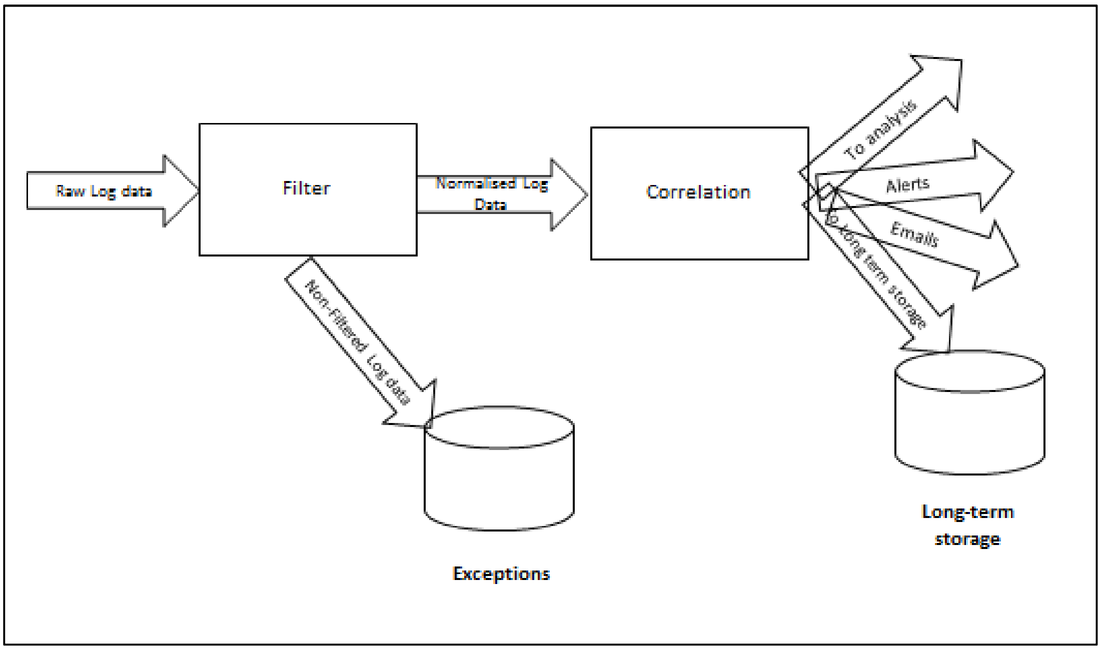


Figure 4.4, Flow for filtering and correlation (Chuvakin, Phillips & Schmidt, 2013)

Log Sources

As can be observed from Figure 4.3, a SIEM is without value in the absence of events logs. Log sources therefore are an important component of the SIEM architecture. One of the initial steps in the deployment of SIEM is to identify and configure log sources to send events logs to the SIEM (Chuvakin, 2008). While the log sources send event logs to the SIEM in a range of event formats lacking in standards, SIEM are designed to support and parse these varying event log formats either natively or with manual intervention (Kent & Souppaya, 2006).

The range of devices and applications that can be configured to generate event logs continues to grow. Most vendors now incorporate in their application or devices the ability to generate some form of event logs. As with log management, SIEM event log collection can be through the pull method or push method. Whereas with the pull method, an integrated Log collector pulls events from the log source (Kent & Souppaya, 2006), the push method on the other hand requires that the log source send events to the SIEM. Agent-based log collection and agent-less based log collection are also terms associated with the collection of events in SIEM. With Agentless log collection, no software is required to be installed on the log source. In this case either the log collector pulls the log from the log source of the log source pushes the logs to the log collector. With agent-based, software is required to be installed on the log source where the agent collects events from the log source and forwards these events to the SIEM in either real-time or near real-time (Kent & Souppaya, 2006).

Filtering

Kent & Souppaya (2006) describe event filtering as "the suppression of log entries from analysis, reporting, or long-term storage because their characteristics indicate that they are unlikely to contain information of interest". As the number of devices sending event logs to SIEM increases, the volume of events logs generated is expected to equally grow. Mostly log sources are configured to send all event logs, relevant or irrelevant to the central log collector. The increasingly large data set of events has meant that reviewing event logs is a time and resource consuming task in particular when having to deal with a mix of relevant and irrelevant event logs (Chuvakin, Phillips & Schmidt, 2013). Filtering or data reduction under the SIEM architecture addresses this challenge of the large volume of relevant and irrelevant logs coming through to the SIEM. The principle of event filtering has been to ensure that the value of event logs is not lost in the mix of relevant and irrelevant event logs (Jiang, Huang & Zhang, 2012) by keeping events “containing information of interest” while discarding those events they are not of interest (Kent & Souppaya, 2006).

Normalisation

As highlighted in Chapter 3 and in this Chapter, events generated by the various devices on the network are of varying formats due to the lack of event message standardisation among applications and device vendors. This lack of standards in event log format has resulted in challenges with analysing and correlating raw log events (Grimaila, Myers, Mills & Peterson, 2011). The primary purpose of normalising event data within the context of SIEM is to overcome the lack of structure inherently found in events by parsing them into a common format (Chuvakin, Phillips & Schmidt, 2013). In essence, Normalisation takes varying event logs formats, extracts key data elements in the event messages and places the data elements it appropriate fields in the SIEM (Grimaila, Myers, Mills & Peterson, 2011). By doing so, normalisation feeds into and enables correlation and the ability to report on similar types of events which in their raw format would be difficult to report and analyse (Grimaila, Myers, Mills & Peterson, 2011). Common data elements that are parsed and normalised includes for instance date and time, usernames, IP addresses, ports, timestamps and event priority (Chuvakin, Phillips & Schmidt, 2013). Normalising should be done in a manner that does not violate the integrity of the original event, an important requirement in IT security forensics (Forte, 2004).

Rules

Rules in SIEM are designed to identify and categorise events as they are logged into the SIEM (Hutchison, 2009) enabling the differentiation of normal and abnormal traffic patterns (Koike & Takada, 2002). These rules in SIEM deployments are also referred to as signatures, a term drawn from signature based systems such as IDS/IPS and antivirus systems. The setup of rules in SIEM enables alerting, where events matching specific monitored rules can be alerted and reported on (Coppolino, D’Antonio, Formicola & Romano, 2011). Current SIEM technologies are based on static rules capable of only identifying known behaviour (Hutchison, 2009). These static rules are typically developed using logical expressions which can either be activated or not depending on the events coming through to the SIEM (Hutchison, 2009). The use of dynamic rules is a focus of the future of SIEM where artificial intelligence based anomaly detection is integrated into dynamic rules designed to identify previously unknown behaviour (Hutchison, 2009).

Data Enrichment and Contextualisation

Data enrichment in the SIEM architect is aimed at providing context to event data collected by the SIEM. This context enables richer analysis through the availability of relevant event related additional information (Knight, 2010). Generally events messages do not natively include context, the lack of which could result in the misinterpretation or the assigning of multiple interpretations to the meaning of any particular log message (Ganapathi, Oliner & Xu, 2012). Depending on the log messages, the lack of context can in some instance make log messages difficult to understand (Ganapathi, Oliner & Xu, 2012).

Correlation

Correlation is perhaps one of the most important capabilities of SIEM. According to Casey (2008) the "correlation of logs from security and application-related sources is vital for organisations as it allows alerting mechanisms to be fine tuned". Correlation in SIEM is the establishment of relationships among events drawn from multiple log sources enabling the effective identification of potential and real security attacks which would otherwise have not been obvious when considering only one data source. (Casey, 2008). This is illustrated in Figure 4.5, where for instance the ability to detect insider threats increases as the number of events and log sources increases (Grimaila, Myers & Mills, 2009).

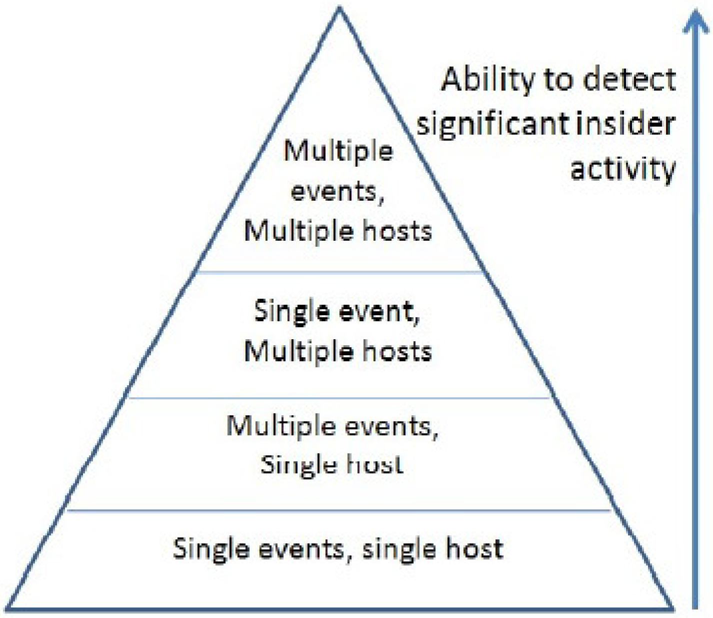

Ability to detect significant insider activity
Multiple events, Multiple hosts
Single event, Multiple hosts
Multiple events, Single host
Single events, single host

Figure 4.5. A hierarchy of event collection in terms of the ability to detect insider activities (Grimaila, Myers & Mills, 2009)

Correlation in SIEM is applied to data elements which as is illustrated in Figure 4.4 would be parsed and normalised into appropriate SIEM field values. For instance parsed and normalised data elements such as timestamps, IP addresses, user information and ports can be referenced in correlation (Schultz, 2010). Event correlation in SIEM requires a deep understanding of a network's architecture, threat taxonomies, log sources, the meaning of individual events and an appreciation of the parsed and normalised data elements (Liu, Mok & Yang, 2010). An understanding of the "causal and temporal relationships among events" is also important to effective correlation (Liu, Mok & Yang, 2010).

Correlation and associated correlation techniques has been a subject of various past research and for this reason there exists a number of correlation methods that have been proposed. Rule based correlation is perhaps the most common and basic form of correlating events (Varandi, 2002). Other approaches to event correlation found in literature include model-based correlation (Liu, Mok & Yang, 2010), finite state machine based correlation (Grimaila, Myers, Mills & Peterson, 2011), graph based correlation (Varandi, 2002), codebook based correlation (Grimaila, Myers, Mills & Peterson, 2011) and case-based reasoning correlation (Grimaila, Myers, Mills & Peterson, 2011). Each correlation approach as expected has its own unique benefits and limitations, the use of which is strongly environment dependent (Grimaila, Myers, Mills & Peterson, 2011). Some SIEM vendors have also come up with correlation methods used in products. For instance OSSIM allows for three types of correlation which are: Inventory correlation, Cross correlation and logical correlation. In an OSSIM context Inventory correlation is asset focused, Cross correlation compares events with vulnerability analysis results and Logical correlation as the name implies is correlation based on logical expressions (Coppolino, D'Antonio, Formicola & Romano, 2011).

Aggregation

Event aggregation in the SIEM architecture refers to the consolidation of similar event entries which would have occurred within a specified timeframe into a single event entry. Aggregation is directed at log reduction and designed to improve SIEM performance (Kent & Souppaya, 2006).

Storage

Log Storage and associated retention policies are some of the key requirements for audit and regulatory compliance (Coppolino, D'Antonio, Formicola & Romano, 2011). Storage on log sources has traditionally been based on various file based log storage formats which have been any one of text-based, binary log files or compressed log files (Chuvakin, Phillips & Schmidt, 2013). Text file log storage formats were used in early log management technology (Chuvakin, Phillips & Schmidt, 2013). Text-based files with indexing capability were also introduced at some point to enable the sorting and querying of stored event logs (Chuvakin, Phillips & Schmidt, 2013). Current SIEM systems however tend to use database technology to store event logs. The database technology used by today's SIEM is either proprietary or leverages third party database systems such as Microsoft SQL server, Oracle database or MySQL database. (Chuvakin, Phillips & Schmidt, 2013).

4.3 SIEM and Event Analysis

Event log analysis involves the interrogation, aggregation and correlation of events to derive meaning from them. Deriving meaning from collected events within a log storage infrastructure can be said to be one of the ultimate goals of deploying SIEM (Chuvakin, Phillips & Schmidt, 2013). Event log analysis is what was performed in the Chapter 6 as part of exploring the applicability of SIEM to IT security. In deriving meaning from the event logs information security professionals are able to gain an understanding of the security status of a network, meet compliance and regulatory demand and respond to security incidences (Chuvakin, Phillips & Schmidt, 2013). Because of the complexity and advance seen in some of today's security threats, skill, experience and rigour are said to be important attributes of performing log analysis (Arasteh, Debbabi, Sakha & Saleh, 2007).

Two common approaches to log analysis observed in literature are the Top-down log analysis approach and the bottom-up approach (Forte, 2004). The Top-down approach to log analysis is where the information security professional starts from observing an attack through say a tool such as SIEM and then working their way down to analysing the events that would have triggered the attack (Forte, 2004). With the bottom up approach; the information security professional starts at the individual log level then works their up correlating events on their way as required (Forte, 2004). The top-down analysis approach is commonly used in forensic investigations. Other approaches to log analysis observed include Expert systems analysis and Visualisation log analysis (Stearly, 2004). With expert systems the information security professional creates regular expression based rules that are then employed to monitor specific types of events. Alerts can be setup to notify the security professional when these rules are matched (Stearly, 2004).

A combination of the top-down approach and expert systems analysis where employed for experimental part of this research. The top approach was more applicable as it allowing for analysis starting from the observing an attack and then working down through the events. Correlation rules defined for the lab allowed the research to take an Experts Systems event analysis approach to exploring SIEM use cases identified for the experiment.

4.4 Use Cases

4.4.1 Background to Use Cases

The term Use Case introduced in 1987 by Iva Jacobson (Linic, 2007), is a concept that has been used for some time in system analysis and design for providing a way of defining user requirements (Burd, Jackson & Satzinger, 2004). According to Lee

(1999), Use Cases were also designed to assist in managing the complexity in specifying user requirements as well as assist in the development of user centric requirements (Linic, 2007). An important goal of clearly defining Use Cases is to allow users of a system to understand, verify, validate and ultimately test the system requirements once defined (Linic, 2007).

Various techniques have been proposed for identifying Use Cases. Perhaps the more popular use case identification techniques were defined by Burd, Jackson & Satzinger (2004). The use case techniques are: the User Goal Use case definition technique, Event decomposition Use case technique and CRUD Use case definition. (Misnomer, 2012b) also presents an 8 step SIEM use case identification methodology, which focuses specifically on SIEM use case identification. These techniques are briefly outlined in this section to give background and context to the concept of SIEM use cases, a concept that will be repeatedly referred to in Chapters 5 and 6.

4.4.2 User Goal Use Case Definition Technique

The User Goal technique employs structured interviews in an attempt to have users describe their goals when adopting a new system. As part of the interview, the user is guided through specifying in explicit terms the goals expected to be achieved in deploying the system. The User Goal technique is defined as an 8 step process composed of: system user identification, potential user classification by functional role, potential user classification by organisational level, user interview and goal identification, Use Case preliminary definition, use case refinement, user-use case association and finally review of defined use cases.

4.4.3 Event Decomposition Use Case Definition Technique

The Event Decomposition use case identification technique is said to be more comprehensive than both the User Goal and Crud Use Case techniques. The Event Decomposition technique as the name implies is event driven and is initiated with noting events that require a business response, each event is then translated into a Use Case. The Event Decomposition technique categorises events into three broad categories of External events, Temporal Events and State Events. For each of these broad categories, as many specific events as can be listed are identified under each category which are refined while working closely with the user.

Seven phase make up the Event Decomposition Use Case definition technique which are: Identification of external events that require responses, mapping of use cases to identified external events, consideration of temporal events that require responses, mapping each temporal event to a use case, Consideration of state events that require response, mapping the state events to use case and finally refinement of all identified use cases.

4.4.4 CRUD Use case definition technique

The CRUD technique is often used in the context of database management (Burd, Jackson & Satzinger, 2004). With the CRUD, data types are modelled as data entities. The 4 steps which make up the CRUD use case technique are: Identification of all data entities, in the system, verification that use cases that perform database tasks for each data type have been identified, identification of stakeholders and where integrated applications apply, clear definition should be made of the application consuming the data and the application responsible for maintain the data (Burd, Jackson & Satzinger, 2004).

4.4.5 Misnomer's Eight Step SIEM Use Case Identification Technique

(Misnomer, 2012b) proposes an 8 step structured approach to SIEM use case development. With this technique organisational focus areas are identified which require security event monitoring. The identified focus areas are then mapped to use cases (Misnomer, 2012b). Whereas use case techniques outlined by Burd, Jackson & Satzinger (2004) are more generic and defined from a systems analysis and design perspective, (Misnomer, 2012b) Use Case development methodology is defined specifically for the SIEM Use Case context.

The 8 steps towards developing SIEM use cases as outlined by Misnomer (2012b) are:

Step 1- Requirements definition stage

At this step, specific Use Case requirements are outlined. These Use Case requirements are mapped from specific monitoring requirements aligned to business requirements.

Step 2 – Requirements Scope Definition

At the requirement scope definition step, IT infrastructure to be monitored is identified

Step 3 – Event Source Listing

The Event source listing stage looks at particular events and aligns these to SIEM Use Cases and ultimately to monitoring requirements.

Step 4 – Event Source Validation

The Event source validation step validates that identified event source infrastructure elements are capable of generating expected event logs that satisfy defined Use Cases.

Step 5 – Logic Definition

The Logic Definition phase is the stage relevant rules are defined and created to process incoming events

Step 6 – Implementation and Testing phase

As the phase title implies, log source are configured to generate event and the defined logic is implemented within SIEM followed by rigorous testing.

Step 7 – Use Case Response Definition

The previous 6 phases of this Use Case development methodology are incomplete without defining appropriate response in event that defined logic is triggered by events flowing into the SIEM.

Step 8 – SIEM on-going maintenance

The final phase in this SIEM use case development methodology involves on-going maintenance of the SIEM and continually refining the SIEM Use Case in order for them to remain relevant to new or existing security threats.

4.4.6 SIEM Use Case

A challenge organisations face in deploying SIEM for IT security is identifying use cases beyond the compliance SIEM use case (Lozito, 2011).

SIEM Use Cases can be defined as "a Logical, Actionable and Reportable component of an Event Management system (SIEM). It can be a Rule, Report, Alert or Dashboard which solves a set of needs or requirements" (Misnomer, 2012b). Observations from literature have been made that before deploying a SIEM, a critical consideration is to identify and define use cases for which the SIEM and associated events will be used (Chuvakin, 2008). In other words SIEM use cases are said to be one of the key starting points in the process of the deployment of SIEM in IT Security (Rothman, 2012). It can also said that one of the measures of the success in deploying a SIEM is the extent in which SIEM Use Cases have been clearly defined followed by how they meet organisational requirements (Misnomer, 2012a).

SIEM use case can be generic use cases applying across different organisations or specific to a single organisation. Ideally SIEM Use Case when defined should be very specific to allow for clearly defined expected outcome once they have been implemented (Chuvakin, 2008). Understanding the needs of an enterprise from both a security and business perspective is critical to the successful identification and implementation of SIEM use cases (Kelly, 2004).

Some of the more common SIEM use cases include:

- Insider abuse and unauthorised access detection and prevention
- Regulatory compliance satisfaction
- Forensic analysis and correlation
- Suspicious behaviour tracking
- User activity monitoring
- Application performance measurement

SIEM use cases were leveraged in Chapter 5 and Chapter 6 to explore the applicability of SIEM to IT security. Grimaila, Myers, Mills & Peterson (2011) followed a similar research approach when comparing distributed event correlation to centralised event correlation. Their research identified and selected 13 use cases based on the OWASP top 10 in evaluating distributed versus centralising logging to detect malicious behaviour (Grimaila, Myers, Mills & Peterson, 2011). Rotham (2012) suggests organisations actually building a “use case portfolio” even before selecting a SIEM.

While Grimaila, Myers, Mills & Peterson (2011) reference the OWASP top 10 in selecting use case, the approach use for this research was formulate a structure process for identifying SIEM use case to use for the experiment. The structured process was formulated using a combination of Misnomer’s (2012b) SIEM use case identification method and the Event Decomposition use case technique. Chapter 5 expounds further on this structured process and its relevance to the SIEM experiment.

5. SIEM in Information Security: Lab Setup

5.1 Introduction and Experimental Lab Rational

This Chapter presents the experimental exploration of the applicability of SIEM to IT security. Firstly an overview of the process that was followed in setting up the experimental lab is presented followed by an outline of the SIEM use cases that were identified for exploration in the lab. The Chapter ends with an overview of the limitations of the experimental lab.

Chapter 6 presents the actual practical exploration of SIEM in IT security through the application of the SIEM use cases outlined in this Chapter. Following this practical exploration, a discussion is presented on the observations and finally a discussion on the applicability of SIEM to IT security.

The two primary elements in undertaking this experiment were the SIEM lab and 4 SIEM use cases that had been identified for exploring the applicability of SIEM to IT security. From a research methodology perspective, this research could have been carried out by making use of the case study or the survey research methodology approach. Coppolino, D’Antonio, Formicola & Romano (2011) employs a case study approach in investigating SIEM deployment for monitoring Critical Infrastructure Protection (CIP). Survey based SIEM researches observed were mostly employed in the commercial world compared to academic world with this researcher observing few survey-based academic SIEM research. Observation by this researcher has however been of wide use of the experimental methodology in SIEM research as highlighted in section 1.2 of this paper.

As mentioned in the introduction in Chapter 1, the contribution of this research is to investigate, understand and add to the growing body of knowledge of SIEM in IT security. Specifically adding to the understanding of the where, how and if SIEM fits into the contemporary multi-dimensional approach to IT security. The use of an experimental approach was not only to assist in answering the research questions but also for this researcher to obtain a deeper practical understanding of SIEM by use of lab environment. Acquiring this foundational understanding was important for future research that this researcher would hope to undertake particularly when researching within the context of a real organisation SIEM environments. Because of the sensitivity of the data typically collected by SIEM deployed in organisations, it was not possible for this researcher to carry out the same tasks in a real live SIEM deployment and hence the choice of carrying out the experiment in a SIEM lab environment.

5.2 Experimental Lab Setup Process

The structured process in designing and setting up the experimental lab was carried out from the perspective of deploying a SIEM within an actual real organisation environment. The objective of doing so was to have a similar experience when undertaking the experiment to that of a real SIEM deployment. The setup of the lab was achieved through the following 4 phases:

1. Network design conceptualisation
2. SIEM Use Case identification
3. Infrastructure selection and setup
4. SIEM Use Case exploration

Figure 5.1 is a flow diagram of the lab design and setup process. Rectangular shapes in Figure 5.1 represent the five key setup phases while the parallelogram shapes define the main output from the phases. A description of each of the phases follows.

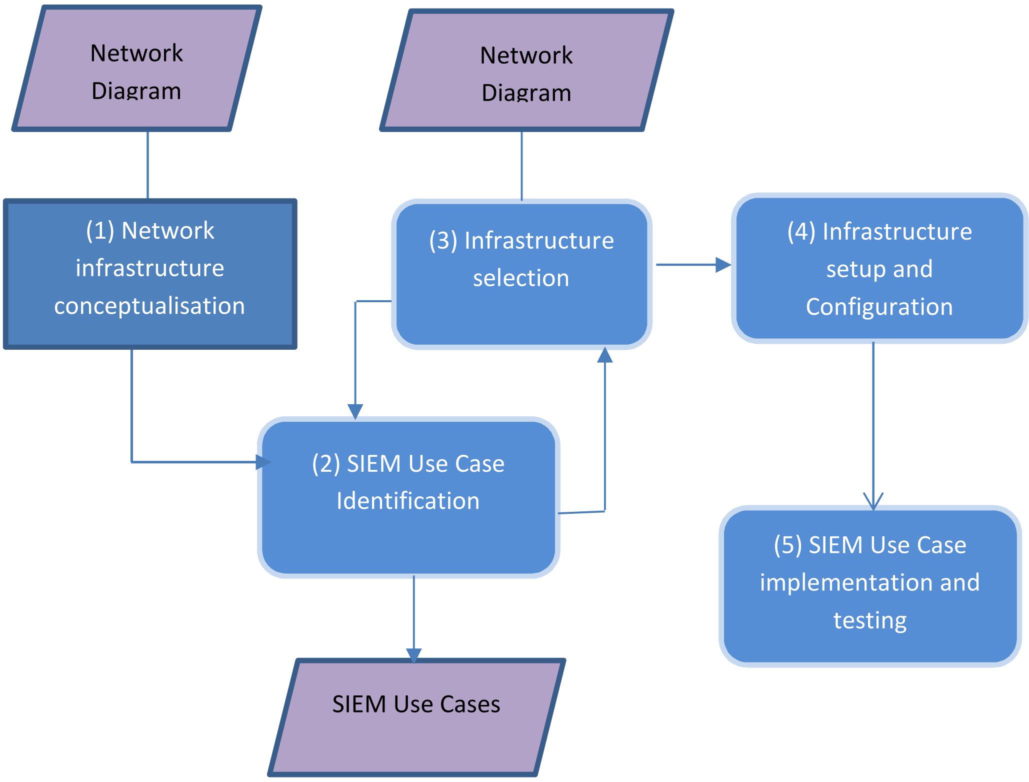

Network Diagram
Network Diagram
(1) Network infrastructure conceptualisation
(3) Infrastructure selection
(4) Infrastructure setup and Configuration
(2) SIEM Use Case Identification
(5) SIEM Use Case implementation and testing
SIEM Use Cases

Figure 5.1. Experimental lab setup process

Phase 1: Network design conceptualisation

This phase involved the conceptualisation of the experimental lab environment. The output from this phase was a high level network diagram showing the various lab components, relationship and the data flow between the components. The primary network components were:

- Log Sources
- The SIEM (Splunk)
- Physical networking infrastructure (cabling)
- Internet access

Log sources where the devices and applications on which events where generated and sent to the SIEM. The SIEM collected, parsed and stored the event logs. The physical networking provided the physical connectivity between the various network components and Internet access as expected provided connectivity to the untrusted Internet. Figure 5.2 was the output network diagram from this phase.

The lab network consisted of 4 zones:

- A trusted internal network
- A semi-trusted Demilitarized Zone 1 (DMZ1)
- A semi-trusted Demilitarized Zone 2 (DMZ2)
- An external facing Internet zone connecting up to the Internet service provider (ISP)

Within the trusted internal network user devices were connected to the network via a transparent wireless access point connecting up to the firewall. User devices in this zone included a windows 7 host, a windows XP virtual host running on the windows 7 host and mobile smartphones. The Firewall was central to network providing connectivity and routing between the 4 zones. The Firewall external interface (WAN1) connected up to the ISP router in the Internet zone providing access to the Internet. A physical windows server hosts and a physical Ubuntu based server were placed in the semi-trusted Demilitarized Zone 1. Within this zone was also a cisco switch and a cisco router all connected up to a 4 port switch and onto the firewall.

The server running the SIEM software Splunk was placed in the semi-trusted Demilitarized Zone 2 separated from all other hosts for security reasons. This was to ensure any compromise of hosts within the trusted internal network and semi-trusted Demilitarized Zone 1 did not affect the SIEM and events collected and stored by the SIEM.

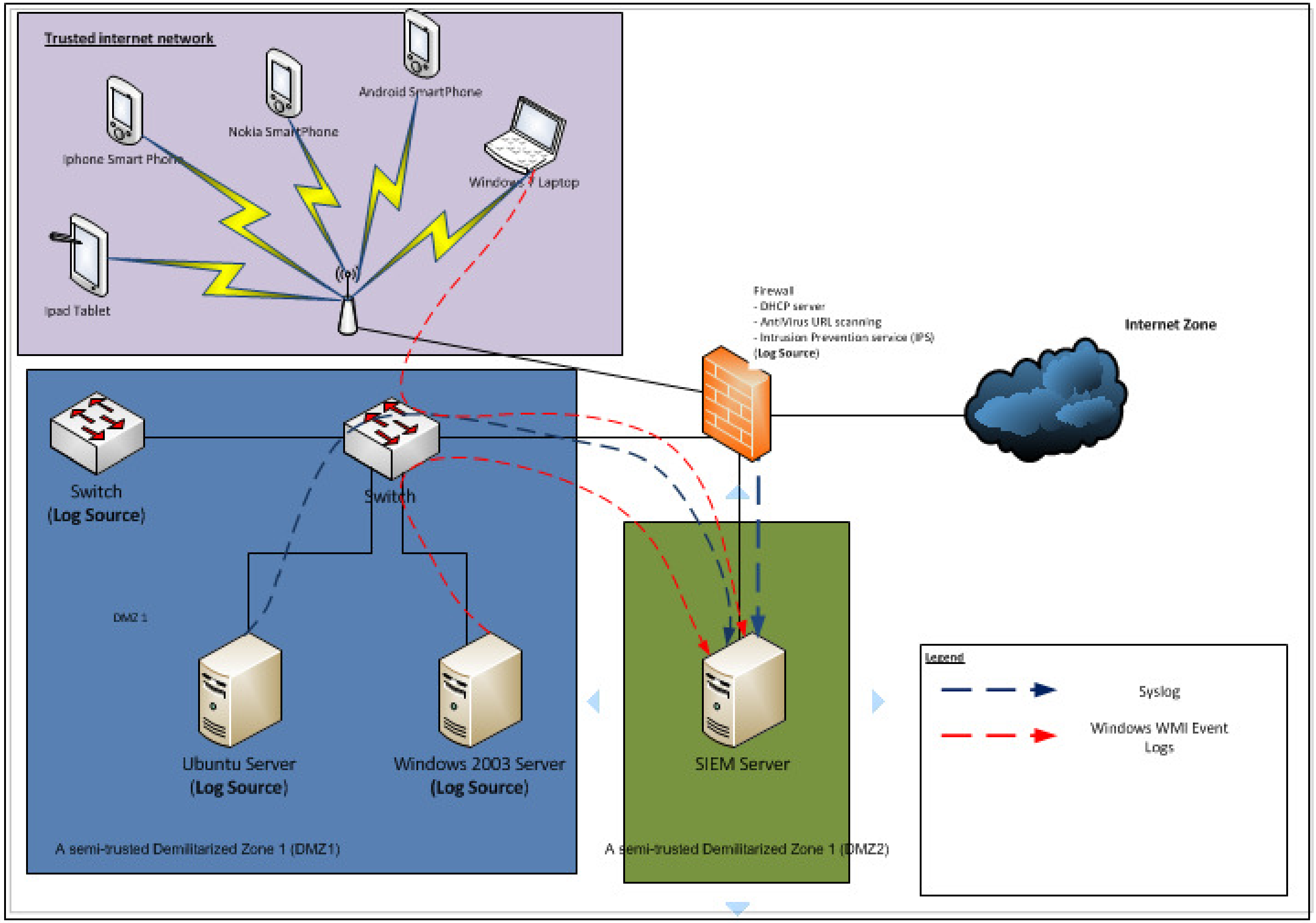


Figure 5.2. Experimental SIEM lab high level network diagram

Phase 2: SIEM Use Case identification

The second phase in the lab setup process was the identification of SIEM use cases. As introduced in section 4.4.6 of Chapter 4, identification of SIEM Use Cases was performed through a five step SIEM use case identification process specifically defined for the purpose of this experiment. The five step process was derived from a review of the Events Decomposition use case technique and the 8 step SIEM Use case development methodology suggested by (Misnomer, 2012b) both of which are outlined in Chapter 4.

The objective of defining a SIEM use case identification process was to enable an iterative and repetitive process that would allow for consistency and thoroughness in identifying and describing SIEM Use Case applied in the experiment. It was also to allow for a custom process particularly applicable to this experiment.

The output from phase 2 was a 5 step SIEM use case identification process illustrated in Figure 5.3 and four SIEM use cases presented in section 5.3. A description of each of the SIEM use case identification steps follows.

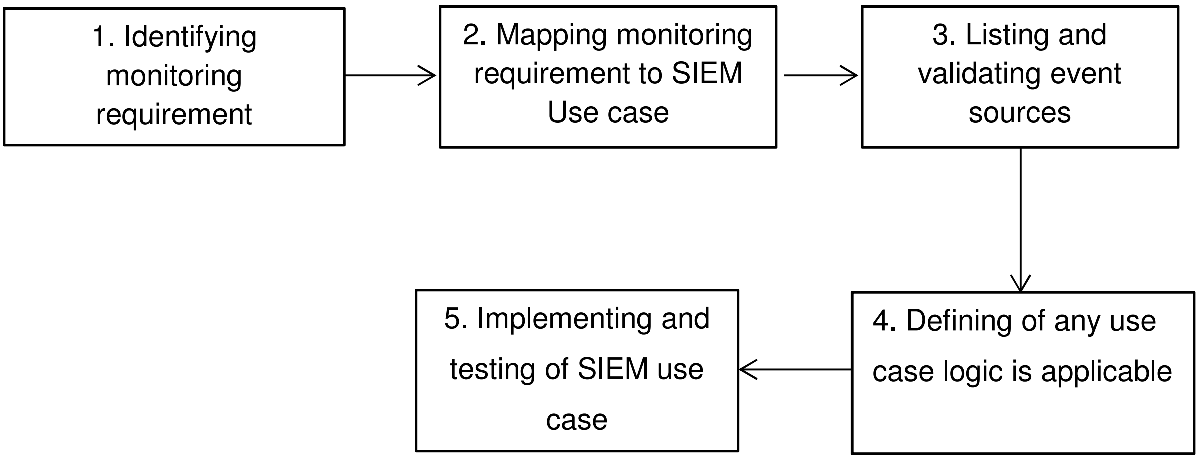

1. Identifying monitoring requirement
2. Mapping monitoring requirement to SIEM Use case
3. Listing and validating event sources
4. Defining of any use case logic is applicable
5. Implementing and testing of SIEM use case

Figure 5.3. SIEM use case identification process

SIEM use case identification step 1: ***Identification of monitoring requirement***

Within the context of a real world SIEM deployment, the monitoring requirement identified in step 1 is usually one that is mapped from a specific business monitoring requirement. A monitoring requirement can be seen as what the business or IT security requires to actively track, alert and report on using the SIEM. It is recommended that monitoring requirements when specified be as specific as possible. An example of a monitoring requirement that can be mapped from a business requirement is the need to monitor activity on specific or all privileged user accounts.

SIEM use case identification step 2: ***Mapping monitoring requirement to SIEM Use case***

On identification of a monitoring requirement, it was then mapped to a SIEM use case. The mapping of a monitoring requirement to a SIEM Use Case is designed to have the monitoring requirement described in terms that allow for implementation in the SIEM. Table 5.1 was employed as a template for describing the identified SIEM use cases. The template essentially allowed this researcher to easily describe identified SIEM use cases.

	Description
Monitoring requirement	*The monitoring requirement describes what is to be monitored*
Use Case title	*The Use Case title is the title of the Use Case*
Event Sources	*Event sources are a list of the event log sources applicable to the particular Use Case.*
Logic definition (if applicable)	*Logic definition defines any logic application to Use Case. The defined logic is typically translated into rules*
Implementation tasks	*There are the tasks followed in implementing the Use Case*

Table 5.1. SIEM use case description template

SIEM use case identification step 3: ***Listing and validating event sources***

At this step event sources for the SIEM use case were listed and validated. The purpose of validating the event sources was to ensure the applicability of the log sources to the SIEM use case.

SIEM use case identification step 4: ***Definition of use case logic if applicable***

Any processing logic applicable to a particular SIEM use case was defined in high level terms at this step. Not all SIEM use cases require processing logic to be defined. In other words, some SIEM use cases would not require additional correlation or logical processing while others would require specific logic to be defined.

SIEM use case identification step 5: ***Implementation and testing of the SIEM use case***

The final step in the process involved the actual experimenting using SIEM use cases on the data collected by the Splunk SIEM.

Phase 3: Infrastructure selection

In phase 3 of the experimental lab setup, the actual hardware and software infrastructure components required to implement SIEM use cases was sourced. Mostly free and open source software was employed in the lab due to ease of access, little or no cost and no licensing requirement.

Table 5.2 describes the specifications of the server log sources employed in the experimental lab.

	Windows server log source	**Linux server log source**	**Windows desktop log source**	**SIEM**
Operating system	Windows 2003 Server R2 SP2	Ubuntu linux server 12.04.3	Windows 7	Ubuntu linux server 12.04.3
CPU	Intel Celeron	AMD Sempron	Intel Pentium	Intel Pentium
Memory	512	1B	4GB	756MB
Disk Space	40GB	40GB	300GB	40GB
Purpose in the lab	Windows server log source	Linux server log source	Windows desktop log source	Server running the Splunk 6.0 SIEM
Server name	Mcis	mondemo	Mypc	usplk

Table 5.2. Log source specifications

In Table 5.3 are presented descriptions of the other log source components used in the experimental lab.

Log source	Description
FortiGate Firewall	The FortiGate firewall was configured as the gateway for the Internal DMZ 1 and DMZ 2 zones
Cisco Switch	Basic event logs including authentication and interface status logs were sent to the SIEM from the Cisco switch
Cisco Router	Basic event logs including authentication and interface status logs were sent to the SIEM from the Cisco router
Smart devices	3 smartphones and a tablet were connected wirelessly in the internal zone with access to the Internet. Firewall rules were configured with logging enabled and the firewall sending the syslog's to the Splunk SIEM

Table 5.3. Other log sources

The primary applications sending event logs to the SIEM were:

- MySQL database
- Squid proxy
- Dansguardian web filter,
- Webmin linux web management interface,
- ssh server,
- clamav,
- apache,
- fail2ban

<u>Select the Splunk SIEM</u>

In selection a SIEM for this experiment, three open source SIEM software were reviewed. These were:

- Alientvault OSSIM
- Cyberoam iView
- Splunk

Of the three, the Splunk SIEM was selected for its extensibility and the availability of a wide range of vendor and event specific log analysis add-ons. As expected at the commencement of this research, the experiment had not been fully outlined but it was critical to initiate the collection of event data in the SIEM with on-going analysis as the thesis developed. Splunk offered ease in terms basic log analysis of the events, easier manipulation of the events in the SIEM. It should however be noted that significant learning curve existed for more advanced Splunk features such as correlation.

The actual deployment of Splunk was based on the basic Splunk with direct network inputs deployment model (Splunk, 2009). The choice of this deployment model was that it allowed the researcher to easily add log sources as required. The specification of the server on which the Splunk SIEM was installed is outlined in Table 5.2.

Phase 4: Infrastructure setup and configuration

The infrastructure setup and configuration phase involved the installation and configuration of configurable applications and hardware.

Phase 5: SIEM Use Case Implementation and testing

Phase 5 was the actual implementation of SIEM use case as presented in Chapter 6.

5.3 SIEM Use Cases

In this section is presented the 4 SIEM use cases that were identified through the five step SIEM use case identification process outlined in section 5.2 of this Chapter. Each of the SIEM use cases is described using the template in Table 5.1. The background to each of the SIEM use cases is presented in section 6.2 of Chapter 6

5.3.1 SIEM Use Case 1: Outbound Traffic Threat Monitoring

Use Case Definition step	Description
Monitoring requirement	*Monitor for potential and real threats related to traffic destined for the Internet*
SIEM Use Case	*Outbound traffic threat monitoring*
Event Source(s)	*Event logs from FortiGate firewall* *Intrusion Prevention System logs from FortiGate firewall* *Squid logs from web proxy*
Logic definition	*N/A*
Implementation	• *Verify that firewall logs were being sent to the SIEM* • *Verify that squid logs were being sent to the SIEM* • *Verify that the key monitoring field of* ***destination IP address*** *is logged and therefore present in the event logs coming from both the Firewall and squid Log sources to the SIEM* • *Integrate SIEM with Honeyport.org IP reputation service* • *Generate or ensure relevant traffic is generated* • *Scan the logs traffic to potential or real malicious destination*

Table 5.4. Outbound traffic threat monitoring SIEM use case

5.3.2 SIEM Use Case 2: User Activity Monitoring

Use Case Definition step	Description
Monitoring requirement	*Monitor user activity (failed and successful)*
Use Case	*User activity monitoring*
Event Source(s)	*All log sources*
Logic definition (if applicable)	*N/A*
Implementation	• *Search for user activity related events* • *Extract user related data elements from relevant events if required* • *Normalise privileged event related logs for privileged user account monitoring if required* • *Monitor and report on user activity*

Table 5.5. User activity monitoring SIEM use case

5.3.3 SIEM Use Case 3: Compliance and Reporting SIEM Use Case

Use Case Definition step	Description
Monitoring requirement	*Satisfy relevant PCI DSS requirements*
Use Case	*Compliance and reporting*
Event Source(s)	*All log sources*
Logic definition (if applicable)	*Not applicable*
Implementation	*Follow audit approach in use case exploration*

Table 5.6. Compliance and Reporting SIEM Use Case

5.3.4 SIEM Use Case 4: Correlation SIEM Use Case

Use Case Definition step	Description
Monitoring requirement	*Correlation of traffic traversing the firewall with that web proxy traffic in order to identify potential threats*
Use Case	*Correlation for threat monitoring*
Event Source(s)	• *Traffic logs from Fortigate firewall* • *IPS logs from Fortigate firewall* • *Squid proxy logs*
Logic definition (if applicable)	
Implementation	

Table 5.7. Correlation SIEM use case

5.4 Experimental Lab Setup Limitations

A number of limitations were faced in setting up the experimental lab which to a certain degree had limited this researcher's scope for investigating the applicability of SIEM to IT security. This was however expected as noted the introduction to this Chapter that the experiment was designed to simulate a real-live SIEM deployment environment on a smaller scale. There were limitations generating events that would have allowed the researcher to have an extended SIEM use case list. It should however be noted that there was always going to be limitations in particular around relevant data that could be generated for the purpose of applying SIEM use cases. Other Limitations faced included:

- The use of open source software. Open source software while free and readily available can sometimes be difficult to install. The lack of support also meant that getting the software up and running was a challenge and limited the data that could be used for correlation. For instance the researcher attempted to install the modsecurity web application firewall in order to collect web application related security events. The data was to be used in the correlation SIEM use case for correlation web security threats with log events coming off the firewall. This was however ultimately not possible due to problems installing the software.

- The limited number of hosts generating relevant traffic.
- The SIEM learning curve. While the Splunk SIEM was the best available free SIEM according to this researcher when compared to other free SIEM reviewed, it must be noted a significant learning curve was experienced in becoming familiar with the software in order to effectively analyse the data that was collected by the Splunk SIEM. This did also limit the identification of SIEM use cases identified for the experiment.
- Limits in available threat related events. While the researcher was able to generate traffic to malicious destinations for the outbound threat monitoring SIEM use case, significant resources would have been required in order to purposely create a wider range of potential threats that could be analyse in the SIEM lab. Future work is however expected to consider and investigate how this limitation could be addressed in order to generate a richer data set for SIEM experimentation.

6. SIEM in IT Security: Lab Observations and Discussion

Chapter 6 presents the actual experiment exploring the applicability of SIEM to IT security through the use of SIEM use cases identified and briefly outlined in Chapter 5. The Chapter begins with an overview of the data collected. A detailed description of the experimental analysis follows. Finally a discussion on the applicability of SIEM to IT security is presented.

6.1 Data Collection

In section 5.2, log sources used in the experimental lab were presented. In section 5.3, specific log sources applicable to the actual experiment were outlined as part of defining the 4 SIEM use cases. It should be noted that even though only a subset of the log sources and associated events defined in section 5.2 was actually employed in the experiment it was important to get as much event logs into the SIEM as could be logged. It was observed that the configuration of as many log sources to send events to the SIEM not only avails to the information security professional relevant security events critical for security monitoring, but also importantly enables the security professional to potentially identify other security related SIEM use cases that might not have been identified had there been limitations on log sources sending events to the SIEM. This was also observed as a recommended practice in the deployment of SIEM where initially all relevant and irrelevant events are sent through the SIEM in spite of their perceived immediate use (Chuvakin, 2010). It was observed that over time as an understanding of the data collected by the SIEM is gained, filtering can then be applied with the main purpose of discarding particularly meaningless and irrelevant events. This application of filtering is known to improving the performance of the SIEM in terms of indexing and search response times. Within the context of this research's experimental lab, no particular filtering of events was however applied to event logs collected in the lab.

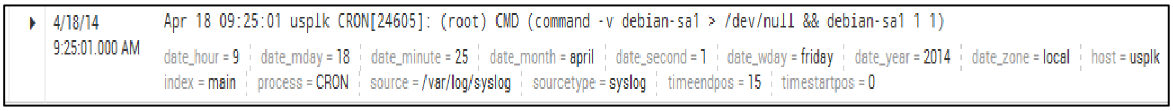
A total of 86,972,144 events were collected and stored by the lab Splunk SIEM. Logging to the SIEM was initiated at the commencement of this research. Post the identification of the 4 SIEM use cases, it was also realised that purpose event logs needed to be manually generated to enable experimentation through the identified SIEM use cases. The majority of events, representing 78%, were logged via the syslog protocol, a ubiquitous protocol supported by a wide range of vendor devices. The rest of the events were logged via Windows Management Instrumentation (WMI) and file based data sources.

Table 6.1 gives summary statistics of event logs collected listed by log source host. The data presented in Table 6.1 only shows event statistics from the main log sources.

Log Source	Number of events	% of total events	Log source description
mondemo	54,410,581	62.56%	Linux server
172.16.16.1	15,451,229	17.77%	Fortigate Firewall
mics	14,540,971	16.72%	Windows server
monw2k3	1,710,432	1.97%	Windows server
usplk	822,615	0.95%	SIEM (Linux server)
192.168.2.101	33,974	0.04%	Windows 7 host
172.16.16.21	2,407	0.00%	Cisco switch
172.16.16.20	18	0.00%	Cisco router
Other	18	0.00%	Other minor log sources
Total Events	**86,972,245**	**100.00%**	

Table 6.1. Summary of events collected by lab SIEM

Of the total number of events, about 62% of the events were logged from the Linux host named mondemo. Because no filtering was applied to the events coming through to the Splunk SIEM, over 50% of the events associated with the Linux server mondemo were deemed irrelevant. Figure 6.1 shows what could be defined as an irrelevant event collected by the SIEM. The event is a cron related task irrelevant for security monitoring in this context. It should be however noted that what is deemed irrelevant in one SIEM environment might be seen as relevant in another environment.



```
▸ 4/18/14          Apr 18 09:25:01 usplk CRON[24605]: (root) CMD (command -v debian-sa1 > /dev/null && debian-sa1 1 1)
  9:25:01.000 AM   date_hour = 9 | date_mday = 18 | date_minute = 25 | date_month = april | date_second = 1 | date_wday = friday | date_year = 2014 | date_zone = local | host = usplk
                   index = main | process = CRON | source = /var/log/syslog | sourcetype = syslog | timeendpos = 15 | timestartpos = 0
```

Figure 6.1. Sample “irrelevant” event

Table A1.1 in Appendix 1, shows the variety of log sources types identified in the lab SIEM. Events sent from the Linux hosts were mostly system type events. All events coming from the windows server were collected from the windows application, security and systems event logs. Events logged from the firewall log source were from two main types of sources:

- General traffic traversing the firewall between the trusted internal network and the Internet, the trusted internal network and the DMZs. between the DMZs and between the DMZs and the Internet.
- Intrusion Prevention System (IPS) events trigged by real or potential anomalies detections seen in traffic traversing the firewall between the various Zones.

Other log sources sending events to the Splunk SIEM included a cisco switch, a cisco router and the researchers Windows 7 host.

Events referenced in the experiment

The variety in the type of log source types collected by the SIEM was observed to illustrate the potential for a significant number of other SIEM use cases that could have been outlined beyond the 4 explored in this paper. Time and resource limitations however meant that only a limited number of security related SIEM use cases could be explored.

Of the events collected by the lab Splunk SIEM, the following event types were actually referenced in this experiment:

- Squid events generated from web traffic monitoring
- Windows events
- Cisco switch and router events
- Firewall event traffic
- IPS events

6.2 SIEM Experimentation Leveraging SIEM Use Cases

As presented in Chapter 4, section 4.4.6 and Chapter 5, the applicability of SIEM in IT security was explored through four SIEM use case defined and outlined in section 5.3 of Chapter 5. The four SIEM use case were:

1. Outbound threat monitoring SIEM use case
2. User activity monitoring SIEM use case
3. Compliance and reporting SIEM use case
4. Correlation SIEM use case

The next sections outline how these use case were experimentally applied in exploring SIEM in IT security.

6.2.1 Outbound Traffic Threat Monitoring SIEM Use Case

6.2.1.1 Background to SIEM Use Case

The security of networks has traditionally focused on protecting the trusted internal network from security threats mostly originating from the untrusted Internet network. These external threats could be Denial of Service (Dos) attacks, attempts to compromise webservers, attacks on hardware or software vulnerabilities, phishing attacks delivered through spam email or any of the many security threats targeting the trusted internal network. The protection of the trusted internal network from external threats has been through the deployment of perimeter defences such as firewalls, email filtering gateways, architectural based defences such as DMZs and most recently web application firewalls (WAFs) deployed to protect web application from exploits targeting web application vulnerabilities.

As discussed in this paper, the evolving nature of IT security threats has led to a realisation of the need of a multi-dimensional approach to security. Active and regular event log based monitoring of traffic leaving the trusted internal network to the untrusted Internet has become an important daily task in this multi-dimensional approach to IT security. Information security professionals analysing these events logs using any of the log analysis techniques presented in section 4.4 of Chapter 4, would typically be on the lookout for suspicious or malicious traffic that could be for instance traffic coming from a compromised internal host and trying to communicate back to a malicious external host such as a botnet’s command and control centre.

Traffic leaving the trusted internal network has mostly been controlled and monitored through web proxies and email gateways. With web proxies, rules are defined which allow or block explicitly or transparently redirected web traffic going through the proxy. The rules usually restrict access based on some defined organisational security policy. Blocking can be based on websites category, file type or simply blocking of content that is considered a threat to the internal network. Email gateways on the other hand monitor email traffic leaving or entering the trusted network, blocking any email that is deemed to contain or potentially contain malicious content. Intrusion Prevention Systems have also remained an important part of the multi-dimensional approach to IT security, employing a signature based approach to monitoring traffic entering or leaving the trusted internal network. The approach used in the just mentioned devices is more of a configure-and-wait approach, in that rules or signatures are applied and the information security professional responds reactively when notifications on suspicious activity are raised from any of the security devices.

The expectation with SIEM however is not that of a configure-and-wait for warning notifications approach, which explains the log analysis techniques highlighted in Chapter 4.

The focus of exploring SIEM through this use case was to investigate the applicability of SIEM in IT security from an outbound threat monitoring perspective.

6.2.1.2 Exploring use case

The exploration of SIEM through this use case leveraged threat intelligence based on IP reputation and associated IP threat scores. IP reputation in general is concept in which IP addresses are assigned a reputation score or risk rating derived from the real-time and historical analysis of traffic patterns to or from an IP addresses (Sadan, 2012). Traffic history used to determine the risk rating of any particular IP address, could be based on email sending patterns, or suspicious traffic patterns such as botnet behaviour. Essentially a reputation service will, through analysing patterns and aggregating feedback from participants, develop over a period of time what can be described as threatscores or risk ratings associated with each IP address in its database. These threatscores are considered representative of the trustworthiness or reputation of any particular IP address in the database (Sadan, 2012). In practical applications of an IP reputation service, traffic to or from destinations with a high threatscores or high risk rating can be monitored or even blocked in order to reduce or eliminate the threat to hosts on the trusted network. This is typical in the event of a host or hosts on the trusted network attempting to communicate with a host marked by the IP reputation service as having a high risk rating.

The relevancy of employing IP reputation for this SIEM use case can be noted from the research by Sadan (2012) where the researcher investigates the enhancement of IP reputation based services which are today used by a significant number of technologies for threat intelligence. Examples of technologies that employ IP reputation based threat intelligence including anti-spam services, firewall services with integrated threat intelligence, and reputation based web filtering. IP reputation services are widely employed by organisations either using a commercial or non-commercial service.

The exploration of SIEM using this SIEM use case employed the free IP reputation service provided by the Project Honey Pot database via DNS-Blacklist requests. The Project Honey Pot database provides blacklist lookups which can be used to monitor or block traffic to IP destinations based on their reputation. The identification and exploration of an *Outbound traffic threat monitoring* SIEM use case was targeted at exploring the role of SIEM in IT security for monitoring threats in outbound traffic.

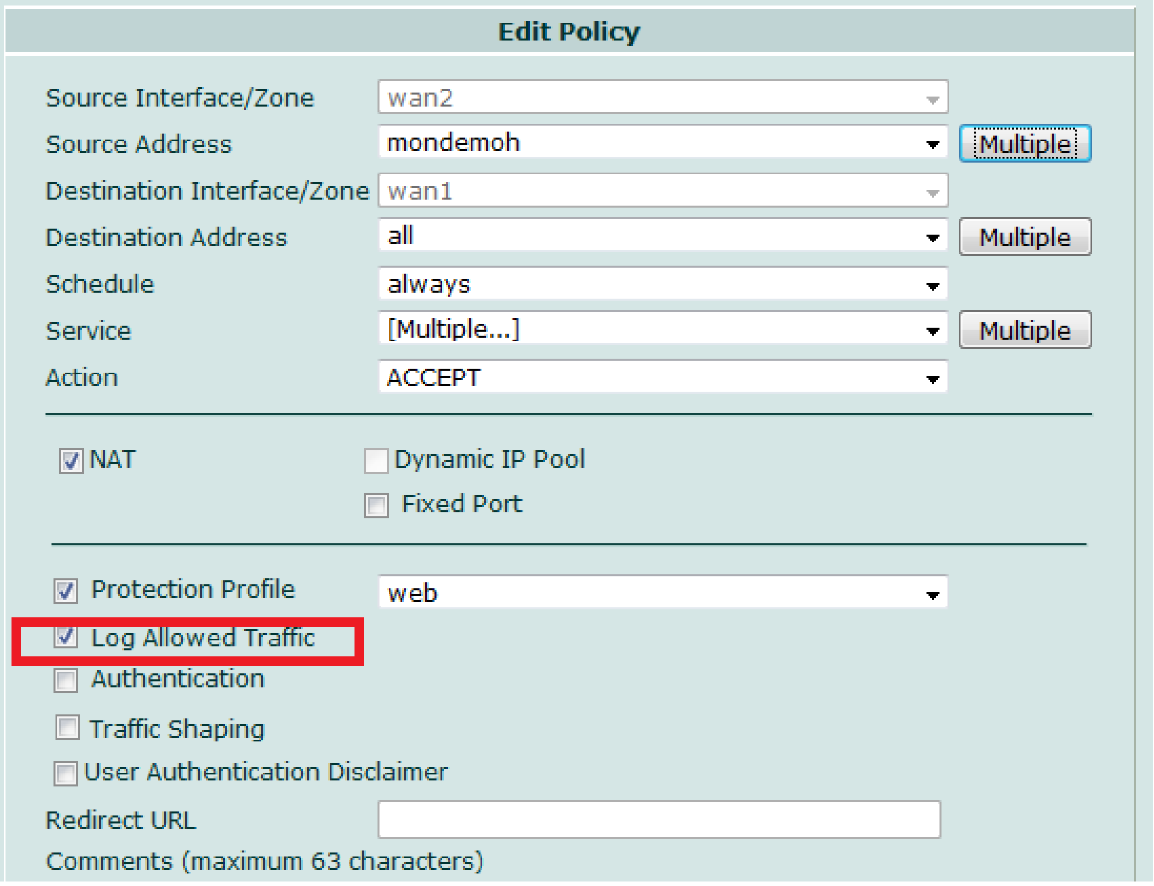
Event logs from the FortiGate firewall and a squid proxy were employed in the investigating using this SIEM Use Case.

The tasks performed in this exploration were:

- Verification that firewall logs were sent to the SIEM
- Verification that squid logs were sent to the SIEM
- Verification that the key monitoring field of ***destination IP address*** appeared in event logs coming from both the Firewall and squid Log sources to the SIEM
- Integration of the SIEM with the Honeyport.org IP reputation service
- Manual generation of relevant traffic
- Analysing the event logs for high risk destinations based on threat score.

Verification that firewall logs were being sent to the SIEM

FortiGate firewall rules were configured to log traffic traversing the firewall to the Internet. The FortiGate firewall was then configured to send these logs to the Splunk SIEM via the syslog protocol on User Datagram protocol (UDP) port 514. Verification that the logs were stored by the SIEM was performed by logging onto the SIEM and filtering for logs coming from the firewall log source IP address of 172.16.16.1. Figure 6.2 shows a firewall rule configured to log any traffic matching the firewall rule. The example in Figure 6.2 is that of traffic traversing the firewall to the Internet and coming from clients whose traffic had been redirected through the squid proxy.



Edit Policy

Source Interface/Zone: wan2

Source Address: mondemoh [Multiple]

Destination Interface/Zone: wan1

Destination Address: all [Multiple]

Schedule: always

Service: [Multiple...] [Multiple]

Action: ACCEPT

☑ NAT

□ Dynamic IP Pool

□ Fixed Port

☑ Protection Profile: web

☑ Log Allowed Traffic

□ Authentication

□ Traffic Shaping

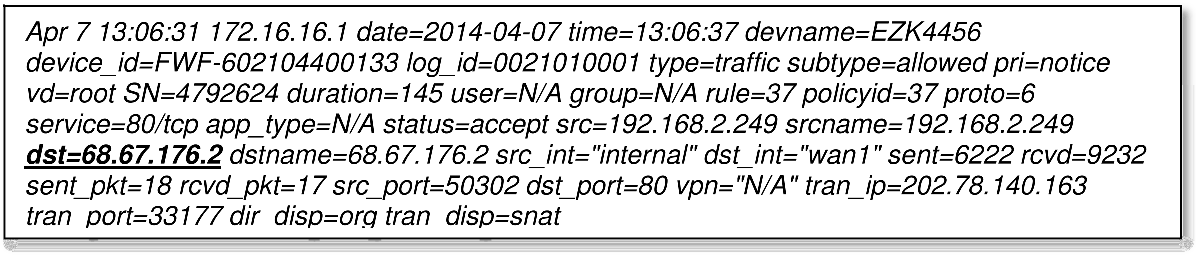
□ User Authentication Disclaimer

Redirect URL:

Comments (maximum 63 characters)

Figure 6.2. Sample firewall rule configured to log events matching the rule

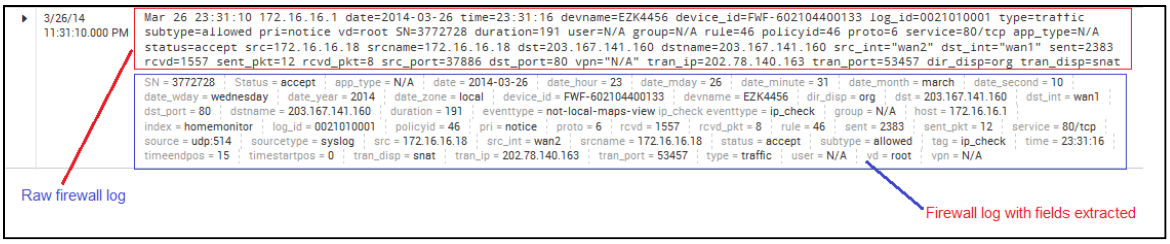
Figure 6.3 is a sample raw FortiGate firewall raw event log with the destination IP address data element highlighted.



```
Apr 7 13:06:31 172.16.16.1 date=2014-04-07 time=13:06:37 devname=EZK4456
device_id=FWF-602104400133 log_id=0021010001 type=traffic subtype=allowed pri=notice
vd=root SN=4792624 duration=145 user=N/A group=N/A rule=37 policyid=37 proto=6
service=80/tcp app_type=N/A status=accept src=192.168.2.249 srcname=192.168.2.249
dst=68.67.176.2 dstname=68.67.176.2 src_int="internal" dst_int="wan1" sent=6222 rcvd=9232
sent_pkt=18 rcvd_pkt=17 src_port=50302 dst_port=80 vpn="N/A" tran_ip=202.78.140.163
tran_port=33177 dir_disp=org tran_disp=snat
```

Figure 6.3. Sample raw event from the firewall

Figure 6.4 is a sample of an event log sent from the FortiGate firewall as collected by the SIEM.



3/26/14
11:31:10.000 PM

```
Mar 26 23:31:10 172.16.16.1 date=2014-03-26 time=23:31:16 devname=EZK4456 device_id=FWF-602104400133 log_id=0021010001 type=traffic
subtype=allowed pri=notice vd=root SN=3772728 duration=191 user=N/A group=N/A rule=46 policyid=46 proto=6 service=80/tcp app_type=N/A
status=accept src=172.16.16.18 srcname=172.16.16.18 dst=203.167.141.160 dstname=203.167.141.160 src_int="wan2" dst_int="wan1" sent=2383
rcvd=1557 sent_pkt=12 rcvd_pkt=8 src_port=37886 dst_port=80 vpn="N/A" tran_ip=202.78.140.163 tran_port=53457 dir_disp=org tran_disp=snat
```

```
SN = 3772728 | Status = accept | app_type = N/A | date = 2014-03-26 | date_hour = 23 | date_mday = 26 | date_minute = 31 | date_month = march | date_second = 10 |
date_wday = wednesday | date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | dir_disp = org | dst = 203.167.141.160 | dst_int = wan1 |
dst_port = 80 | dstname = 203.167.141.160 | duration = 191 | eventtype = not-local-maps-view ip_check eventtype = ip_check | group = N/A | host = 172.16.16.1 |
index = homemonitor | log_id = 0021010001 | policyid = 46 | pri = notice | proto = 6 | rcvd = 1557 | rcvd_pkt = 8 | rule = 46 | sent = 2383 | sent_pkt = 12 | service = 80/tcp |
source = udp:514 | sourcetype = syslog | src = 172.16.16.18 | src_int = wan2 | srcname = 172.16.16.18 | status = accept | subtype = allowed | tag = ip_check | time = 23:31:16 |
timeendpos = 15 | timestartpos = 0 | tran_disp = snat | tran_ip = 202.78.140.163 | tran_port = 53457 | type = traffic | user = N/A | vd = root | vpn = N/A
```

Figure 6.4. Firewall event log as appears in SIEM

Of note with the firewall event logs stored in the SIEM was that the destination IP address data element required for this SIEM use case was parsed and stored in the "***dst***" field

<u>*Verification that Squid web traffic logs were being sent to SIEM*</u>

The squid proxy was installed on the Linux server mondemo for the purpose of monitoring Internet bound web traffic coming from clients on the network. It was important to ensure that the raw squid proxy event logs also contained the destination IP addresses data element in all traffic logged by the proxy.

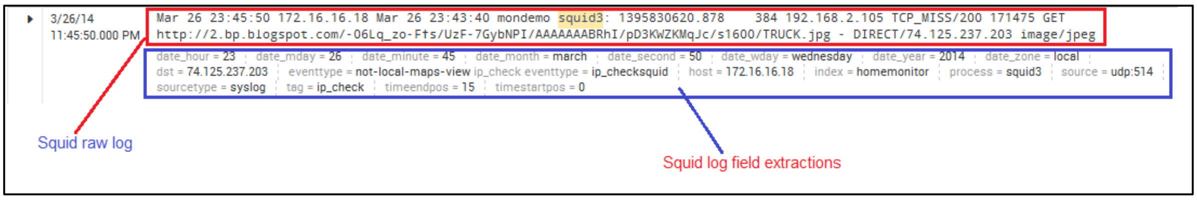
Other data elements which though not relevant for this particular SIEM use case are expected to be logged by web proxies include the source IP address of clients browsing the Internet through the squid proxy, the actual URL visited by a client and the date and time a particular URL was accessed. These data elements are relevant not only for reporting on client web browsing behaviour but can be correlated with other SIEM events in forensic investigations.

Figure 6.5 is a sample raw log from the squid proxy with key data elements in the event log highlighted. Underlined are some of the important data elements in this raw log.

```
Apr  7 07:26:16 mondemo squid3: 1396812376.426    179 192.168.2.105 TCP_MISS/200 8495 GET
http://i.stuff.co.nz/sport/football/9909645/Wellington-Phoenix-close-in-on-two-EPL-giants -
DIRECT/202.21.128.102 text/html
```

Figure 6.5. Raw squid event log

The squid web proxy was also configured to send logs via the syslog protocol to the SIEM. Figure 6.6 is a sample Squid log as stored in the SIEM.

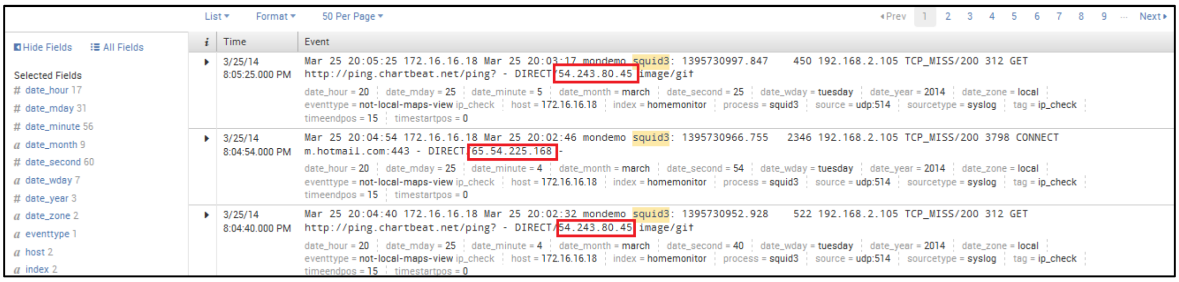


```
3/26/14           Mar 26 23:45:50 172.16.16.18 Mar 26 23:43:40 mondemo squid3: 1395830620.878    384 192.168.2.105 TCP_MISS/200 171475 GET
11:45:50.000 PM   http://2.bp.blogspot.com/-06Lq_zo-Fts/UzF-7GybNPI/AAAAAAABRhI/pD3KWZKMqJc/s1600/TRUCK.jpg - DIRECT/74.125.237.203 image/jpeg
                  date_hour = 23 | date_mday = 26 | date_minute = 45 | date_month = march | date_second = 50 | date_wday = wednesday | date_year = 2014 | date_zone = local |
                  dst = 74.125.237.203 | eventtype = not-local-maps-view ip_check eventtype = ip_checksquid | host = 172.16.16.18 | index = homemonitor | process = squid3 | source = udp:514 |
                  sourcetype = syslog | tag = ip_check | timeendpos = 15 | timestartpos = 0
```

Figure 6.6. Sample squid event log as seen in the SIEM

Verified that the key monitoring field of the destination IP address was showing in the logs

As the destination IP address was the key data element in the exploration using this SIEM use case, it was critical that this data element appeared in the event logs. The destination IP address was required to be available in both the squid proxy and Firewall logs. It was observed that the destination IP address in the firewall event logs was natively parsed and extracted by the SIEM into a destination IP address field labelled as "***dst***". The destination IP address in the squid web proxy log was however not natively parsed and extracted by the SIEM as shown in Figure 6.7. In order for the successful monitoring of outbound traffic using the SIEM, this data element had to be available for logical process.

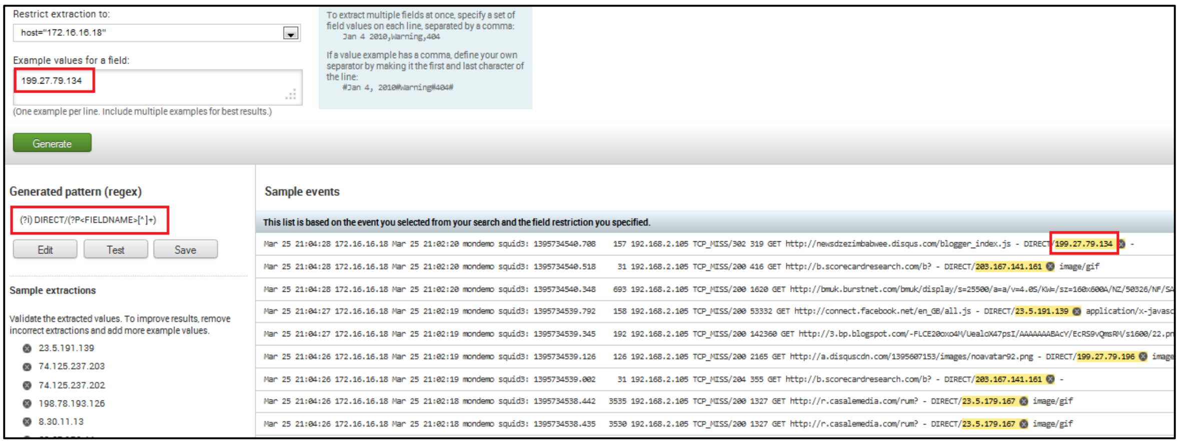


```
Selected Fields
# date_hour 17
# date_mday 31
# date_minute 56
a date_month 9
# date_second 60
a date_wday 7
# date_year 3
a date_zone 2
a eventtype 1
a host 2
a index 2
```

```
i  Time             Event
▸  3/25/14          Mar 25 20:05:25 172.16.16.18 Mar 25 20:03:17 mondemo squid3: 1395730997.847    450 192.168.2.105 TCP_MISS/200 312 GET
   8:05:25.000 PM   http://ping.chartbeat.net/ping? - DIRECT/54.243.80.45 image/gif
                    date_hour = 20 | date_mday = 25 | date_minute = 5 | date_month = march | date_second = 25 | date_wday = tuesday | date_year = 2014 | date_zone = local |
                    eventtype = not-local-maps-view ip_check | host = 172.16.16.18 | index = homemonitor | process = squid3 | source = udp:514 | sourcetype = syslog | tag = ip_check |
                    timeendpos = 15 | timestartpos = 0
▸  3/25/14          Mar 25 20:04:54 172.16.16.18 Mar 25 20:02:46 mondemo squid3: 1395730966.755    2346 192.168.2.105 TCP_MISS/200 3798 CONNECT
   8:04:54.000 PM   m.hotmail.com:443 - DIRECT/65.54.225.168 -
                    date_hour = 20 | date_mday = 25 | date_minute = 4 | date_month = march | date_second = 54 | date_wday = tuesday | date_year = 2014 | date_zone = local |
                    eventtype = not-local-maps-view ip_check | host = 172.16.16.18 | index = homemonitor | process = squid3 | source = udp:514 | sourcetype = syslog | tag = ip_check |
                    timeendpos = 15 | timestartpos = 0
▸  3/25/14          Mar 25 20:04:40 172.16.16.18 Mar 25 20:02:32 mondemo squid3: 1395730952.928    522 192.168.2.105 TCP_MISS/200 312 GET
   8:04:40.000 PM   http://ping.chartbeat.net/ping? - DIRECT/54.243.80.45 image/gif
                    date_hour = 20 | date_mday = 25 | date_minute = 4 | date_month = march | date_second = 40 | date_wday = tuesday | date_year = 2014 | date_zone = local |
                    eventtype = not-local-maps-view ip_check | host = 172.16.16.18 | index = homemonitor | process = squid3 | source = udp:514 | sourcetype = syslog | tag = ip_check |
                    timeendpos = 15 | timestartpos = 0
```

Figure 6.7. Destination IP address not parsed by SIEM

A field extraction process as illustrated in Figure 6.8 and Figure 6.9, was performed to manually extract the destination IP address data element in a relevant field, which in this case was the "***dst***" labelled field consistent with that of the firewall logs.



Restrict extraction to:

host="172.16.16.18"

Example values for a field:

199.27.79.134

(One example per line. Include multiple examples for best results.)

To extract multiple fields at once, specify a set of field values on each line, separated by a comma:

```
Jan 4 2010,Warning,404
```

If a value example has a comma, define your own separator by making it the first and last character of the line:

```
#Jan 4, 2010#Warning#404#
```

Generated pattern (regex)

```
(?i) DIRECT/(?P<FIELDNAME>[^ ]+)
```

Sample extractions

Validate the extracted values. To improve results, remove incorrect extractions and add more example values.

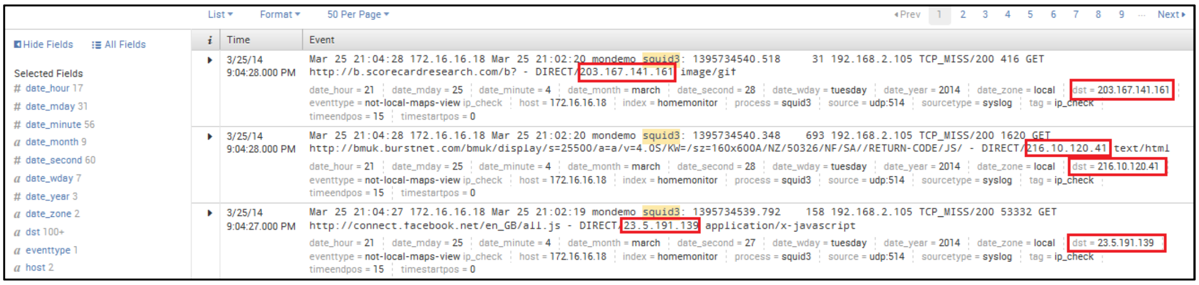
- 23.5.191.139
- 74.125.237.203
- 74.125.237.202
- 198.78.193.126
- 8.30.11.13

Sample events

This list is based on the event you selected from your search and the field restriction you specified.

```
Mar 25 21:04:28 172.16.16.18 Mar 25 21:02:20 mondemo squid3: 1395734540.708   157 192.168.2.105 TCP_MISS/302 319 GET http://newsdzezimbabwee.disqus.com/blogger_index.js - DIRECT/199.27.79.134 -
Mar 25 21:04:28 172.16.16.18 Mar 25 21:02:20 mondemo squid3: 1395734540.518    31 192.168.2.105 TCP_MISS/200 416 GET http://b.scorecardresearch.com/b? - DIRECT/203.167.141.161 image/gif
Mar 25 21:04:28 172.16.16.18 Mar 25 21:02:20 mondemo squid3: 1395734540.348   693 192.168.2.105 TCP_MISS/200 1620 GET http://bmuk.burstnet.com/bmuk/display/s=25500/a=a/v=4.0S/KW=/sz=160x600A/NZ/50326/NF/SA
Mar 25 21:04:27 172.16.16.18 Mar 25 21:02:19 mondemo squid3: 1395734539.792   158 192.168.2.105 TCP_MISS/200 53332 GET http://connect.facebook.net/en_GB/all.js - DIRECT/23.5.191.139 application/x-javasc
Mar 25 21:04:27 172.16.16.18 Mar 25 21:02:19 mondemo squid3: 1395734539.345   192 192.168.2.105 TCP_MISS/200 142360 GET http://3.bp.blogspot.com/-FLCE20oxo4M/UealoX47psI/AAAAAAABAcY/EcRS9vQmsRM/s1600/22.pn
Mar 25 21:04:26 172.16.16.18 Mar 25 21:02:19 mondemo squid3: 1395734539.126   126 192.168.2.105 TCP_MISS/200 2165 GET http://a.disquscdn.com/1395607153/images/noavatar92.png - DIRECT/199.27.79.196 image
Mar 25 21:04:26 172.16.16.18 Mar 25 21:02:19 mondemo squid3: 1395734539.002    31 192.168.2.105 TCP_MISS/204 355 GET http://b.scorecardresearch.com/b? - DIRECT/203.167.141.161 -
Mar 25 21:04:26 172.16.16.18 Mar 25 21:02:18 mondemo squid3: 1395734538.442  3535 192.168.2.105 TCP_MISS/200 1327 GET http://r.casalemedia.com/rum? - DIRECT/23.5.179.167 image/gif
Mar 25 21:04:26 172.16.16.18 Mar 25 21:02:18 mondemo squid3: 1395734538.435  3530 192.168.2.105 TCP_MISS/200 1327 GET http://r.casalemedia.com/rum? - DIRECT/23.5.179.167 image/gif
```

Figure 6.8. Extraction of destination IP address in squid event message



Selected Fields

- # date_hour 17
- # date_mday 31
- # date_minute 56
- a date_month 9
- # date_second 60
- a date_wday 7
- # date_year 3
- a date_zone 2
- a dst 100+
- a eventtype 1
- a host 2

i	Time	Event
▸	3/25/14 9:04:28.000 PM	Mar 25 21:04:28 172.16.16.18 Mar 25 21:02:20 mondemo squid3: 1395734540.518 31 192.168.2.105 TCP_MISS/200 416 GET http://b.scorecardresearch.com/b? - DIRECT/203.167.141.161 image/gif date_hour = 21 ¦ date_mday = 25 ¦ date_minute = 4 ¦ date_month = march ¦ date_second = 28 ¦ date_wday = tuesday ¦ date_year = 2014 ¦ date_zone = local ¦ dst = 203.167.141.161 ¦ eventtype = not-local-maps-view ip_check ¦ host = 172.16.16.18 ¦ index = homemonitor ¦ process = squid3 ¦ source = udp:514 ¦ sourcetype = syslog ¦ tag = ip_check ¦ timeendpos = 15 ¦ timestartpos = 0
▸	3/25/14 9:04:28.000 PM	Mar 25 21:04:28 172.16.16.18 Mar 25 21:02:20 mondemo squid3: 1395734540.348 693 192.168.2.105 TCP_MISS/200 1620 GET http://bmuk.burstnet.com/bmuk/display/s=25500/a=a/v=4.0S/KW=/sz=160x600A/NZ/50326/NF/SA//RETURN-CODE/JS/ - DIRECT/216.10.120.41 text/html date_hour = 21 ¦ date_mday = 25 ¦ date_minute = 4 ¦ date_month = march ¦ date_second = 28 ¦ date_wday = tuesday ¦ date_year = 2014 ¦ date_zone = local ¦ dst = 216.10.120.41 ¦ eventtype = not-local-maps-view ip_check ¦ host = 172.16.16.18 ¦ index = homemonitor ¦ process = squid3 ¦ source = udp:514 ¦ sourcetype = syslog ¦ tag = ip_check ¦ timeendpos = 15 ¦ timestartpos = 0
▸	3/25/14 9:04:27.000 PM	Mar 25 21:04:27 172.16.16.18 Mar 25 21:02:19 mondemo squid3: 1395734539.792 158 192.168.2.105 TCP_MISS/200 53332 GET http://connect.facebook.net/en_GB/all.js - DIRECT/23.5.191.139 application/x-javascript date_hour = 21 ¦ date_mday = 25 ¦ date_minute = 4 ¦ date_month = march ¦ date_second = 27 ¦ date_wday = tuesday ¦ date_year = 2014 ¦ date_zone = local ¦ dst = 23.5.191.139 ¦ eventtype = not-local-maps-view ip_check ¦ host = 172.16.16.18 ¦ index = homemonitor ¦ process = squid3 ¦ source = udp:514 ¦ sourcetype = syslog ¦ tag = ip_check ¦ timeendpos = 15 ¦ timestartpos = 0

Figure 6.9. Extracted of destination IP address in squid event message

Integrating SIEM with Honeyport.org IP reputation service

Analysis of outbound traffic for potential communication with a high risk Internet hosts was achieved through a Splunk application called IP reputation (Splunk, 2014). The use of the app was to enable the analysis of outbound traffic based on IP threat scoring or risk rating. The threat scoring or risk rating is a value describing the risk of an IP address based its observed historical activity (Project Honeypot, unknown), activity that could be spam sending patterns, hosting of malicious websites or performing suspicious behaviour. Table 6.2 shows the logarithmic interpretation of the threat rating based on the equivalent of sending spam (Project Honeypot, unknown).

Threat Rating	IP is seen as one that has sent:
25	100 spam messages
50	10,000 spam messages
75	1,000,000 spam messages

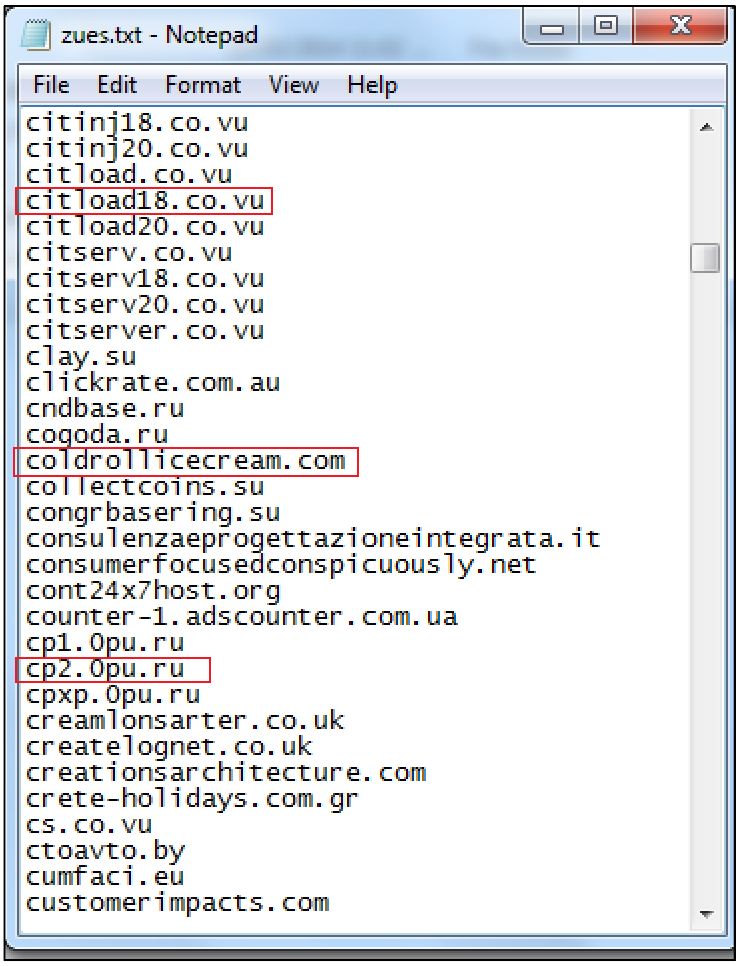
Table 6.2. Logathirmic threat scoring

For the service to be integrated with Splunk, a free account was setup with the project honeypot to obtain a key an *http:BL Authorization Key* which was when configured on the SIEM to allow for IP reputation lookups.

One of the problems observed when working with the IP reputation app was that the underlying search logic was designed for a set of data elements that were different to those available in the events collected in this lab. For instance the reference destination IP address field in the application was different from the destination field in the events in the SIEM lab, For this researcher, it highlighted the challenge that has been observed in literature and past research regarding the lack of event format standards (discussed in Chapter 3). The IP reputation application was designed for a particular data set, expecting specific event log data elements which in this case were not present in the event logs collected for this experiment. For the logic behind the threatscore determination to therefore work, it was critical to customise the underlying IP reputation logic in order to match the destination IP address data element as it was in this SIEM lab.

Generating traffic

While the firewall was constantly monitoring traffic from internal devices to the Internet and sending traffic logs to the SIEM, most of this traffic was observed to be mostly to "safe" destinations. There was therefore a need to purposely generate outbound traffic to potentially malicious destinations in order get the SIEM to consume events destined for malicious destinations and not only events to "safe" destinations. This was achieved through an automated web browsing tool called Webtimer capable of automatically browsing websites based on a list of domains fed to the tool from a text file. Figure 6.10 is a sample of the text file that was used. As illustrated the text file contain a list of domains that were referenced by the Webtimer tool.



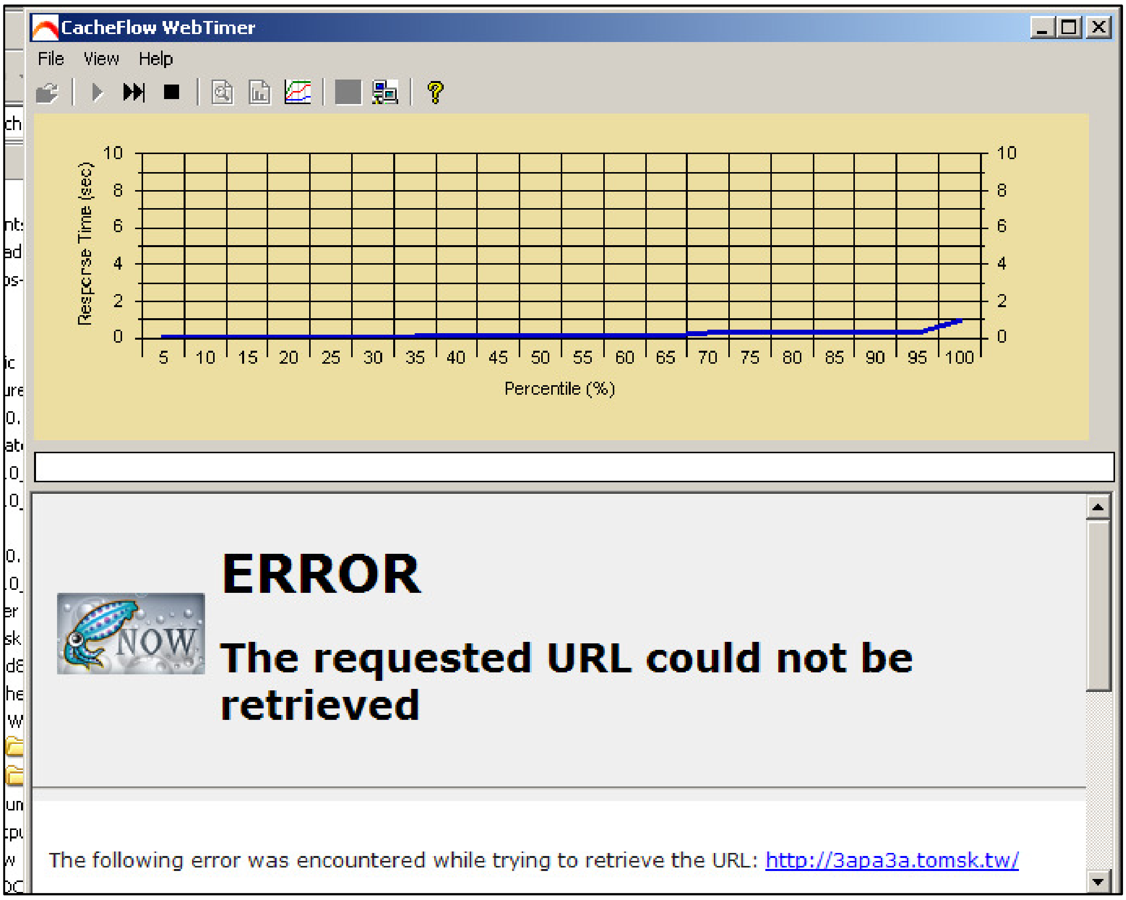
```
citinj18.co.vu
citinj20.co.vu
citload.co.vu
citload18.co.vu
citload20.co.vu
citserv.co.vu
citserv18.co.vu
citserv20.co.vu
citserver.co.vu
clay.su
clickrate.com.au
cndbase.ru
cogoda.ru
coldrollicecream.com
collectcoins.su
congrbasering.su
consulenzaeprogettazioneintegrata.it
consumerfocusedconspicuously.net
cont24x7host.org
counter-1.adscounter.com.ua
cp1.0pu.ru
cp2.0pu.ru
cpxp.0pu.ru
creamlonsarter.co.uk
createlognet.co.uk
creationsarchitecture.com
crete-holidays.com.gr
cs.co.vu
ctoavto.by
cumfaci.eu
customerimpacts.com
```

Figure 6.10. Sample text file with URLs supplied to Webtimer

The text file that was fed to the SIEM contained 700 potentially malicious or suspicious URLs. Webtimer was then executed to automatically access each of the URLs in the text file via the squid proxy. Access by Webtimer to each of the domains was logged by the squid proxy and events sent to the SIEM.

The purpose behind using an automated tool was it allowed this researcher to browse to as many potentially malicious or suspicious URLs, an exercise that manually done would have taken an unreasonable significant amount of time. The use of purpose generation of events has been observed in research. Grimaila, Myers, Mills & Peterson (2011) use a proprietary tool called SAST to generate events for analysis in the research. Aguirre & Alonso (2012), use a tool called Nikto to generate web scans against a webserver, monitored by the SNORT IDS. The generated events were logged to log storage for further analysis.

Figure 6.11 shows the WebTimer tool accessing a URL listed in the text file.



ERROR

The requested URL could not be retrieved

The following error was encountered while trying to retrieve the URL: http://3apa3a.tomsk.tw/

Figure 6.11. Web timer automatically accessing potentially malicious URL

Monitoring traffic to potential and real malicious destination

Once the preparatory work of setting up the log sources to send logs to the SIEM, ensuring key data elements were parsed and extracted into relevant fields and relevant traffic was generated to malicious or suspicious destinations; the logic around using the SIEM to identify traffic destined for potential or real malicious destinations and to determine the threatscore or risk of this traffic was then defined.

The logic to determine traffic to potentially malicious or suspicious and associated threatscore was initially expressed as illustrated in Figure 6.12.

Extract from the SIEM traffic from the squid and firewall log sources

Determine which field to lookup against IP reputation service which in this case was the honeyport.org IP reputation service

Lookup the threatscore of each of the destination IP addressed against the external IP reputation service.

Filter out for destination IP addresses with a threatscore greater than 0

Filter out duplicates

Check results against other IP reputation services

Figure 6.12. Outbound threat monitoring SIEM use case logic

The logic in Figure 6.12 was finally translated into a language interpretable by the Splunk SIEM as illustrated in Figure 6.13 where:

- | ***(eventtype=ip_check OR eventtype=ip_checksquid)*** | defined the event sources of the firewall and squid logs
- | ***lookup threatscore clientip*** | was a function that looked up the threatscore of the ***clientip*** with ***clientip*** referring to the destination IP address
- | ***search threatscore>0*** | was expected to return any destination IP address with a threat score greater than zero
- |***dedup clientip***| remove duplicates
- |***table clientip threatscore***| presented the results in table format

```
(eventtype=ip_check OR eventtype=ip_checksquid ) | lookup threatscore clientip | search threatscore>0 | dedup clientip | table clientip threatscore
```

Figure 6.13. Translated SIEM use case logic

Table 6.3 give the results of running the IP reputation logic against the squid and firewall event logs in the SIEM.

Honeyport.org Threatscore or risk rating	No of IP addresses	% of total No. of IP address
43	1	3%
32	2	5%
30	1	3%
29	2	5%
27	1	3%
26	4	11%
25	1	3%
24	1	3%
22	2	5%
21	4	11%
20	2	5%
18	1	3%
16	1	3%
13	1	3%
12	1	3%
10	1	3%
8	5	13%
7	1	3%
6	2	5%
4	4	11%
Totals	**38**	**100%**

Table 6.3. Destination IP address and threatscore results

As shown in Table 6.3, a total of 38 destination IP address were identified with a threatscore greater than 0. The threatscores ranged between 4 and 43 with an average threatscore of 19.65.

For verification and risk rating confirmation and to get a second opinion, each of the 38 IP addresses' risk rating was checked against another publicly available IP reputation service managed by WatchGuard® and hosted at www.reputationauthority.org. The results of this additional verification is shown in Table 6.4.

Reputationauthory.org Risk rating	No of IP addresses	% of total No. of IP address
100	2	5%
98	1	3%
96	1	3%
95	2	5%
92	1	3%
91	1	3%
90	2	5%
89	10	26%
86	1	3%
85	8	21%
52	2	5%
50	3	8%
49	1	3%
29	1	3%
22	1	3%
13	1	3%
	38	100%

Table 6.4. Reputation.org based risk rating

As shown in Table 6.4, the risk rating based on the reputationauthority.org website ranged between 13 and 100 with an average score of 71.

It was observed that the reputationauthority.org IP reputation service gave higher risk ratings on the 38 IP addresses compared to the Honeyport.org threat scoring service. The average risk rate difference between the honeyport.org and the reputationauthory.org was 60 for each of the IP addresses. It could not be confirmed the reason behind the significant difference in risk rating, however it can be assumed that the differences are due to the background data collection and feedback mechanism that is used by the two services to rate the websites, the actual mechanism of which are not normally available in the public domain.

Section 6.3 of this Chapter puts these observations into perspective.

6.2.2 User Activity Monitoring SIEM Use Case

6.2.2.1 Background to SIEM Use Case

User activity monitoring in the last few year has seen increased attention primarily driven by the threat of exposure from insiders, commonly referred to as insider threats (Grimaila, Myers, Mills, 2009). User related activity that can typically be monitored includes:

- Successful and failed authentication events
- Attempts to access particular applications
- The use of privileged accounts on the network
- Creation and deletion of user accounts

Emphasis by regulatory standards such as the PCI DSS (PCI Security Standards Council, 2013) on the need to monitor user activity has also entrenched the need for organisations to monitor user activity in particular privileged user account (Chuvakin, Phillips & Schmidt, 2013). For instance requirement 10 of the PCI DSS standard specifies (PCI Security Standards Council, 2013) to "Track and monitor all access to network resources and cardholder data". Sub requirement 10.2.1 further expounds on this requirement by stating that automated audit trails be implemented to monitor "All individual user accesses to cardholder data". Sub requirement 10.2.2 on the other hand requires the monitoring of "All actions taken by any individual with root or administrative privileges" (PCI Security Standards Council, 2013).

While ordinary user accounts have limited access privileges, and are typically based on the principle of least privilege (Chapple, Stewart & Tittel, 2008), privileged user accounts would have been assigned literally unlimited administrative access rights. Grimaila, Myers & Mills (2009), note that while these additional, unlimited user rights are assigned for use in genuine daily operational tasks, there is growing abuse of privileged user access to carry out malicious activity.

The focus of exploring this SIEM use case was to investigate the applicability of SIEM in IT security from a user activity perspective.

6.2.2.2 Exploring SIEM use case

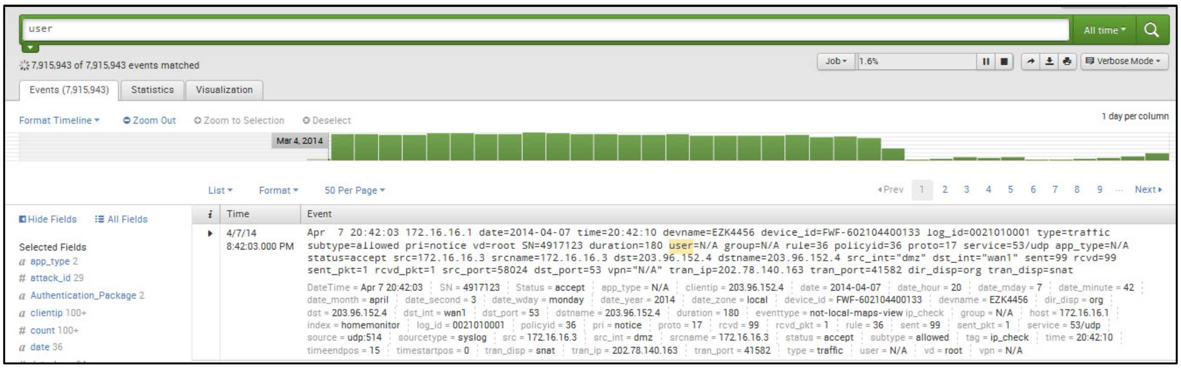
The actions performed for this SIEM use case focused on exploring using the SIEM to actively monitoring user activity from an IT security perspective. The normalisation SIEM capability as presented in Chapter 4 had a key role in exploring this SIEM use case

The main tasks performed in exploring this SIEM use case were:

- Searching for all user activity related events
- Extracting user related data elements from these events
- Normalising privileged user event related logs for privileged user account monitoring

<u>Searching for user activity related events</u>

The first task that was performed in searching for user activity related events was to enter the search string "*user*" to find all user related activity. Figure 6.14 shows the string "*user*" enter into the Splunk SIEM search bar.



```
user
```

7,915,943 of 7,915,943 events matched

Selected Fields
a app_type 2
# attack_id 29
a Authentication_Package 2
a clientip 100+
# count 100+
a date 36

4/7/14
8:42:03.000 PM

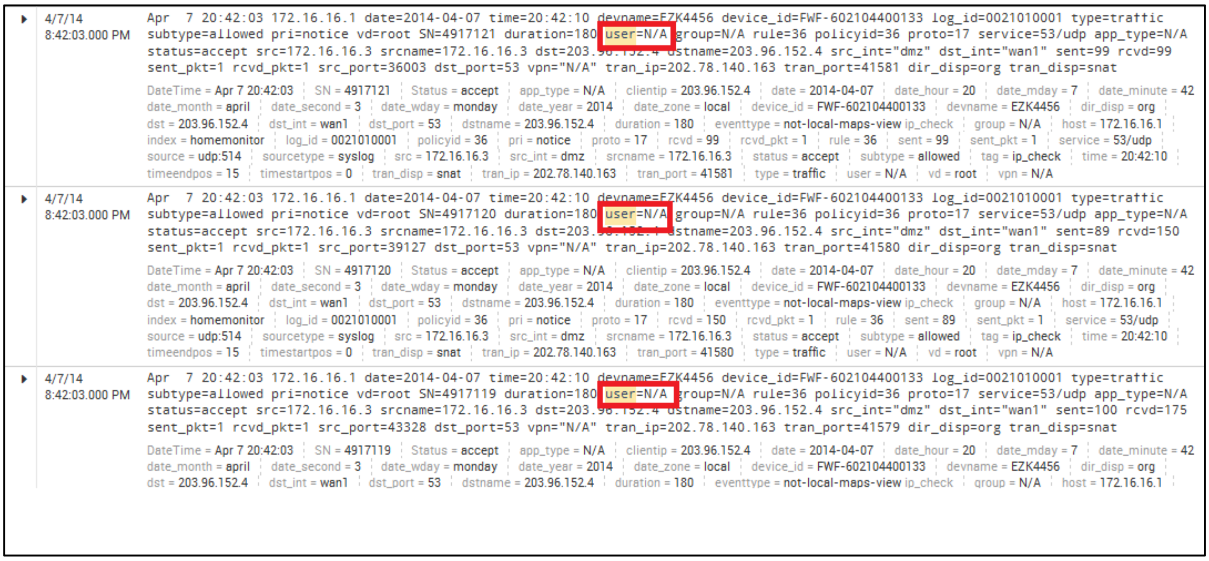
```
Apr  7 20:42:03 172.16.16.1 date=2014-04-07 time=20:42:10 devname=EZK4456 device_id=FWF-602104400133 log_id=0021010001 type=traffic
subtype=allowed pri=notice vd=root SN=4917123 duration=180 user=N/A group=N/A rule=36 policyid=36 proto=17 service=53/udp app_type=N/A
status=accept src=172.16.16.3 srcname=172.16.16.3 dst=203.96.152.4 dstname=203.96.152.4 src_int="dmz" dst_int="wan1" sent=99 rcvd=99
sent_pkt=1 rcvd_pkt=1 src_port=58024 dst_port=53 vpn="N/A" tran_ip=202.78.140.163 tran_port=41582 dir_disp=org tran_disp=snat
```

```
DateTime = Apr 7 20:42:03 | SN = 4917123 | Status = accept | app_type = N/A | clientip = 203.96.152.4 | date = 2014-04-07 | date_hour = 20 | date_mday = 7 | date_minute = 42
date_month = april | date_second = 3 | date_wday = monday | date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | dir_disp = org
dst = 203.96.152.4 | dst_int = wan1 | dst_port = 53 | dstname = 203.96.152.4 | duration = 180 | eventtype = not-local-maps-view ip_check | group = N/A | host = 172.16.16.1
index = homemonitor | log_id = 0021010001 | policyid = 36 | pri = notice | proto = 17 | rcvd = 99 | rcvd_pkt = 1 | rule = 36 | sent = 99 | sent_pkt = 1 | service = 53/udp
source = udp:514 | sourcetype = syslog | src = 172.16.16.3 | src_int = dmz | srcname = 172.16.16.3 | status = accept | subtype = allowed | tag = ip_check | time = 20:42:10
timeendpos = 15 | timestartpos = 0 | tran_disp = snat | tran_ip = 202.78.140.163 | tran_port = 41582 | type = traffic | user = N/A | vd = root | vpn = N/A
```

Figure 6.14. Searching for user related activity

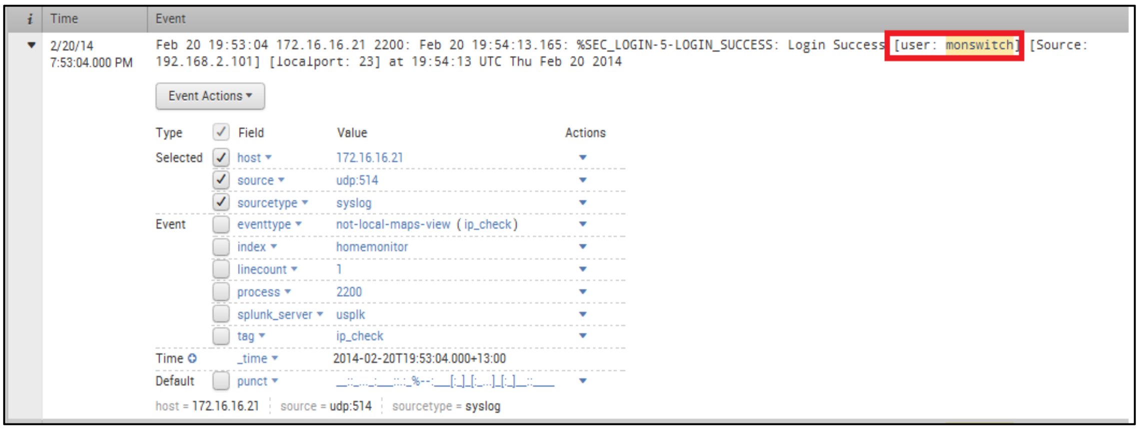
The assumption in this first step was that the majority of user related events would have the string "user" in the event message, followed by the actual user name of the user that triggered that event.

It was observed that the results returned from this search contained a significant amount of irrelevant user data such as that shown in Figure 6.15 where the event logs contained the string "**user=N/A**". The search was therefore refined to exclude event logs containing "***user=N/A***" by applying the search string "**user *user!=N/A***". From this refined search result, it was observed that while some events such as those from the cisco switch contained user related activity, the SIEM had not natively parsed and extracted the username data element as shown in Figure 6.16 where the user monswitch has not been extracted into the relevant user related field. It was also observed that the initial assumption that user related events would contain the "*user*" string followed by the username was not valid. Some events such as those from the ftp server just listed the username without the pre-pending "*user*" string.



```
4/7/14          Apr  7 20:42:03 172.16.16.1 date=2014-04-07 time=20:42:10 devname=FZK4456 device_id=FWF-602104400133 log_id=0021010001 type=traffic
8:42:03.000 PM  subtype=allowed pri=notice vd=root SN=4917121 duration=180 user=N/A group=N/A rule=36 policyid=36 proto=17 service=53/udp app_type=N/A
                status=accept src=172.16.16.3 srcname=172.16.16.3 dst=203.96.152.4 dstname=203.96.152.4 src_int="dmz" dst_int="wan1" sent=99 rcvd=99
                sent_pkt=1 rcvd_pkt=1 src_port=36003 dst_port=53 vpn="N/A" tran_ip=202.78.140.163 tran_port=41581 dir_disp=org tran_disp=snat
                DateTime = Apr 7 20:42:03 | SN = 4917121 | Status = accept | app_type = N/A | clientip = 203.96.152.4 | date = 2014-04-07 | date_hour = 20 | date_mday = 7 | date_minute = 42
                date_month = april | date_second = 3 | date_wday = monday | date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | dir_disp = org
                dst = 203.96.152.4 | dst_int = wan1 | dst_port = 53 | dstname = 203.96.152.4 | duration = 180 | eventtype = not-local-maps-view ip_check | group = N/A | host = 172.16.16.1
                index = homemonitor | log_id = 0021010001 | policyid = 36 | pri = notice | proto = 17 | rcvd = 99 | rcvd_pkt = 1 | rule = 36 | sent = 99 | sent_pkt = 1 | service = 53/udp
                source = udp:514 | sourcetype = syslog | src = 172.16.16.3 | src_int = dmz | srcname = 172.16.16.3 | status = accept | subtype = allowed | tag = ip_check | time = 20:42:10
                timeendpos = 15 | timestartpos = 0 | tran_disp = snat | tran_ip = 202.78.140.163 | tran_port = 41581 | type = traffic | user = N/A | vd = root | vpn = N/A
4/7/14          Apr  7 20:42:03 172.16.16.1 date=2014-04-07 time=20:42:10 devname=FZK4456 device_id=FWF-602104400133 log_id=0021010001 type=traffic
8:42:03.000 PM  subtype=allowed pri=notice vd=root SN=4917120 duration=180 user=N/A group=N/A rule=36 policyid=36 proto=17 service=53/udp app_type=N/A
                status=accept src=172.16.16.3 srcname=172.16.16.3 dst=203.96.152.4 dstname=203.96.152.4 src_int="dmz" dst_int="wan1" sent=89 rcvd=150
                sent_pkt=1 rcvd_pkt=1 src_port=39127 dst_port=53 vpn="N/A" tran_ip=202.78.140.163 tran_port=41580 dir_disp=org tran_disp=snat
                DateTime = Apr 7 20:42:03 | SN = 4917120 | Status = accept | app_type = N/A | clientip = 203.96.152.4 | date = 2014-04-07 | date_hour = 20 | date_mday = 7 | date_minute = 42
                date_month = april | date_second = 3 | date_wday = monday | date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | dir_disp = org
                dst = 203.96.152.4 | dst_int = wan1 | dst_port = 53 | dstname = 203.96.152.4 | duration = 180 | eventtype = not-local-maps-view ip_check | group = N/A | host = 172.16.16.1
                index = homemonitor | log_id = 0021010001 | policyid = 36 | pri = notice | proto = 17 | rcvd = 150 | rcvd_pkt = 1 | rule = 36 | sent = 89 | sent_pkt = 1 | service = 53/udp
                source = udp:514 | sourcetype = syslog | src = 172.16.16.3 | src_int = dmz | srcname = 172.16.16.3 | status = accept | subtype = allowed | tag = ip_check | time = 20:42:10
                timeendpos = 15 | timestartpos = 0 | tran_disp = snat | tran_ip = 202.78.140.163 | tran_port = 41580 | type = traffic | user = N/A | vd = root | vpn = N/A
4/7/14          Apr  7 20:42:03 172.16.16.1 date=2014-04-07 time=20:42:10 devname=FZK4456 device_id=FWF-602104400133 log_id=0021010001 type=traffic
8:42:03.000 PM  subtype=allowed pri=notice vd=root SN=4917119 duration=180 user=N/A group=N/A rule=36 policyid=36 proto=17 service=53/udp app_type=N/A
                status=accept src=172.16.16.3 srcname=172.16.16.3 dst=203.96.152.4 dstname=203.96.152.4 src_int="dmz" dst_int="wan1" sent=100 rcvd=175
                sent_pkt=1 rcvd_pkt=1 src_port=43328 dst_port=53 vpn="N/A" tran_ip=202.78.140.163 tran_port=41579 dir_disp=org tran_disp=snat
                DateTime = Apr 7 20:42:03 | SN = 4917119 | Status = accept | app_type = N/A | clientip = 203.96.152.4 | date = 2014-04-07 | date_hour = 20 | date_mday = 7 | date_minute = 42
                date_month = april | date_second = 3 | date_wday = monday | date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | dir_disp = org
                dst = 203.96.152.4 | dst_int = wan1 | dst_port = 53 | dstname = 203.96.152.4 | duration = 180 | eventtype = not-local-maps-view ip_check | group = N/A | host = 172.16.16.1
```

Figure 6.15. Search showing irrelevant user event data



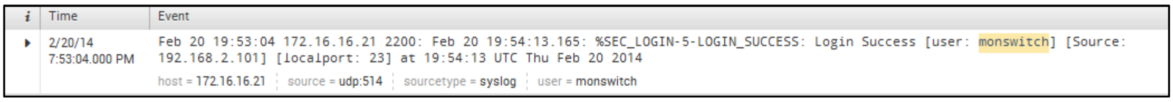
```
i  Time            Event
   2/20/14         Feb 20 19:53:04 172.16.16.21 2200: Feb 20 19:54:13.165: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: monswitch] [Source:
   7:53:04.000 PM  192.168.2.101] [localport: 23] at 19:54:13 UTC Thu Feb 20 2014

                   Event Actions ▾

                   Type      ☑  Field            Value                                   Actions
                   Selected  ☑  host ▾           172.16.16.21                            ▾
                             ☑  source ▾         udp:514                                 ▾
                             ☑  sourcetype ▾     syslog                                  ▾
                   Event     □  eventtype ▾      not-local-maps-view ( ip_check )        ▾
                             □  index ▾          homemonitor                             ▾
                             □  linecount ▾      1                                       ▾
                             □  process ▾        2200                                    ▾
                             □  splunk_server ▾  usplk                                   ▾
                             □  tag ▾            ip_check                                ▾
                   Time ⚙       _time ▾          2014-02-20T19:53:04.000+13:00
                   Default   □  punct ▾          __::_..._:___:::_%--:___[:_]_[:_...]_[:_]__::____   ▾
                   host = 172.16.16.21 | source = udp:514 | sourcetype = syslog
```

Figure 6.16. Cisco switch event message

Extracting user related fields from these events

The username data element was therefore manually extracted for cisco switch user related events using a similar data element extraction process as presented in section 6.3

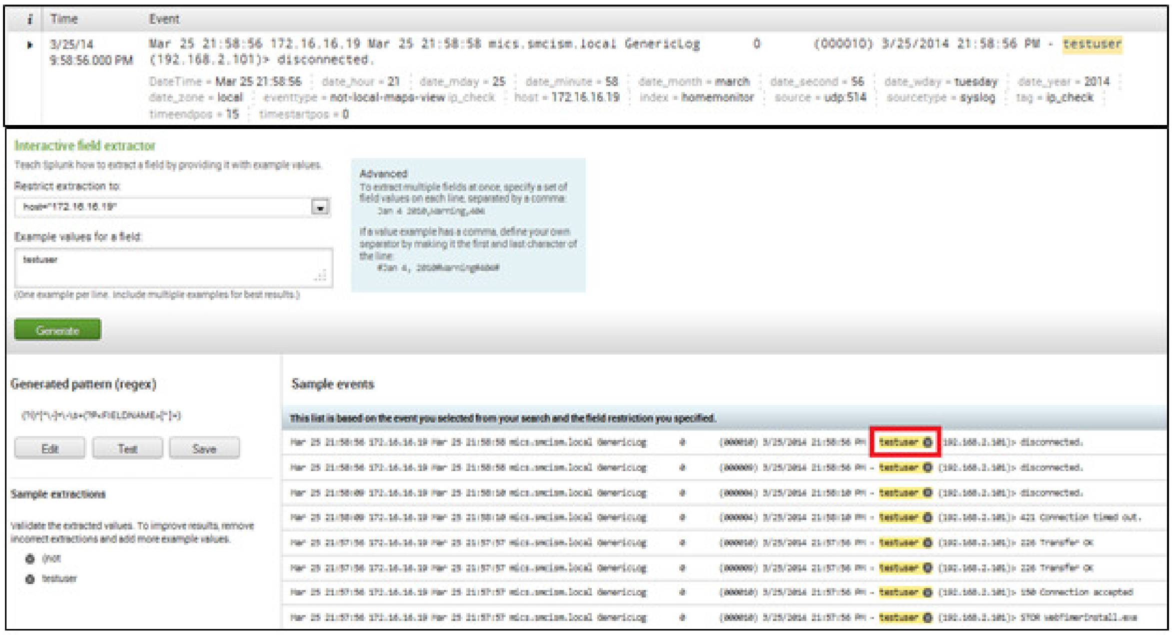
of this Chapter. This user extraction process was repeated for all manually identified event logs where the user related data element was not natively parsed by the SIEM. For the cisco switch logs, the result of the username data element extraction is as shown in Figure 6.17.



i	Time	Event
▸	2/20/14 7:53:04.000 PM	Feb 20 19:53:04 172.16.16.21 2200: Feb 20 19:54:13.165: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: monswitch] [Source: 192.168.2.101] [localport: 23] at 19:54:13 UTC Thu Feb 20 2014 host = 172.16.16.21 ⁞ source = udp:514 ⁞ sourcetype = syslog ⁞ user = monswitch

Figure 6.17. Extracted user related data from Cisco switch event

Figure 6.18 shows the username in the Filezilla logs being extracted. Figure 6.19 shows the results.



i	Time	Event
▸	3/25/14 9:58:56.000 PM	Mar 25 21:58:56 172.16.16.19 Mar 25 21:58:58 mics.smcism.local GenericLog 0 (000010) 3/25/2014 21:58:56 PM - testuser (192.168.2.101)> disconnected. DateTime = Mar 25 21:58:56 ⁞ date_hour = 21 ⁞ date_mday = 25 ⁞ date_minute = 58 ⁞ date_month = march ⁞ date_second = 56 ⁞ date_wday = tuesday ⁞ date_year = 2014 ⁞ date_zone = local ⁞ eventtype = not-local-maps-view ip_check ⁞ host = 172.16.16.19 ⁞ index = homemonitor ⁞ source = udp:514 ⁞ sourcetype = syslog ⁞ tag = ip_check ⁞ timeendpos = 15 ⁞ timestartpos = 0

Interactive field extractor

Teach Splunk how to extract a field by providing it with example values.

Restrict extraction to: host="172.16.16.19"

Example values for a field: testuser

(One example per line. Include multiple examples for best results.)

Advanced
To extract multiple fields at once, specify a set of field values on each line, separated by a comma:
    Jan 4 2010,Warning,404

If a value example has a comma, define your own separator by making it the first and last character of the line:
    #Jan 4, 2010#Warning#404#

Generated pattern (regex)

```
(?i)^[^\-]*\-\s+(?P<FIELDNAME>[^ ]+)
```

Sample extractions

Validate the extracted values. To improve results, remove incorrect extractions and add more example values.

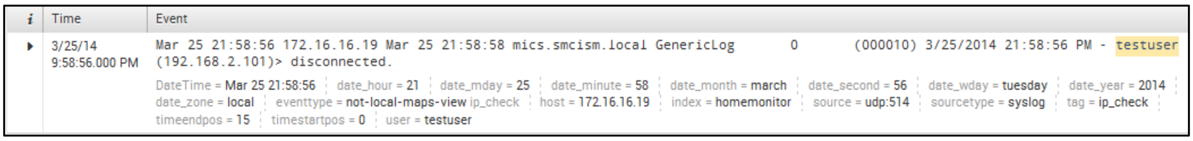
- (not
- testuser

Sample events

This list is based on the event you selected from your search and the field restriction you specified.

```
Mar 25 21:58:56 172.16.16.19 Mar 25 21:58:58 mics.smcism.local GenericLog    0    (000010) 3/25/2014 21:58:56 PM testuser (192.168.2.101)> disconnected.
Mar 25 21:58:56 172.16.16.19 Mar 25 21:58:58 mics.smcism.local GenericLog    0    (000009) 3/25/2014 21:58:56 PM - testuser (192.168.2.101)> disconnected.
Mar 25 21:58:09 172.16.16.19 Mar 25 21:58:10 mics.smcism.local GenericLog    0    (000004) 3/25/2014 21:58:10 PM - testuser (192.168.2.101)> disconnected.
Mar 25 21:58:09 172.16.16.19 Mar 25 21:58:10 mics.smcism.local GenericLog    0    (000004) 3/25/2014 21:58:10 PM - testuser (192.168.2.101)> 421 Connection timed out.
Mar 25 21:57:56 172.16.16.19 Mar 25 21:57:57 mics.smcism.local GenericLog    0    (000010) 3/25/2014 21:57:56 PM - testuser (192.168.2.101)> 226 Transfer OK
Mar 25 21:57:56 172.16.16.19 Mar 25 21:57:57 mics.smcism.local GenericLog    0    (000009) 3/25/2014 21:57:56 PM - testuser (192.168.2.101)> 226 Transfer OK
Mar 25 21:57:56 172.16.16.19 Mar 25 21:57:57 mics.smcism.local GenericLog    0    (000010) 3/25/2014 21:57:56 PM - testuser (192.168.2.101)> 150 Connection accepted
Mar 25 21:57:56 172.16.16.19 Mar 25 21:57:57 mics.smcism.local GenericLog    0    (000010) 3/25/2014 21:57:56 PM - testuser (192.168.2.101)> STOR webTimerInstall.exe
```

Figure 6.18. Filezilla username extraction

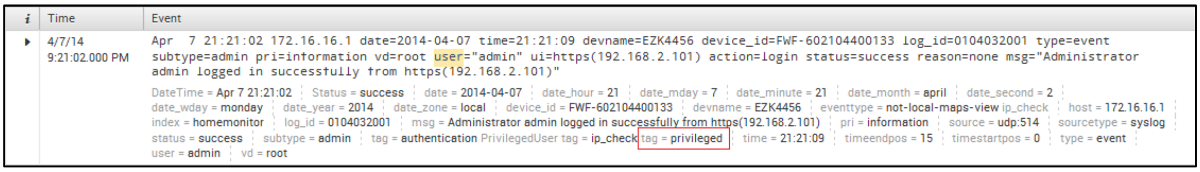


i	Time	Event
▸	3/25/14 9:58:56.000 PM	Mar 25 21:58:56 172.16.16.19 Mar 25 21:58:58 mics.smcism.local GenericLog 0 (000010) 3/25/2014 21:58:56 PM - testuser (192.168.2.101)> disconnected. DateTime = Mar 25 21:58:56 ⁞ date_hour = 21 ⁞ date_mday = 25 ⁞ date_minute = 58 ⁞ date_month = march ⁞ date_second = 56 ⁞ date_wday = tuesday ⁞ date_year = 2014 ⁞ date_zone = local ⁞ eventtype = not-local-maps-view ip_check ⁞ host = 172.16.16.19 ⁞ index = homemonitor ⁞ source = udp:514 ⁞ sourcetype = syslog ⁞ tag = ip_check ⁞ timeendpos = 15 ⁞ timestartpos = 0 ⁞ user = testuser

Figure 6.19. Extracted filezilla username

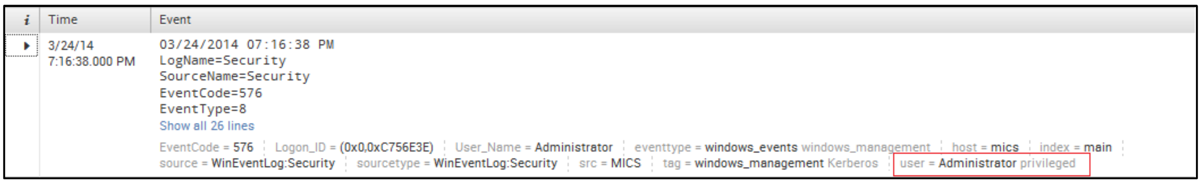
Normalising privileged event related logs for privileged user account monitoring

The final task in exploring this SIEM use case was investigate the process of normalising user related events to track privileged user account activity. This was achieved through the Splunk SIEM process called tagging. Prior to tagging of privileged user account related activity, there was little way of identifying only privileged user accounts except through the knowledge of the actual privileged user accounts. By tagging each privileged user account with the "*privileged*" tag as shown in Figure 6.20, Figure 6.21 and Figure 6.22, it was then possible to easily monitor privileged user accounts simply by retrieving accounts with the tag "*privileged*".



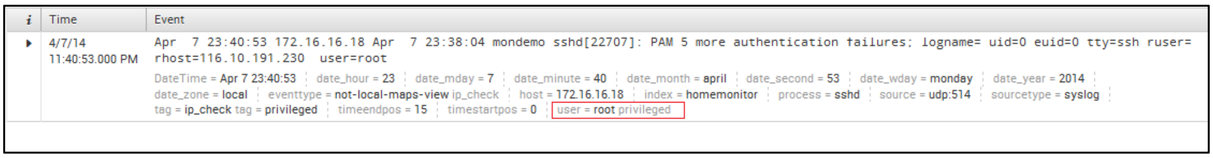
```
4/7/14
9:21:02.000 PM

Apr  7 21:21:02 172.16.16.1 date=2014-04-07 time=21:21:09 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032001 type=event
subtype=admin pri=information vd=root user="admin" ui=https(192.168.2.101) action=login status=success reason=none msg="Administrator
admin logged in successfully from https(192.168.2.101)"
DateTime = Apr 7 21:21:02 | Status = success | date = 2014-04-07 | date_hour = 21 | date_mday = 7 | date_minute = 21 | date_month = april | date_second = 2 |
date_wday = monday | date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | eventtype = not-local-maps-view ip_check | host = 172.16.16.1 |
index = homemonitor | log_id = 0104032001 | msg = Administrator admin logged in successfully from https(192.168.2.101) | pri = information | source = udp:514 | sourcetype = syslog |
status = success | subtype = admin | tag = authentication PrivilegedUser tag = ip_check tag = privileged | time = 21:21:09 | timeendpos = 15 | timestartpos = 0 | type = event |
user = admin | vd = root
```

Figure 6.20. Tagged firewall event log



```
3/24/14
7:16:38.000 PM

03/24/2014 07:16:38 PM
LogName=Security
SourceName=Security
EventCode=576
EventType=8
Show all 26 lines
EventCode = 576 | Logon_ID = (0x0,0xC756E3E) | User_Name = Administrator | eventtype = windows_events windows_management | host = mics | index = main |
source = WinEventLog:Security | sourcetype = WinEventLog:Security | src = MICS | tag = windows_management Kerberos | user = Administrator privileged
```

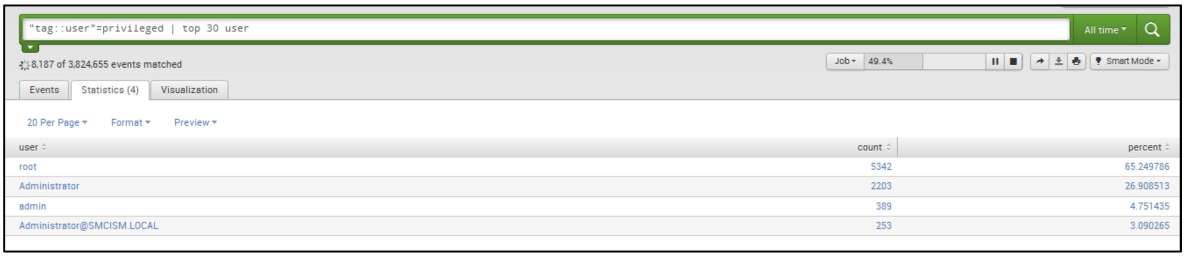
Figure 6.21. Tagged windows event log



```
4/7/14
11:40:53.000 PM

Apr  7 23:40:53 172.16.16.18 Apr  7 23:38:04 mondemo sshd[22707]: PAM 5 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser=
rhost=116.10.191.230  user=root
DateTime = Apr 7 23:40:53 | date_hour = 23 | date_mday = 7 | date_minute = 40 | date_month = april | date_second = 53 | date_wday = monday | date_year = 2014 |
date_zone = local | eventtype = not-local-maps-view ip_check | host = 172.16.16.18 | index = homemonitor | process = sshd | source = udp:514 | sourcetype = syslog |
tag = ip_check tag = privileged | timeendpos = 15 | timestartpos = 0 | user = root privileged
```

Figure 6.22. Tagged Linux event log

Figure 6.23 shows a search for privileged user accounts after normalisation through tagging.



```
"tag::user"=privileged | top 30 user
```

8,187 of 3,824,655 events matched

user	count	percent
root	5342	65.249786
Administrator	2203	26.908513
admin	389	4.751435
Administrator@SMCISM.LOCAL	253	3.090265

Figure 6.23. Normalised user event log

One of the primary observations highlighted from this exploration was, having to analyse event messages of varying formats. As observed, some manual intervention was required in order to get to the point where for instance privileged user account could easily be monitored and reported on. From an IT security perspective it was noted the importance of having a tool capable of enabling this process automatically or manually.

Section 6.3 of this Chapter further expounds on this exploration.

6.2.3 Compliance and Reporting Monitoring SIEM Use Case

6.2.3.1 Background to SIEM Use Case

The contribution of compliance to SIEM adoption in information security has been discussed in Chapters 2, 3 and 4. In essence, research has shown that the majority of organisations are adopting SIEM due to a need to satisfy one or more regulatory or audit requirements (Chuvakin, 2010). It therefore seemed natural in the identification of SIEM use cases to consider exploring SIEM and compliance.

A review of the major compliance and regulatory standards revealed that regulatory standards such as SOX, HIPAA and the Gramm-Leach-Bliley Act have generic requirements around log management (Kent & Souppaya, 2006). Of the compliance standards reviewed, the PCI DSS security standard (PCI Security Standards Council, 2013) provides much more specifics on logging and monitoring requirements compared to the other regulatory standards (Chuvakin, Phillips & Schmidt, 2013). There currently exists no “industry-wide recommendation” regarding logging requirements (Chuvakin, Phillips & Schmidt, 2013).

The logging and monitoring requirements in PCI DSS are defined in 8 sub requirement mandates which specify what in relation to monitoring should be monitored. PCI DSS mandate requirement 10 mandates organisations to “Regularly monitor and test networks” (PCI Security Standards Council, 2013). Sub mandates under requirement mandate 10 are summarised as:

- Sub requirement 10.1 requires the implementation of audit trails linking access to users.
- Sub requirement 10.2 mandates the implementation of audit trails to monitor:

    - User access to cardholder data
    - Privileged user account activity
    - User access to audit trails and any other related activity
    - Any failed access attempts
    - Any modifications to user accounts
    - Creation of modification to system objects
- Sub requirement 10.3 mandates user activity audit trails to contain
    - Details of the user
    - Event type
    - The date and time
    - Status of user activity, for instance success or failure
    - Event origination
    - Log source identity
- Sub requirement 10.4 mandates time synchronisation for log sources
- Sub requirement 10.5 mandates the securing of the audit trails from alteration
- Sub requirement 10.6 mandates the reviewing of the event logs in order to identify any "anomalies or suspicious traffic"
- Sub requirement 10.7 mandates log retention of up to a year for the audit trail history and up to 3 months of events retained for analysis purposes
- Sub requirement 10.8 mandates user awareness with respect polices that address access monitoring

Because of the detail provided by the PCI DSS standard with respect to event logs, the PCI DSS security standard was referenced in the exploration of the compliance SIEM use case. Specifically, requirement mandate 10 of the standard was referenced. The full PCI DSS mandate 10 is show in Appendix 2.

The focus of exploring this SIEM use case was to investigate the applicability of SIEM in IT security from a compliance perspective.

6.2.3.2 Exploring SIEM use case

The exploration of SIEM and compliance followed an audit based approach (Chickowski, 2012). A similar approach would be followed in a real SIEM deployment. While a similar process within a real environment would be more involving

(Chickowski, 2012); a typical auditor would be working with similar data as that which was collected for this experiment. While the requirements in mandate 10 of the PCI DSS standard are targeted at cardholder environments, this researcher approached the mandate from a generality perspective. Of the 8 sub requirements under requirement mandate 10, three sub requirements were deemed applicable to this SIEM use case. These sub requirements were sub requirements 10.1; 10.2; 10.3 and 10.6. The other 4 sub requirements were left out as they could not be fully explored in this experiment.

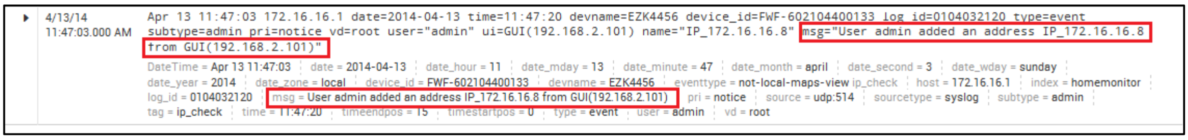
The following are the observations of compliance within the context of a SIEM environment.

Sub requirement 10.1 of the PCI DSS security standard mandates the implementation of audit trails on monitored systems with the additional requirement of having the ability to ensure a relationship exists between individual user activity and generated events. It has been observed in Chapter 4 as well as in section 6.2.1 that a SIEM environment is enabled by applications, systems and devices sending logs to the SIEM. With respect to requirement 10.1 of the PCI DSS, the deployment of a SIEM within an environment where PCI DSS compliance is mandated it is expected that auditing or logging is enabled on monitored devices. These logs or audit trails should then be sent to the SIEM for centralised storage and centralised analysis.

Sub requirement 10.2 of the PCI DSS security standard mandates the implementation of audit trails for the purpose of monitoring specific activity within a cardholder environment. The monitored activity is further defined through minor sub requirements 10.2.1 to 10.2.7 (Refer to Appendix 2). The minor sub requirements address; monitoring of individual access to card holder data, monitoring of privileged user activity, monitoring of access to event logs or audit trails, monitoring of failed system access, change monitoring, monitoring of attempts to modify functionality of the logging service and any attempts to modify system objects. Section 6.2.2 in part explored the applicability of SIEM to general user and privileged user activity monitoring. From the event logs collected in the SIEM, it was also observed that it was possible to monitor:

- Monitoring of failed systems access
- User changes
- Attempts to modify system objects
- Other user authentication related events

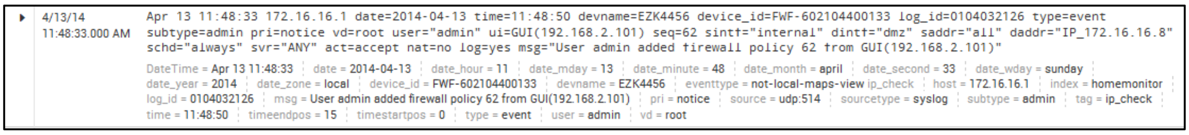
As an example, the following was observed with respect to the change monitoring requirement. The observation was for monitoring configuration changes on the firewall Figure 6.24 shows an event logged for a new IP address object created on the firewall.



```
4/13/14            Apr 13 11:47:03 172.16.16.1 date=2014-04-13 time=11:47:20 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032120 type=event
11:47:03.000 AM    subtype=admin pri=notice vd=root user="admin" ui=GUI(192.168.2.101) name="IP_172.16.16.8" msg="User admin added an address IP_172.16.16.8
                   from GUI(192.168.2.101)"
                   DateTime = Apr 13 11:47:03 ¦ date = 2014-04-13 ¦ date_hour = 11 ¦ date_mday = 13 ¦ date_minute = 47 ¦ date_month = april ¦ date_second = 3 ¦ date_wday = sunday ¦
                   date_year = 2014 ¦ date_zone = local ¦ device_id = FWF-602104400133 ¦ devname = EZK4456 ¦ eventtype = not-local-maps-view ip_check ¦ host = 172.16.16.1 ¦ index = homemonitor ¦
                   log_id = 0104032120 ¦ msg = User admin added an address IP_172.16.16.8 from GUI(192.168.2.101) ¦ pri = notice ¦ source = udp:514 ¦ sourcetype = syslog ¦ subtype = admin ¦
                   tag = ip_check ¦ time = 11:47:20 ¦ timeendpos = 15 ¦ timestartpos = 0 ¦ type = event ¦ user = admin ¦ vd = root
```

Figure 6.24. New IP address object added to firewall

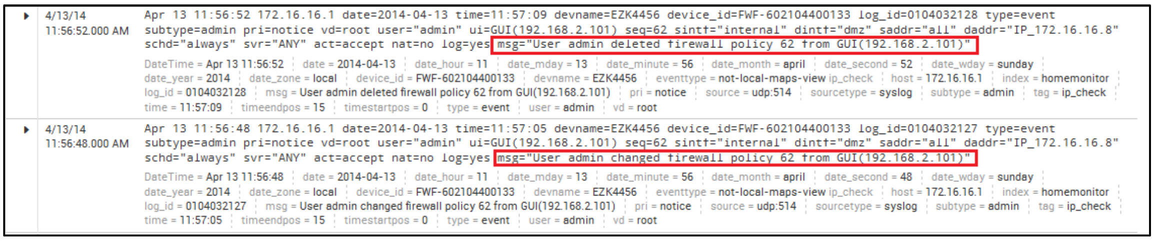
Figure 6.25 on the other hand, shows the creation of a firewall rule referencing the newly created IP address object of 172.16.16.8 shown in Figure 6.24



```
4/13/14            Apr 13 11:48:33 172.16.16.1 date=2014-04-13 time=11:48:50 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032126 type=event
11:48:33.000 AM    subtype=admin pri=notice vd=root user="admin" ui=GUI(192.168.2.101) seq=62 sintf="internal" dintf="dmz" saddr="all" daddr="IP_172.16.16.8"
                   schd="always" svr="ANY" act=accept nat=no log=yes msg="User admin added firewall policy 62 from GUI(192.168.2.101)"
                   DateTime = Apr 13 11:48:33 ¦ date = 2014-04-13 ¦ date_hour = 11 ¦ date_mday = 13 ¦ date_minute = 48 ¦ date_month = april ¦ date_second = 33 ¦ date_wday = sunday ¦
                   date_year = 2014 ¦ date_zone = local ¦ device_id = FWF-602104400133 ¦ devname = EZK4456 ¦ eventtype = not-local-maps-view ip_check ¦ host = 172.16.16.1 ¦ index = homemonitor ¦
                   log_id = 0104032126 ¦ msg = User admin added firewall policy 62 from GUI(192.168.2.101) ¦ pri = notice ¦ source = udp:514 ¦ sourcetype = syslog ¦ subtype = admin ¦ tag = ip_check ¦
                   time = 11:48:50 ¦ timeendpos = 15 ¦ timestartpos = 0 ¦ type = event ¦ user = admin ¦ vd = root
```

Figure 6.25. New firewall rule added

Finally Figure 6.26 shows an event from the firewall were a user disabled a rule on the firewall.



```
4/13/14            Apr 13 11:56:52 172.16.16.1 date=2014-04-13 time=11:57:09 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032128 type=event
11:56:52.000 AM    subtype=admin pri=notice vd=root user="admin" ui=GUI(192.168.2.101) seq=62 sintf="internal" dintf="dmz" saddr="all" daddr="IP_172.16.16.8"
                   schd="always" svr="ANY" act=accept nat=no log=yes msg="User admin deleted firewall policy 62 from GUI(192.168.2.101)"
                   DateTime = Apr 13 11:56:52 ¦ date = 2014-04-13 ¦ date_hour = 11 ¦ date_mday = 13 ¦ date_minute = 56 ¦ date_month = april ¦ date_second = 52 ¦ date_wday = sunday ¦
                   date_year = 2014 ¦ date_zone = local ¦ device_id = FWF-602104400133 ¦ devname = EZK4456 ¦ eventtype = not-local-maps-view ip_check ¦ host = 172.16.16.1 ¦ index = homemonitor ¦
                   log_id = 0104032128 ¦ msg = User admin deleted firewall policy 62 from GUI(192.168.2.101) ¦ pri = notice ¦ source = udp:514 ¦ sourcetype = syslog ¦ subtype = admin ¦ tag = ip_check ¦
                   time = 11:57:09 ¦ timeendpos = 15 ¦ timestartpos = 0 ¦ type = event ¦ user = admin ¦ vd = root
4/13/14            Apr 13 11:56:48 172.16.16.1 date=2014-04-13 time=11:57:05 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032127 type=event
11:56:48.000 AM    subtype=admin pri=notice vd=root user="admin" ui=GUI(192.168.2.101) seq=62 sintf="internal" dintf="dmz" saddr="all" daddr="IP_172.16.16.8"
                   schd="always" svr="ANY" act=accept nat=no log=yes msg="User admin changed firewall policy 62 from GUI(192.168.2.101)"
                   DateTime = Apr 13 11:56:48 ¦ date = 2014-04-13 ¦ date_hour = 11 ¦ date_mday = 13 ¦ date_minute = 56 ¦ date_month = april ¦ date_second = 48 ¦ date_wday = sunday ¦
                   date_year = 2014 ¦ date_zone = local ¦ device_id = FWF-602104400133 ¦ devname = EZK4456 ¦ eventtype = not-local-maps-view ip_check ¦ host = 172.16.16.1 ¦ index = homemonitor ¦
                   log_id = 0104032127 ¦ msg = User admin changed firewall policy 62 from GUI(192.168.2.101) ¦ pri = notice ¦ source = udp:514 ¦ sourcetype = syslog ¦ subtype = admin ¦ tag = ip_check ¦
                   time = 11:57:05 ¦ timeendpos = 15 ¦ timestartpos = 0 ¦ type = event ¦ user = admin ¦ vd = root
```

Figure 6.26. Firewall rule disabled

With this level of detail collected in the SIEM from various devices, it was observed that the SIEM could for instance be used to satisfy the change monitoring requirement of the PCI DSS standard. The SIEM event logs provided valuable information such as the user making the changes and the time the change was made thus providing crucial accountability detail which is a key compliance requirement in the PCI DSS standard. These events when correlated with similar change related events from multiple log sources can provide rich change monitoring and management reporting to an organisation.

Sub requirement 10.3 requires specific data elements to be recorded at a minimum. The data expected data elements are:

- User detail
- Event type
- Date and time
- Status of action that is successful or failed
- Where the event originated
- Affect systems component.

It was observed that, the requirements defined by sub requirement 10.3 are more log source dependent than SIEM dependent as a typical SIEM only collects events. The expectation is therefore that the application or system vendor would ensure that these data elements are available in the events logs. It was however also observed that from a security perspective, should these events be available in the event logs, the SIEM had to be capable of parsing and normalising the data elements for the data elements to be available for logical processing, for instance using correlation rules.

Finally sub requirement 10.6 states "Review logs and security events for all system components to identify anomalies or suspicious activity" (PCI Security Standards Council, 2013). It was observed that information security professional performing log analysis tasks as outlined in section 5.3.1 regularly could leverage a SIEM to meet this PCI DSS requirement.

6.2.4 Correlation SIEM Use Case

6.2.4.1 Background to SIEM Use Case

As presented in Chapter 4 correlation is the cross referencing of events drawn from multiple log sources in order to identify potential or real security attacks which would otherwise have not been obvious when considering only one data source. (Rothman, 2010). As also highlighted in Chapter 4, the SIEM correlation capability is one of the primary reasons why organisations deploy SIEM in information security.

The focus of the SIEM use case was the investigation of the applicability of SIEM to IT security through exploring correlation.

6.2.4.2 Exploring SIEM Use Case

In exploring correlation and SIEM in IT security, correlation-based threat monitoring scenarios were identified and run through the data collected in the SIEM lab. Three correlation-based threat monitoring scenarios were investigated which were:

1) Correlation of IPS events generated by the firewall IPS module with those of the squid proxy to determine which top 20 destination IP addresses triggering signatures with a severity level of critical.
2) Correlation of IPS events generated by the firewall IPS module with that of traffic coming through the squid proxy to determine which top 20 signatures are triggered and to determine their severity.
3) Determination of the threat score of the web traffic which triggered an IDS event having a severity level of critical.

The following describes the correlation-based threat scenarios.

Threat monitoring correlation scenario 1

Correlation of IPS events generated by the firewall IPS module with those of the squid proxy to determine which top 20 destination IP addresses triggering signatures with a severity level of critical

For this correlation scenario, correlation logic was first defined followed by the application of the correlation rule logic to data in the SIEM and observing the result. The correlation logic was defined as follows:

- Identification of IPS traffic generated by the firewall.
- Identification of redirected client traffic logged by the squid proxy and seen traversing the firewall
- Correlation of both sets of IPS and proxy traffic based on the destination IP address to determine the top 20 signatures triggering signatures with a severity level of critical.

The first step of identifying IPS generated by the firewall was performed filtering for events of "*type=IPS*". Figure 6.27 shows a sample of an IPS type event

```
Apr 8 22:01:23 172.16.16.1 date=2014-04-08 time=22:01:31 devname=EZK4456
device_id=FWF-602104400133 log_id=0420073001 type=ips subtype=anomaly
pri=alert vd=root serial=5410903 attack_id=100663396 severity=critical
src=172.16.16.19 dst=172.16.16.3 src_port=22112 dst_port=8000 src_int="wan2"
dst_int=N/A status=clear_session proto=6 service=8000/tcp user=N/A group=N/A
ref="http://www.fortinet.com/ids/VID100663396" count=51384492 msg="anomaly:
tcp_syn_flood, 16 > threshold 5, repeats 591 times"
```

Figure 6.27. IPS type event

The next step was to identify squid proxy events stored in the SIEM. This was performed by filtering for events of "*process=squid3*". Figure 6.28 shows a sample squid proxy event message.

```
Apr 7 18:37:16 172.16.16.18 Apr 7 18:34:28 mondemo squid3: 1396852468.600 60003
192.168.2.105 TCP_MISS/000 0 GET
http://www.facebook.com/plugins/share_button.php? - DIRECT/31.13.70.81 -
```

Figure 6.28. Sample squid event message

Finally in order to correlate the IPS events from the firewall and the squid proxy traffic on the destination IP address, it was important that destination IP address from both event types be available in the event messages. Once the IPS and squid events had been filtered out and the destination IP address confirmed as available for logical processing, the final correlation logic was defined as shown in Figure 6.29.

```
eventtype="IPSEvents" severity=critical | JOIN type=inner dst [SEARCH
eventtype="ip_checksquid"] | top 20 dst
```

Figure 6.29. Scenario 1 correlation rule

Table 6.5 shows the results of applying the correlation logic.

IP Address	Count
172.16.16.18	377
203.167.141.138	202
203.167.141.153	164
68.232.44.121	139
203.167.141.155	127
203.167.141.146	113
31.13.70.81	76
203.167.141.139	69
203.167.141.182	67
203.97.86.218	65
203.167.141.137	58
203.167.141.145	46
68.232.44.111	41
31.13.70.17	41
203.167.141.160	41
202.89.45.50	40
203.167.141.152	39
203.167.141.154	36
74.125.237.203	35
74.125.237.171	33

Table 6.5. Top 20 destination IP addresses triggering signatures with a severity level of critical

Through this scenario, it was observed to be possible to correlate events from two log sources extracting data that would be critical for security threat monitoring. This extension of this form of correlation to multiple log sources was observed to potentially provide important threat related data to the information security professional. Section 6.3 expounds on this observation.

Threat monitoring correlation scenario 2

Correlation of IPS events generated by the firewall IPS module with that of traffic coming through the squid proxy to determine which top 20 signatures are triggered and to determine their severity

Building from the threat monitoring correlation in scenario 1, the modified correlation rule in Figure 6.30 was defined and applied to the data in the SIEM. The results are shown in Table 6.6.

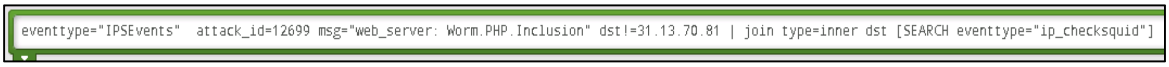
```
eventtype="IPSEvents" | join type=inner dst [SEARCH eventtype="ip_checksquid"] |
top 20 attack_id msg severity
```

Figure 6.30. Scenario 2 Correlation rule

attack_id	Msg	severity	count	percent
12699	web_server: Worm.PHP.Inclusion	high	9723	42.753496
11570	web_server: Apache.CGI.Byterange.Request.DoS	medium	7415	32.604872
107347981	http_decoder: HTTP.Unknown.Tunnelling	info	1468	6.455017
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 2 times	critical	527	2.317298
100663396	anomaly: tcp_syn_flood, 6 > threshold 5	critical	408	1.794037
12699	web_server: Worm.PHP.Inclusion, repeated 2 times	high	388	1.706094
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 3 times	critical	240	1.055316
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 4 times	critical	222	0.976167
12699	web_server: Worm.PHP.Inclusion, repeated 4 times	high	159	0.699147
12699	web_server: Worm.PHP.Inclusion, repeated 3 times	high	158	0.69475
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 5 times	critical	146	0.641984
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 6 times	critical	140	0.615601
12240	web_server: LongSlash	medium	110	0.483687
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 8 times	critical	95	0.417729
12699	web_server: Worm.PHP.Inclusion, repeated 5 times	high	87	0.382552
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 7 times	critical	77	0.338581
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 9 times	critical	69	0.303403
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 10 times	critical	66	0.290212
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 11 times	critical	46	0.202269
100663396	anomaly: tcp_syn_flood, 6 > threshold 5, repeats 15 times	critical	43	0.189077

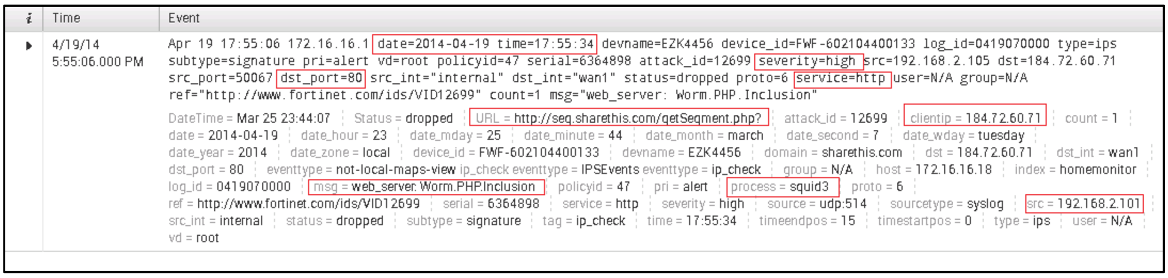
Table 6.6. Top 20 signatures triggered and their severity

From data in Table 6.6, it was observed that through correlation it was possible to determine which signatures were triggered the most and to determine the severity of the signatures. From the data in Table 6.6, it was observed that analysing the events further using the top-down log analysis approach outlined in Chapter 4, provided further important information regarding the threat. For instance drilling down into signature the *web_server: Worm.PHP.Inclusion* as shown Figure 6.31 returned important correlated security information as shown in the sample event in Figure 6.32.



```
eventtype="IPSEvents"  attack_id=12699 msg="web_server: Worm.PHP.Inclusion" dst!=31.13.70.81 | join type=inner dst [SEARCH eventtype="ip_checksquid"]
```

Figure 6.31. Correlation results drill down



```
4/19/14
5:55:06.000 PM

Apr 19 17:55:06 172.16.16.1 date=2014-04-19 time=17:55:34 devname=EZK4456 device_id=FWF-602104400133 log_id=0419070000 type=ips
subtype=signature pri=alert vd=root policyid=47 serial=6364898 attack_id=12699 severity=high src=192.168.2.105 dst=184.72.60.71
src_port=50067 dst_port=80 src_int="internal" dst_int="wan1" status=dropped proto=6 service=http user=N/A group=N/A
ref="http://www.fortinet.com/ids/VID12699" count=1 msg="web_server: Worm.PHP.Inclusion"

DateTime = Mar 25 23:44:07 | Status = dropped | URL = http://seq.sharethis.com/getSegment.php? | attack_id = 12699 | clientip = 184.72.60.71 | count = 1
date = 2014-04-19 | date_hour = 23 | date_mday = 25 | date_minute = 44 | date_month = march | date_second = 7 | date_wday = tuesday
date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | domain = sharethis.com | dst = 184.72.60.71 | dst_int = wan1
dst_port = 80 | eventtype = not-local-maps-view ip_check eventtype = IPSEvents eventtype = ip_check | group = N/A | host = 172.16.16.18 | index = homemonitor
log_id = 0419070000 | msg = web_server: Worm.PHP.Inclusion | policyid = 47 | pri = alert | process = squid3 | proto = 6
ref = http://www.fortinet.com/ids/VID12699 | serial = 6364898 | service = http | severity = high | source = udp:514 | sourcetype = syslog | src = 192.168.2.101
src_int = internal | status = dropped | subtype = signature | tag = ip_check | time = 17:55:34 | timeendpos = 15 | timestartpos = 0 | type = ips | user = N/A
vd = root
```

From Figure 6.32, important pieces of information that were observed to have been identified through this correlation included (the actual data element field name is in brackets):

- The date and time of the event (*date*)
- The severity of the event (*severity*)
- The destination port (*port*)
- The service (*service*)
- The URL (*URL*)
- The correlated log sources (*process=squid3 and eventtype=IPSevents*)
- The source ip address (*src*)
- The destination address (*dst*) and finally
- The signature name (*msg*)

This observation is expounded on in section 6.3.

Threat monitoring correlation scenario 3

> *Determination of the threat score of the web traffic which triggered an IDS event having a severity level of critical.*

Again building on correlation scenarios 1 and 2, the correlation rule in Figure 6.32 was applied to the data in the SIEM lab.

```
eventtype="IPSEvents" severity=critical | JOIN type=inner dst [SEARCH
eventtype="ip_checksquid"] | lookup threatscore clientip | search threatscore>0 | dedup
clientip | table clientip msg threatscore
```

Figure 6.32. Scenario 3 correlation rule

A single result was returned for the correlation, for an IP address with a threatscore of 20. The trigger was for the signature “anomaly: tcp_syn_flood, 6 > threshold 5, repeats 26 times".

6.3 Discussion: SIEM in IT Security Observations

The analysis performed not only in section 6.2 but throughout the period of monitoring events collected in the SIEM, enabled this researcher through experimentation while leveraging SIEM use cases to experience the role of SIEM in IT security. Most importantly it enabled this researcher in conjunction with the review of literature to address the research questions set forth for this research. The tasks performed in sections 6.2.1, 6.2.2 and 6.2.4 though seemingly simple, required a significant level of preparatory work. For instance, getting IP reputation to work with the SIEM, determining, finding a reasonable number of malicious URLs that could be used for generating threat traffic and ensuring that the threat activity did not negatively impact other log sources in the lab. The determination of the final correlation logic also required a reasonable amount of work.

Though on a much smaller scale to a real SIEM deployment; which in some instances could involve thousands of log sources and equally a significant number of information security professional analysing events collected by the SIEM; the experiment performed in this research allowed this researcher to explore SIEM gaining valuable foundational SIEM in IT security experience and understanding. The analysis of events was performed using the top-down log analysis technique presented in section 4.4 of Chapter 4.

In this section is a summary of the observations noted from the main experimental tasks performed in section 6.2 as well as from the lab setup tasks that were performed in section 5.2 and section 5.3.

The key observations were:

- SIEM can enhance threat monitoring in IT security
- SIEM can aid in meeting relevant compliance requirements
- SIEM correlation capability can enhance threat monitoring in IT security
- SIEMs capability of handling events the variety of event log formats is essential in IT security and can enhance security monitoring.
- The more events are collected from multiple log sources, the more:
    - threat monitoring through correlation is enhanced
    - opportunities for identifying a wider range of security related SIEM use cases
    - difficult, complex, time and resource consuming security event analysis becomes.

<u>SIEM can enhance threat monitoring in IT security</u>

It was observed that the use of SIEM in IT security can enhance the security professional's ability to monitor security related threats. In outbound traffic threat monitoring for instance, this researcher was able to observe traffic destined for external destinations with high risk rating based on IP reputation. While this traffic was manually generated and only from a few log sources, the availability of such event data in a high event volume, centralised SIEM integrated with threat intelligence services like IP reputation would give a security professional valuable data to monitor security threats on the network. In a research by Jeong, Kim, Kim & So (2011) into botnet detection, the researchers outline botnet detection methods similarly based on the monitoring of outbound traffic to potential botnet command and control centres. It was observed that monitoring of outbound traffic for threats enrichment and correlated with for example user related event data would give the security professional a clearer view both the internal source of the suspicious traffic and the destination of the traffic.

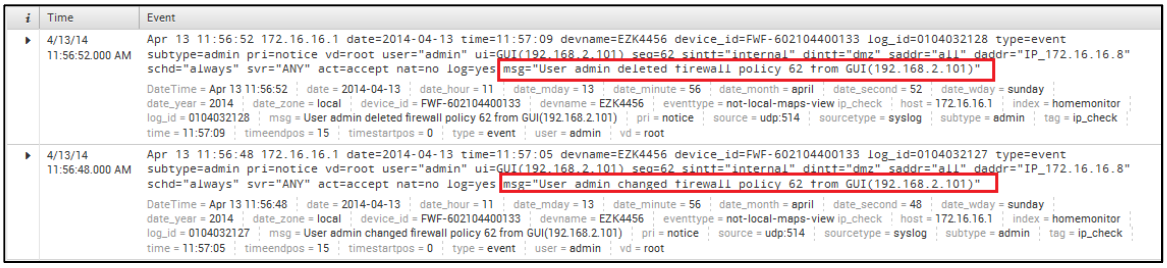
Insider threats are increasingly recognised as equally destructive as external security threats (Grimaila, Myers & Mills, 2011) and therefore the need to monitor user activity as outlined in section 6.2.2. It was observed that using the SIEM allowed for easier monitoring of user related events in spite of the differences in structure and format of events coming from the various log sources. The use of the SIEM allowed this researcher to handle the varying log formats into some form of common format that could then be easily analysed. The correlation aspect as observed in section 6.3.4 was seen to enhance security related event analysis allowing the observation of relationships in events from multiple log sources. In an environment where there exist a large number of log sources, the correlation of events that might be part of a composite security threat can improve threat detection while enabling the reduction of the false positives that have been a challenge in security systems such as IDSs (Edmundo, Goldenstein, Mauro & Stroeh, 2013).

SIEM can aid in meeting some compliance requirements

While only a subset of PCI DSS related compliance requirements were considered in section 6.3.3 for understanding compliance and SIEM, it was observed that SIEM can aid in meeting relevant compliance requirements. A similar observation was made by Chuvakin, Phillips & Schmidt (2013) where the authors discuss logging and compliance based on the sub requirements in mandate 10 of the PCI DSS standard. The importance of ensuring that events logs sent to the SIEM contained required basic data elements was a key observation; in particular when these events logs are collected within the context of regulatory compliance. These data elements include, user details for instance user name, client source IP address, destination IP address, data and time, severity of the message and a description of the message.

Another observation from SIEM use case 2 was that of SIEM enhancing user activity monitoring for both the general user and privileged user accounts. Again while user related events came in varying formats, it was possible using the SIEM to parse these events into some form of common format. The normalisation SIEM capability was observed as enhancing log analysis as highlighted in section 6.2.2 with the extraction of user data into the user data field.

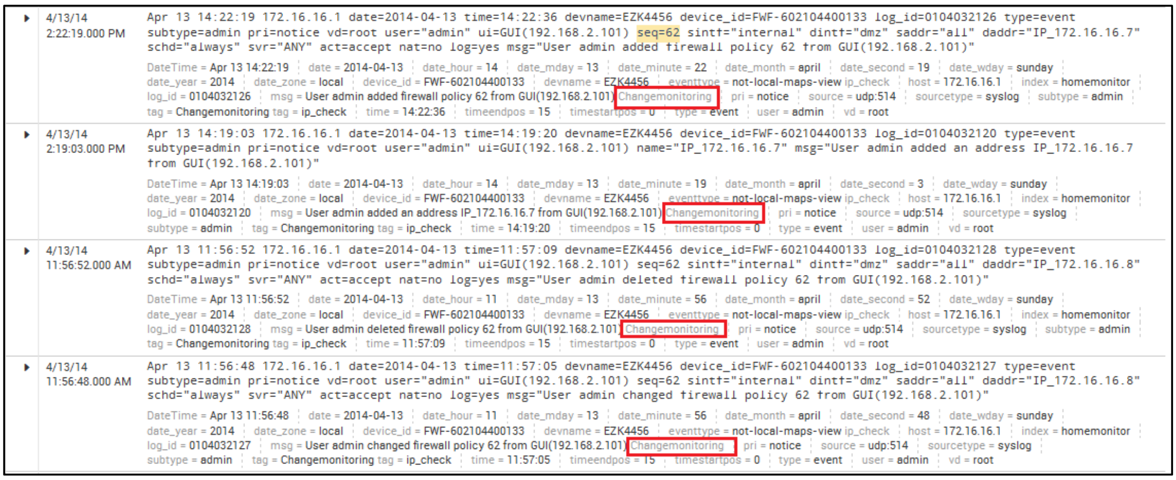
Data contextualization is designed to add meaning to events in centralized storage. It was observed that using the SIEM, it was possible to add context to events thus enhancing security relating monitoring. For instance, consideration was given of the change monitoring requirement in PCI DSS where it is required that user configuration changes be closely monitored. Figure 6.32 shows 2 change related monitored events logged in the SIEM.



```
i  Time                 Event
▸  4/13/14              Apr 13 11:56:52 172.16.16.1 date=2014-04-13 time=11:57:09 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032128 type=event
   11:56:52.000 AM      subtype=admin pri=notice vd=root user="admin" ui=GUI(192.168.2.101) seq=62 sintf="internal" dintf="dmz" saddr="all" daddr="IP_172.16.16.8"
                        schd="always" svr="ANY" act=accept nat=no log=yes msg="User admin deleted firewall policy 62 from GUI(192.168.2.101)"
                        DateTime = Apr 13 11:56:52 | date = 2014-04-13 | date_hour = 11 | date_mday = 13 | date_minute = 56 | date_month = april | date_second = 52 | date_wday = sunday |
                        date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | eventtype = not-local-maps-view ip_check | host = 172.16.16.1 | index = homemonitor |
                        log_id = 0104032128 | msg = User admin deleted firewall policy 62 from GUI(192.168.2.101) | pri = notice | source = udp:514 | sourcetype = syslog | subtype = admin | tag = ip_check |
                        time = 11:57:09 | timeendpos = 15 | timestartpos = 0 | type = event | user = admin | vd = root
▸  4/13/14              Apr 13 11:56:48 172.16.16.1 date=2014-04-13 time=11:57:05 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032127 type=event
   11:56:48.000 AM      subtype=admin pri=notice vd=root user="admin" ui=GUI(192.168.2.101) seq=62 sintf="internal" dintf="dmz" saddr="all" daddr="IP_172.16.16.8"
                        schd="always" svr="ANY" act=accept nat=no log=yes msg="User admin changed firewall policy 62 from GUI(192.168.2.101)"
                        DateTime = Apr 13 11:56:48 | date = 2014-04-13 | date_hour = 11 | date_mday = 13 | date_minute = 56 | date_month = april | date_second = 48 | date_wday = sunday |
                        date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | eventtype = not-local-maps-view ip_check | host = 172.16.16.1 | index = homemonitor |
                        log_id = 0104032127 | msg = User admin changed firewall policy 62 from GUI(192.168.2.101) | pri = notice | source = udp:514 | sourcetype = syslog | subtype = admin | tag = ip_check |
                        time = 11:57:05 | timeendpos = 15 | timestartpos = 0 | type = event | user = admin | vd = root
```

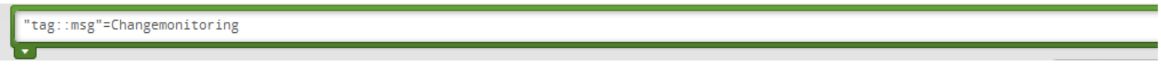
Figure 6.33. Change related events logged in the SIEM

The event messages while descriptive of the actual action performed could be further enriched by providing context to the data. Using the tagging feature of the Splunk SIEM, all change related events could be tagged with say a "*changemonitoring*" string. The result of this enrichment would be as illustrated in Figure 6.33 where change related events have been tagged with the "*changemonitoring*" string enabling log analysis of change related events by simply searching for events tagged with the "*changemonitoring*" string as illustrated in Figure 6.34.



```
4/13/14             Apr 13 14:22:19 172.16.16.1 date=2014-04-13 time=14:22:36 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032126 type=event
2:22:19.000 PM      subtype=admin pri=notice vd=root user="admin" ui=GUI(192.168.2.101) seq=62 sintf="internal" dintf="dmz" saddr="all" daddr="IP_172.16.16.7"
                    schd="always" svr="ANY" act=accept nat=no log=yes msg="User admin added firewall policy 62 from GUI(192.168.2.101)"
                    DateTime = Apr 13 14:22:19 | date = 2014-04-13 | date_hour = 14 | date_mday = 13 | date_minute = 22 | date_month = april | date_second = 19 | date_wday = sunday |
                    date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | eventtype = not-local-maps-view ip_check | host = 172.16.16.1 | index = homemonitor |
                    log_id = 0104032126 | msg = User admin added firewall policy 62 from GUI(192.168.2.101) Changemonitoring | pri = notice | source = udp:514 | sourcetype = syslog | subtype = admin |
                    tag = Changemonitoring tag = ip_check | time = 14:22:36 | timeendpos = 15 | timestartpos = 0 | type = event | user = admin | vd = root

4/13/14             Apr 13 14:19:03 172.16.16.1 date=2014-04-13 time=14:19:20 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032120 type=event
2:19:03.000 PM      subtype=admin pri=notice vd=root user="admin" ui=GUI(192.168.2.101) name="IP_172.16.16.7" msg="User admin added an address IP_172.16.16.7
                    from GUI(192.168.2.101)"
                    DateTime = Apr 13 14:19:03 | date = 2014-04-13 | date_hour = 14 | date_mday = 13 | date_minute = 19 | date_month = april | date_second = 3 | date_wday = sunday |
                    date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | eventtype = not-local-maps-view ip_check | host = 172.16.16.1 | index = homemonitor |
                    log_id = 0104032120 | msg = User admin added an address IP_172.16.16.7 from GUI(192.168.2.101) Changemonitoring | pri = notice | source = udp:514 | sourcetype = syslog |
                    subtype = admin | tag = Changemonitoring tag = ip_check | time = 14:19:20 | timeendpos = 15 | timestartpos = 0 | type = event | user = admin | vd = root

4/13/14             Apr 13 11:56:52 172.16.16.1 date=2014-04-13 time=11:57:09 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032128 type=event
11:56:52.000 AM     subtype=admin pri=notice vd=root user="admin" ui=GUI(192.168.2.101) seq=62 sintf="internal" dintf="dmz" saddr="all" daddr="IP_172.16.16.8"
                    schd="always" svr="ANY" act=accept nat=no log=yes msg="User admin deleted firewall policy 62 from GUI(192.168.2.101)"
                    DateTime = Apr 13 11:56:52 | date = 2014-04-13 | date_hour = 11 | date_mday = 13 | date_minute = 56 | date_month = april | date_second = 52 | date_wday = sunday |
                    date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | eventtype = not-local-maps-view ip_check | host = 172.16.16.1 | index = homemonitor |
                    log_id = 0104032128 | msg = User admin deleted firewall policy 62 from GUI(192.168.2.101) Changemonitoring | pri = notice | source = udp:514 | sourcetype = syslog | subtype = admin |
                    tag = Changemonitoring tag = ip_check | time = 11:57:09 | timeendpos = 15 | timestartpos = 0 | type = event | user = admin | vd = root

4/13/14             Apr 13 11:56:48 172.16.16.1 date=2014-04-13 time=11:57:05 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032127 type=event
11:56:48.000 AM     subtype=admin pri=notice vd=root user="admin" ui=GUI(192.168.2.101) seq=62 sintf="internal" dintf="dmz" saddr="all" daddr="IP_172.16.16.8"
                    schd="always" svr="ANY" act=accept nat=no log=yes msg="User admin changed firewall policy 62 from GUI(192.168.2.101)"
                    DateTime = Apr 13 11:56:48 | date = 2014-04-13 | date_hour = 11 | date_mday = 13 | date_minute = 56 | date_month = april | date_second = 48 | date_wday = sunday |
                    date_year = 2014 | date_zone = local | device_id = FWF-602104400133 | devname = EZK4456 | eventtype = not-local-maps-view ip_check | host = 172.16.16.1 | index = homemonitor |
                    log_id = 0104032127 | msg = User admin changed firewall policy 62 from GUI(192.168.2.101) Changemonitoring | pri = notice | source = udp:514 | sourcetype = syslog |
                    subtype = admin | tag = Changemonitoring tag = ip_check | time = 11:57:05 | timeendpos = 15 | timestartpos = 0 | type = event | user = admin | vd = root
```

Figure 6.34. Enriched change monitoring related events



```
"tag::msg"=Changemonitoring
```

Figure 6.35. Search for enriched change monitoring related events

SIEM correlation capability can enhance threat monitoring in IT security

In section 6.2.4, a correlation use case was explored. Through correlation it was possible to determine the top 20 destination IP addresses triggering signatures with a severity level of critical, the top 20 signatures triggered and to their severity and the threat score of web traffic which triggered an IDS event having a severity level of critical. It was observed that through top-down log analysis, the researcher was able to gather important information regarding a particular threat. The determination of such information is important in IT security. The observation was that event analysis by correlating data from multiple sources enhances the understanding of the security status of an IT security environment. An extension of the threat monitoring scenarios in an environment with a significant number of log sources would be for instance using correlation to then determine the internal source usernames and IP addresses of the suspicious traffic.

6.3.4 The significant of volume of events collected from multiple log sources

According to Grimaila, Myers & Mills (2011), the possibility of detecting security threats is enhanced as the number of log sources increases. While the experiment involved only a few log sources, the volume of events collected in the SIEM lab from the few sources did highlight the potential for an extended list of security related SIEM use case that could have been realised. Other security related SIEM use cases that could have been derived given relevant data included:

- Network port, service and protocol monitoring
- Monitoring of events for potential data loses
- Generic user account monitoring
- Unauthorised application access

In spite of the benefits realised by the increase in volume of events and the number of log sources, it was observed that as the number of events and log sources increased, the variety of event formats also increased and therefore the more difficult it was to analyse the events to identify security threats, an observation also made by Ganapathi, Oliner & Xu (2012).

SIEM and varying event log formats

Finally the lack of event message format standards has been was addressed in Chapter 3 and frequently highlighted in parts of this thesis. The lack of event message standards was seen to have an impact the analysis of related events. Events could be missed out during log analysis as a result of their format being different from other related event messages. For instance Figure 6.35 and Figure 6.36 illustrate the varying format in user related event messages. The events in Figure 6.35 were returned from using the search string "*user user=!N/A*"; returning any event containing the string "*user*". The result did not however include any user related events which did not contain the string "*user*". This was the case with the ftp login events shown in Figure 6.35.

```
Apr 13 00:35:33 192.168.2.101 Apr 13 00:37:35 mypc selfserviceplugin[info] 1 Self-service Plug-in started (user=mypc\chiko)

Apr 12 18:32:35 172.16.16.1 date=2014-04-12 time=18:32:51 devname=EZK4456 device_id=FWF-602104400133 log_id=0104032003 type=event subtype=admin pri=information vd=root user="admin" ui=https(192.168.2.101) action=logout status=success reason=timeout msg="Administrator admin timed out on https(192.168.2.101)"

Apr 7 23:40:53 172.16.16.18 Apr 7 23:38:04 mondemo sshd[22707]: PAM 5 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=116.10.191.230 user=root

Apr 6 14:51:23 172.16.16.18 Apr 6 14:48:37 mondemo perl: pam_unix(webmin:auth): authentication failure; logname= uid=0 euid=0 tty= ruser= rhost= user=root
```

Figure 6.36: User activity related event messages

```
Apr 13 10:50:05 172.16.16.19 Apr 13 10:50:08 mics.smcism.local GenericLog   0(000001)
4/13/2014 10:50:05 AM - filezillalogin (192.168.2.101)> 226 Transfer OK
```

Figure 6.37. Filezilla user related event logs

Using the SIEM, however, it was possible to normalise the event from the ftp server and therefore enable this researcher to include the ftp events user activity related reporting.

6.4 Contribution and Recommendation

This research sought to answer the following research questions:

1. Are SIEM technologies applicable to IT security?
2. Does SIEM technology enhance the ability to monitor and respond to IT security incidences in an environment comprising a significant high volume of application, network and device system logs?

The review of literature highlighted the changing IT security landscape and evolving IT security threats. The review of literature also showed the growing importance of log management in IT security either for detecting IT security threats or for compliance purposes. The review of literature also revealed the increasing transition to SIEM for event collection, storage and analysis as a result of SIEM's enhanced capabilities compared to the limitations of log management. These enhanced SIEM capabilities were noted as filtering, normalisation, aggregation, correlation, alerting and reporting. Through the identification of SIEM use cases and experimentation in a SIEM lab setup, key observations made were:

- SIEM can enhance threat monitoring in IT security
- SIEM can aid in meeting some compliance requirements
- SIEM correlation capability can enhance threat monitoring in IT security
- SIEMs capability of handling events in difference formats is essential in IT security
- The more events are collected from multiple log sources, the more:
    - threat monitoring through correlation is enhanced
    - opportunities for identifying a wider range of security related SIEM use cases
    - difficult, complex and resource consuming security event analysis becomes

Are SIEM technologies therefore applicable to IT security? And does SIEM technology enhance the ability to monitor and respond to IT security incidences in an environment comprising a significant high volume of application, network and device system logs. The review of literature and observations noted through experimentation point to SIEM being relevant or applicable to IT security and that SIEM can enhance the ability to monitor and respond to IT security threats and incidences in an environment comprising of a high volume of security events.

In keeping with the recognition of the importance of SIEM use cases and their application in SIEM deployments, 4 SIEM uses case were identified and used in this research. While the exploration using the 4 SIEM use cases produced valuable observations with regard to SIEM, the recommendation would be for the investigation of SIEM in IT security using an extended list of SIEM use cases. This would allow the researcher to gain a broader perception of SIEM in IT security as a result of the potential range of IT security scenarios, an extended list use case would avail.

6.5 Future Work

Building on the foundation and understanding gained from this research, future work is expected to explore the applicability of SIEM to IT security from the perspective or a real SIEM deployment. As observed in Chapter 1, literature reviewed shows that most SIEM related research has been conducted using the experimental methodology and are largely lab based. Opportunities therefore exist of investigating SIEM and IT security within the context of a real SIEM deployment employing either a survey based research methodology or a case study based approach. Grimaila, Myers, Mills & Peterson (2011) state that “while SIEM has proven to be an effective means of detecting attacks against organizational ICT resources”, organisations have faced challenges implementing and leveraging SIEM in IT security. This researcher is therefore keen as part of future work to explore the applicability of SIEM in IT security in a real SIEM deploying as well as investigating the challenges that organisations face in deploying SIEM.

7. Conclusion

7.1 Summary and Conclusion

This research sought to answer the following research questions regarding SIEM in IT security:

1. Are SIEM technologies applicable to IT security?
2. Does SIEM technology enhance the ability to monitor and respond to IT security incidences in an environment comprising a significant high volume of application, network and device system logs?

The review of literature in Chapter 2 and 3 provided relevant background to the discussion and literature review of SIEM in Chapter 4. Of note in Chapter 2 and 3 was the evolution of the IT security landscape and the increasing recognition of event logs in the multi-dimensional approach to IT security. Chapter 4 addressed SIEM as a successor to log management due to its enhanced features, features relevant to addressing the increasing demands placed by regulatory compliance and the dynamic nature of security threats. The experimental actions performed and described in Chapter 5 and 6 highlighted the important role of SIEM use cases as also pointed out in literature. The observations drawn from Chapter 5 and Chapter 6, point to SIEM as having relevancy in IT security as a resulted of its filtering, normalisation, aggregation, correlation, alerting and reporting capabilities. This study therefore concludes that rightly deployed, with careful consideration and identification of SIEM use cases opportunities exist for SIEM in IT security.

References

Aguirre, I. & Alonso, S. (2012). Improving the Automation of Security Information Management: A Collaborative Approach. *IEEE Security&Privacy Magazine*. 8(1). 55-59

Álvareza, G. & Petrovic´, S. (2003). A new taxonomy of Web attacks suitable for efficient encoding. *Computers & Security*. 22(5). 435-449

Anderson, J.M. (2003). Why we need a new definition of information security. *Computers & Security*. 22(4). 308-313

Arasteh, A.R, Debbabi, M., Sakha, A. & Saleh, M. (2007). Analyzing multiple logs for forensic evidence. *Digital Investigation*. Volume 4. 82-91

Avolio, M. (1998). A Multidimensional Approach to Internet Security. *ACM netWorker*. Volume 2.2.

Azzedine et.al (2007). An agent based and biological inspired real-time intrusion detection and security model for computer network operations. *Computer Communications*. 30(3). 2649 – 2660

Barwinski (2005) Taxonomy of spyware and empirical study of network drive-by-download. Unpublished doctoral dissertation

Bishop, M. (2003). Computer Security, Art and Science. Boston, MA: Addison-Wesley

Blaich, A., Li, Z., Liao, Q., & Striegel, A. (2010). *Fighting botnets with economic uncertainty. Security Comm*. Networks, 4(10). 1104–1113

Botta, D. et.al. (2008). *Towards Understanding IT Security Professionals and Their Tools*. Presented at Symposium On Usable Privacy and Security (SOUPS). Pittsburgh, PA.

Brewer, R. 2012. Protecting critical control systems. Network Security. 2012(3). 7

Burd, S.D., Jackson, R.B. & Satzinger, J.W. (2004). *Systems Analysis and Design in a Changing World*. Boston, MA: Course Technology Press

Casey, D. (2008). Turning Log Files Into a Security Asset. *Network Security*. 2008(2). 4-7

CEE. Common Event Expression: Architecture Overview. Retrieved April 11, 2013, from https://cee.mitre.org/docs/CEE_Architecture_Overview-v0.5.pdf

Chapple, M., Stewart, M.J. & Tittel, E. (2008). CISSP: Certified Information Security Professional Study Guide. Indianapolis, Indiana: Sybex.

Chen, X., Fan, L., Li, J. & Zhang, S. (2008). Building network attack graph for alert causal correlation. Computers & Security. 27(5). 188-196

Chickowski, E. (2012). 10 Ways To Fail A PCI Audit. Retrieved April 15, 2014 from http://www.darkreading.com/10-ways-to-fail-a-pci-audit/d/d-id/1138159

Cichonski, P., Grance, T., Millar, T. & Scarfone, K. (2012). Computer Security Incident Handling Guide. Retrieved March 30, 2014, from http://csrc.nist.gov/publications/nistpubs/800-61rev2/SP800-61rev2.pdf

Chuvakin, A. (2010). SIEM: Moving Beyond Compliance. Retrieved November 11, 2013, from http://www.securitywarriorconsulting.com/pdfs/chuvakin_RSA_2010_SEIMBC_WP_0810.pdf

Chuvakin, A.A, Phillips, C. & Schmidt, K.J. (2013). *Logging and Log Management: The Authoritative Guide to Understanding the Concepts Surrounding Logging and Log Management*. Waltham, MA: Elsevia.

Computer Security Institute. (2001). CSI/FBI 2000 Computer Crime and

Security Survey. Retrieved September 9, 2012, from http://www-tc.pbs.org/wgbh/pages/frontline/shows/hackers/risks/csi-fbi2000.pdf

Coppolino, L., D'Antonio, S., Formicola, V. & Romano, L. (2011). Integration of a System for Critical Infrastructure Protection with the OSSIM SIEM Platform: A dam case study. Computer Safety, Reliability, and Security, Lecture Notes in Computer Science. Volume 6894. 199-212

Daya, B. *Network security: History, importance, and future*. University of Florida Department of Electrical and Computer Engineering, unknown

Dobbins, G.H., Lane, I.M. & Steiner, D. D. (1988). A Note on the Role of Laboratory Methodologies in Applied Behavioural Research: Don't Thow out the Baby with the Bath Water. *Journal of Organisational Behaviour*. 9. 281-286

Dostal, O., Javornik, M., Ledvinka, J. & Slavicek, K. (2008). *Mathematical Processing of Syslog Messages from Routers and Switches*. International Conference on Information and Automation for Sustainability. Colombo

D' Ovidio, R. (2007). The Evolution of Computers and Crime: Complicating Security Practice. *Security Journal*. 20 (1). 45-49

Doug, H. & Kevin, P. (2011). *Security 2020: Reduce security risks this decade*. Indianapolis, IN : Wiley Pub.

Edmundo, R., Goldenstein, S.K., Mauro, M. & Stroeh, K. (2013). An approach to the correlation of security events based on machine learning techniques. *Journal of Internet Services and Applications*. 4(1). 1-16

Forte, V. (2004). The "ART" of log correlation: Part 1. *Computer Fraud & Security*. 2004(6). 7-11

Gabriel, R., Hoppe, T., Pastwa, A. & Sowa, S. (2009). *Analyzing Malware Log Data to Support Security Information and Event Management: Some Research Results*. First International Conference on Advances in Databases, Knowledge, and Data Applications.

Ganapathi, A., Oliner, A. & Xu, W. (2012). Advances and Challenges in Log Analysis. *Communications of the ACM*, 55 (2), 55-61.

Gharaee, H., Madani, A. & Rezayi, S. (2011). *Log Management comprehensive architecture in Security Operation Center(SOC)*. 2011 International Conference on Computational Aspects of Social Networks (CASoN). Salamanca, Spain

Grigorescu, V. & Becker, D. (Unknown). Community:SingleIndexServerDeploymentModels. Retrieved November 12, 2013 from http://www.educause.edu/events/security-professionals-conference/2013/how-advanced-log-management-can-trump-siem-tales-woe-and-glory

Grimaila, M, R., Myers, J. & Mills, R, F. (2011). *Log-Based Distributed Security Event Detection Using Simple Event Correlator*. Proceedings of the 44th Hawaii International Conference on System Sciences, Hawaii

Grimaila, M, R., Myers, J. & Mills, R, F. (2009). *Towards Insider Threat Detection using Web Server Logs*. Proceedings of the 5th Annual Workshop on cyber security and information intelligence research. Oak Ridge, Tennessee.

Grimaila, M, R., Myers, J., Mills, R. F. & Peterson, G. (2011). Design and Analysis of a Dynamically Configured Log-dased Distribution Security Event Detector. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*. 9(3). 219-241

Gordon, L.A., Loeb, M.P & Zhou, L.(2011). The impact of information security breaches: Has there been a downward shift in costs?. *Journal of Computer Security*. 19(1). 35-36.

Gorge, M. (2007). Making sense of log management for security purposes – an approach to best practice log collection, analysis and management. *Computer Fraud & Security*. 2007(5). 5-10

Groom, P.D. (2003). The IT Security Model. *IEEE Potentials*. 22(4). 6-8

Hansman, S. & Hunt, R. (2005). A taxonomy of network and computer attacks. *Elsevier*. 24 (1). 31-33.

Hawkins, B.L & Oblinger, D.G (2006). The Myth about IT Security. *EDUCAUSE Review*. 41(3). 14

Hutchison, A (2009). Unlocking the Opportunity of Siem. Retrieved November 27, 2012 from http://searchsecurity.techtarget.com/magazineContent/Unlocking-the-opportunity-of-SIEM-technology

Hoefelmeyer, R. (2004). *Malicious code: The threat, detection and protection*. In H. F.Tipton & N. Krause (Eds.), Information security management handbook. Raton, FL: CRC Press.

Hoppe, T. Pastwa, A. & Sowa, S (2009). Business Intelligence Based Malware Log Data Analysis as an Instrument for Security Information and Event Management. International. *Journal on Advances in Security*. 2(2&3).

Jariwala, V.J. & Jinwala, D.C. (2009). *A Taxonomy of Security Attacks on the Advanced Encryption Standard*. Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human. Seoul, Korea.

Jenkins, S. (2011). Learning to love SIEM. Network Security. 2011(4). 18-19

Jeong, O., Kim, C., Kim W & So, J. (2011). Botnets: threats and responses. *International Journal of Web Information Systems*. 7(1). 6-17.

Jiang, J., Huang, J. & Zhang, M. (2012). *The design and implement of the centralized log gathering and analysis system*. International Conference on Computer Science and Automation.

Kelly, D. (2004). Security Management via SIM (Security Information Management)- A requirements perspective. *Journal of Network and Systems Management*. 12(1). 137-144

Kent, K. & Souppaya, M. (2006). Computer Security Log Management. Retrieved November 4, 2013, from http://csrc.nist.gov/publications/nistpubs/800-92/SP800-92.pdf

Kessler, G.C. (2004). Information Security: New Threats or Familiar Problems?. *Computer*. 45(2). 59-65

Kavanagn, K.M & Nicolett, M. (2011). Critical Capabilities of Security Information and Event Management Technology. Retrieved November 3, 2013, from https://www.gartner.com/doc/2477017/critical-capabilities-security-information-event

Kim, H.S., Turner, A. & Wong, T. (2007). *Automatic Discovery of Relationships Across Multiple Network Layers*. Proceedings of the 2007 SIGCOMM workshop on Internet network management . New York, NY

Khelafa, H. (2007). Prevention and Handling of Malicious Code in Hershey PA(Ed.), Advances in Enterprise Information Technology Security (pp 239-259)

Kliger, S., Mozes, E. & Yemini, S.A. (1996). High speed and robust event correlation. *Communications Magazine, IEEE* . 34(5). 82-90

Knight, E. (2010). Investigating digital fingerprints: advanced log analysis. *Network Security*. 2010(10). 17-20

Koike, H & Takada, T. (2002). *Tudumi: Information Visualization System for Monitoring and Auditing Computer Logs*. Proceedings of the Sixth International Conference on Information Visualisation.

Kotenko, I., Polubelova, O. & Saenko, I. (2012). The Ontological Approach for SIEM Data Repository Implementation. *Future Internet*. 5(3). 355-375.

Lambeth, A., et.al. (1998). Implementing a Generalized Tool for Network Monitoring. *Information Security Technical Report*,. 3(4). 53-56

Lampson, B.W. (2004). Computer Security in the Real World. *Computer*. 37(6). 37-46

Laurie, Ben. (2004). *Network Forensics. Queue*. 2(4). 50-56

Lee, J. & Xue, N. (1999). Analyzing User Requirements by Use Cases: A Goal-Driven Approach. *Software, IEEE* . 16(4). 92-101

Levin, D. (2009). The convergence of SIEM, log mgmt. *Network World*. 26(12). 21

Linic, J. (2007). Information Systems Modeling with Use Cases. Proceedings of the ITI 2007 29th Int. Conf. on Information Technology Interfaces. Cavtat, Croatia

Liu, G., Mok, A.K & Yang, E.J (2010). *Composite Events for Network Event Correlation*. Proceedings of the Sixth IFIP/IEEE International Symposium on Integrated Network Management, 1999. Distributed Management for the Networked Millennium. Boston, MA.

Lobo, C. (2012). Security Log Management. Network Security. 2003(11). 6-9

Lozito, K. (2011). Mitigating Risk: Analysis of Security Information and Event *Management International Journal of Business Intelligence Research*, 2(2), 67-75

Makanju, A., Milios, E. E. & Zincir-Heywood, A.N. (2009). *Clustering Event Logs Using Iterative Partitioning* . Proceedings of the 15th ACM SIGKDD international

Marty, R. (2011). *Cloud Application Logging for Forensics*. Proceedings of the 2011 ACM Symposium on Applied Computing. TaiChung, Taiwan.

Maruyama, Y & Yamanishi, K. (2005). *Dynamic Syslog Mining for Network Failure Monitoring*. Proceedings of the eleventh ACM SIGKDD international conference on knowledge discovery in data mining, Chicago, Illinois.

Misnomer. (2012a). Adopting SIEM – What you need to know?. Retrieved November 24, 2013 http://infosecnirvana.com/adopting-siem-what-you-need-to-know/

Misnomer. (2012b). SIEM Use Cases – What you need to know?. Retrieved February 6, 2014 http://infosecnirvana.com/siem-use-cases/

Moffatt, S. (2013). Security Analytics: Hype or Huge?. Retrieved April 15, 2014, from http://www.infosecisland.com/blogview/22891-Security-Analytics-Hype-or-Huge.html

Moore, C. (2004). *The Growing Trend of Government Involvement in IT Security*. Proceedings of the 1st annual conference on Information security curriculum development, New York, NY.

Nabil, H. (2009). *Decentralized log event correlation architecture*. Proceedings of the International Conference on management of emergent digital ecosystems, New York, NY.

Nagappan, M. (2010). *Analysis of execution log files*. Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering. Cape Town, South Africa.

OWASP. (2014). Category:OWASP Top Ten Project. Retrieved March 15, 2013, from https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

Park, S., Yuan, D. & Zhou, Y. (2012). *Characterizing Logging Practices in Open-Source Software*. Proceedings of the 34th International Conference on Software Engineering. Zurich, Switzerland.

Parkin, R.K. (1998). The importance of IT security. *Computer Fraud & Security*. 1998(3). 12-15

Pasquinucci, A. (2007). The difficult art of managing logs. *Computer Fraud & Security*, 2007(10), 5-7

PCI Security Standards Council. (2013). Payment Card Industry (PCI) Data Security Standard: Requirements and Security Assessment Procedures. Retrieved November 25, 2013, from https://www.pcisecuritystandards.org/documents/PCI_DSS_v3.pdf

Peltier, T. 2001. Information Security Risk Analysis. Boca Raton, FL: Auerbach Publications

Power, R. (2002) CSI/FBI computer crime and security survey. *Computer Security Issues & Trends 8*, 1–24

Project Honeypot (unknown). The Threat Rating. Retrieved March 26, 2014 from http://www.projecthoneypot.org/threat_info.php

Rothman, M. (2012). Understanding and Selecting SIEM/Log Management: Introduction. Retrieved February 12, 2014 https://securosis.com/blog/understanding-and-selecting-siem-log-management-introduction

Sadan, Z & Schwartz, D.G. (2004). Social network analysis for cluster-based IP spam reputation. *Information Management & Computer Security*. 20(4). 281-295

Schultz, B. (2010). Love It or Leave It. SC Magazine. 21(8). 28

Shenk, J. (2010). SANS Sixth Annual Log Management Survey Report. Retrieved November 11, 2013, from https://www.sans.org/reading-room/analysts-program/logmgtsurvey-2010

Shipley, G. (2008). Are SIEM and log management the same thing?. Retrieved November 12, 2013, from http://www.networkworld.com/reviews/2008/063008-test-siem-log-integration.html

Solms, R.V. (1999). Information security management: why standards are important. *Information Management & Computer Security*. 7(1). 50-58

Splunk. (2009). Community:SingleIndexServerDeploymentModels. Retrieved April 15, 2014 from https://wiki.splunk.com/Community:SingleIndexServerDeploymentModels

Splunk (2014). IP Reputation. Retrieved March 26, 2014 from http://apps.splunk.com/app/1457/

Stearley, J. (2004). *Towards informatic analysis of syslogs*. IEEE International Conference on Cluster Computing.

Oliner, A. & Stearley, J. (2007). *What Supercomputers Say: A Study of Five System Logs*. International Conference on Dependable Systems and Networks. Edinburgh

Stevens, M. (2006). UTM: Ones-stop protection. Network Security. 2006(2). 12-14.

Varandi, R. (2002). SEC - a lightweight event correlation tool. Proceedings of the IEEE Workshop on IP operation and management.

Verizon (2013). *Verizon: 2013 Data Breach Investigations Report*. Retrieved March 23, 2014, from http://www.verizonenterprise.com/DBIR/2013/

Voloudakis, J. (2006). The Continuing Evolution of Effective IT Security

Practices. EDUCAUSE Review. 41(5). 30

Whitman, M.E. (2003). Enemy at The Gate: Threats to Information Security. *Communications of The ACM*. 46(8). 91-95

Weißschuh, T. (2000). IT Security — An Historical. *Computer Fraud & Security*. 2000(9) 9-11

Appendices

Appendix 1: SIEM Log Source Types

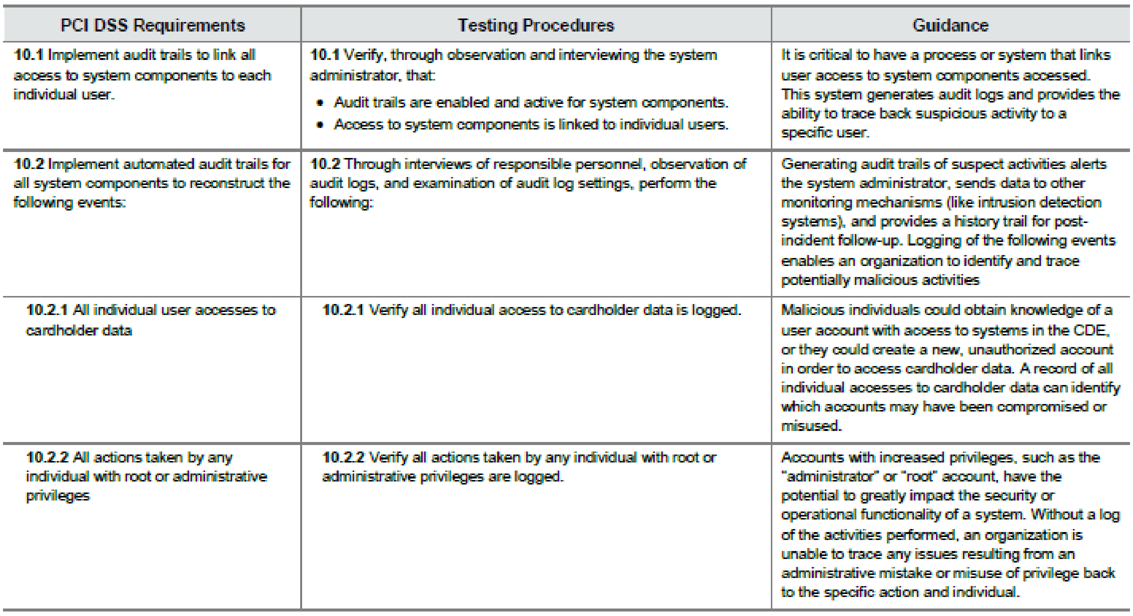
Log source type	Event count
syslog	68,589,047
WinEventLog:Security	14,412,127
linux_audit	1,560,087
Perfmon:Network Interface	975,731
webping	754,068
Perfmon:Available Memory	386,006
Perfmon:CPU Load	345,459
udev	121,561
freshclam.log-3	50,863
freshclam	28,847
freshclam.log-4	24,554
auth-too_small	17,204
dmesg	15,341
stash	10,926
initial-status-2	10,575
partman	10,219
WinEventLog:System	8,097
dpkg.log	7,155
freshclam.log-2	6,036
freshclam.log-6	3,775
freshclam.log-5	3,186
splunkd	2,970
clamav.log-2	2,710
clamav-too_small	2,543
status	2,156
Perfmon:Free Disk Space	2,148
sendmail_syslog	2,052
WinEventLog:Application	1,618
apache_error	1,427
ActiveDirectory	1,422
access.log-too_small	1,380
ureadahead.log-2	1,351
cache.log-3	1,188
ureadahead.log-3	1,162
freshclam-too_small	964
linux_bootlog	961
cache.log	924
clamav.log-3	905
ureadahead-too_small	713
clamav.log-4	451
mysqld	451

postgresql-9.1-main.log-too_small	439
ureadahead-7	417
dpkg-too_small	399
syslog-2	305
ureadahead-9	254
clamav.log-5	252
ureadahead-8	252
access-too_small	226
clamav-7	226
term-too_small	215
procps-static-network-up.log-too_small	180
mail-too_small	173
procps-virtual-filesystems.log-too_small	162
postgresql-9.1-main-too_small	157
splunk_web_service	119
alternatives.log	112
procps-virtual-filesystems-too_small	90
history.log-too_small	80
hardware-summary	79
mail.log-too_small	72
clamav.log-6	67
report-too_small	62
ureadahead.log-too_small	45
console-setup.log-too_small	44
term.log-2	37
postgresql-9.1-main-3	36
error-2	32
proftpd.log-too_small	32
rsyslog.log-too_small	32
clamav.log-too_small	29
container-detect.log-too_small	28
lastcommlog-too_small	28
vsftpd.log-too_small	26
console-setup-too_small	22
xferlog-too_small	21
history-too_small	16
network-interface-eth0.log-too_small	16
container-detect-too_small	15
mail.err-too_small	15
rsyslog-too_small	14
ureadahead-other.log-too_small	11
procps-static-network-up-too_small	9
cryptdisks-enable.log-too_small	8
lsb-release-too_small	8
network-interface-eth-too_small	8
apport.log-too_small	7

fontconfig-too_small	7
WinEventLog:System:IAS	6
splunkd_access	6
ureadahead-other-too_small	4
kern-too_small	3
kern.log-too_small	3
splunkd_stderr	3
term-4	3
boot-too_small	2
checkfs-too_small	2
checkroot-too_small	2
cryptdisks-enable-too_small	2
media-info-too_small	2
networking-too_small	2
networking.log-too_small	2
dovecot-too_small	1
dovecot.log-too_small	1

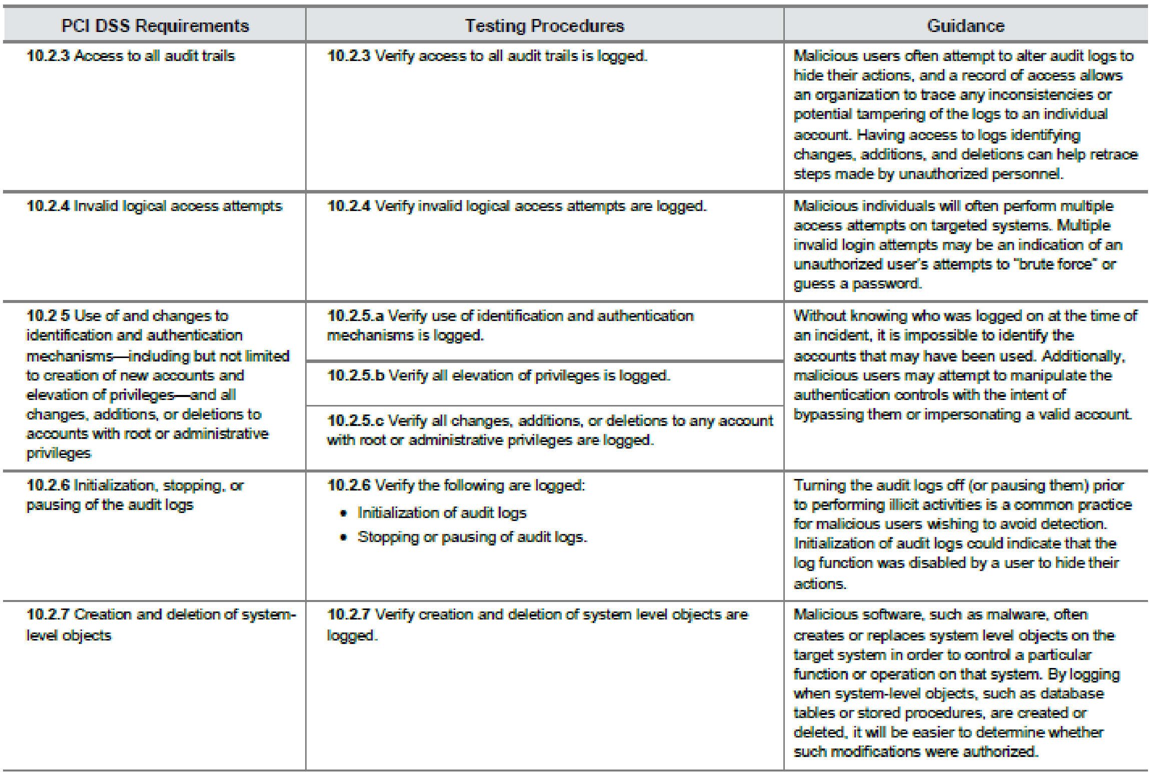
Figure A1.1 – SIEM log source types

Appendix 2: PCI DSS Security Standard



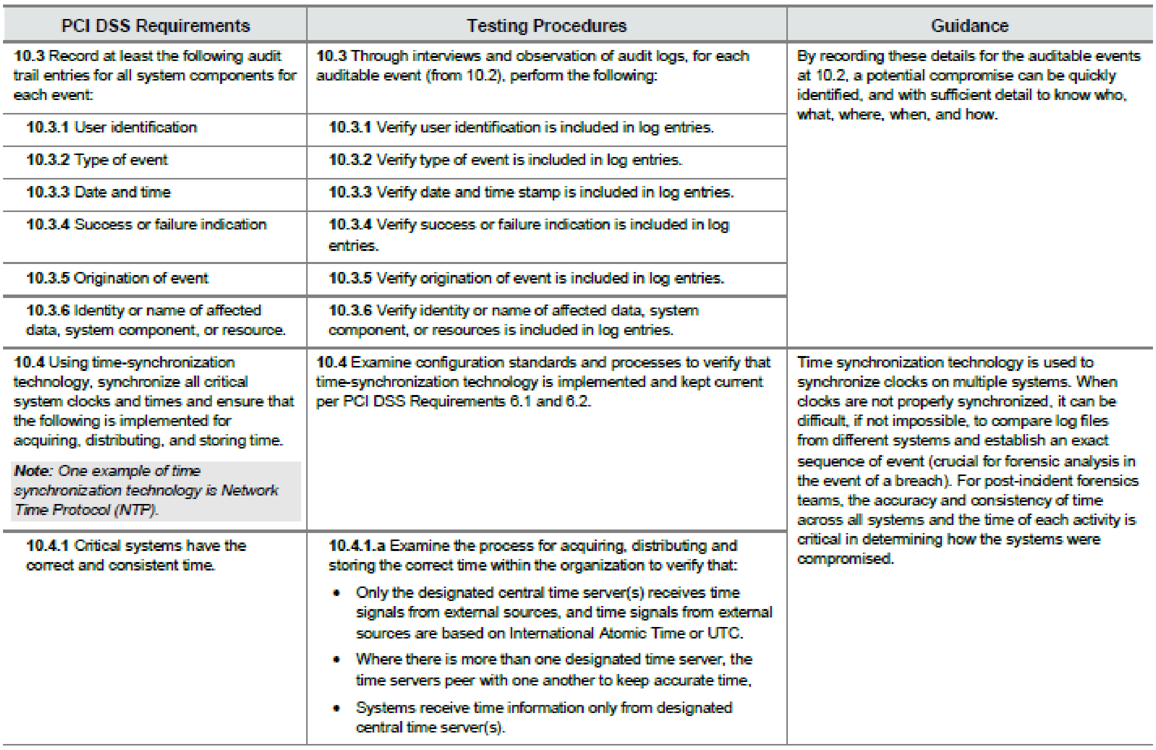
PCI DSS Requirements	Testing Procedures	Guidance
10.1 Implement audit trails to link all access to system components to each individual user.	**10.1** Verify, through observation and interviewing the system administrator, that: • Audit trails are enabled and active for system components. • Access to system components is linked to individual users.	It is critical to have a process or system that links user access to system components accessed. This system generates audit logs and provides the ability to trace back suspicious activity to a specific user.
10.2 Implement automated audit trails for all system components to reconstruct the following events:	**10.2** Through interviews of responsible personnel, observation of audit logs, and examination of audit log settings, perform the following:	Generating audit trails of suspect activities alerts the system administrator, sends data to other monitoring mechanisms (like intrusion detection systems), and provides a history trail for post-incident follow-up. Logging of the following events enables an organization to identify and trace potentially malicious activities
10.2.1 All individual user accesses to cardholder data	**10.2.1** Verify all individual access to cardholder data is logged.	Malicious individuals could obtain knowledge of a user account with access to systems in the CDE, or they could create a new, unauthorized account in order to access cardholder data. A record of all individual accesses to cardholder data can identify which accounts may have been compromised or misused.
10.2.2 All actions taken by any individual with root or administrative privileges	**10.2.2** Verify all actions taken by any individual with root or administrative privileges are logged.	Accounts with increased privileges, such as the "administrator" or "root" account, have the potential to greatly impact the security or operational functionality of a system. Without a log of the activities performed, an organization is unable to trace any issues resulting from an administrative mistake or misuse of privilege back to the specific action and individual.

Figure A2.1 . PCI DSS Security standard mandate 10 (PCI Security Standards Council, 2013)



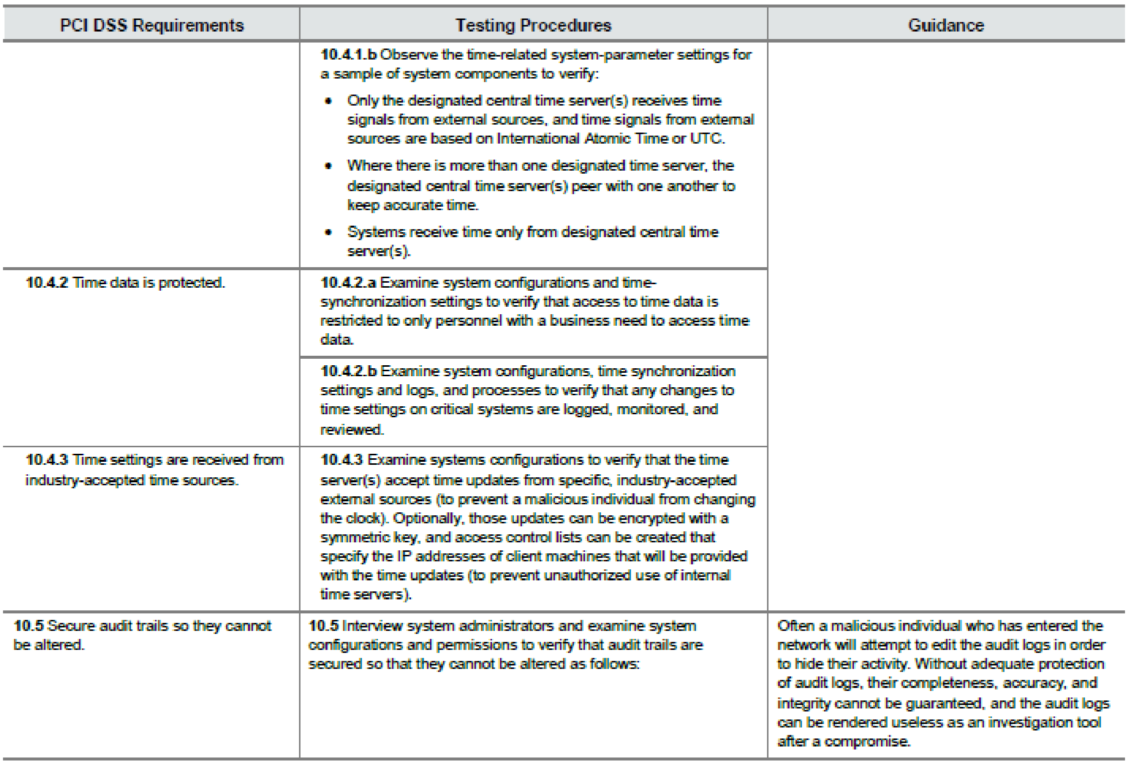
PCI DSS Requirements	Testing Procedures	Guidance
10.2.3 Access to all audit trails	**10.2.3** Verify access to all audit trails is logged.	Malicious users often attempt to alter audit logs to hide their actions, and a record of access allows an organization to trace any inconsistencies or potential tampering of the logs to an individual account. Having access to logs identifying changes, additions, and deletions can help retrace steps made by unauthorized personnel.
10.2.4 Invalid logical access attempts	**10.2.4** Verify invalid logical access attempts are logged.	Malicious individuals will often perform multiple access attempts on targeted systems. Multiple invalid login attempts may be an indication of an unauthorized user's attempts to "brute force" or guess a password.
10.2 5 Use of and changes to identification and authentication mechanisms—including but not limited to creation of new accounts and elevation of privileges—and all changes, additions, or deletions to accounts with root or administrative privileges	**10.2.5.a** Verify use of identification and authentication mechanisms is logged. **10.2.5.b** Verify all elevation of privileges is logged. **10.2.5.c** Verify all changes, additions, or deletions to any account with root or administrative privileges are logged.	Without knowing who was logged on at the time of an incident, it is impossible to identify the accounts that may have been used. Additionally, malicious users may attempt to manipulate the authentication controls with the intent of bypassing them or impersonating a valid account.
10.2.6 Initialization, stopping, or pausing of the audit logs	**10.2.6** Verify the following are logged: • Initialization of audit logs • Stopping or pausing of audit logs.	Turning the audit logs off (or pausing them) prior to performing illicit activities is a common practice for malicious users wishing to avoid detection. Initialization of audit logs could indicate that the log function was disabled by a user to hide their actions.
10.2.7 Creation and deletion of system-level objects	**10.2.7** Verify creation and deletion of system level objects are logged.	Malicious software, such as malware, often creates or replaces system level objects on the target system in order to control a particular function or operation on that system. By logging when system-level objects, such as database tables or stored procedures, are created or deleted, it will be easier to determine whether such modifications were authorized.

Figure A2.2. PCI DSS Security standard mandate 10 (PCI Security Standards Council, 2013)



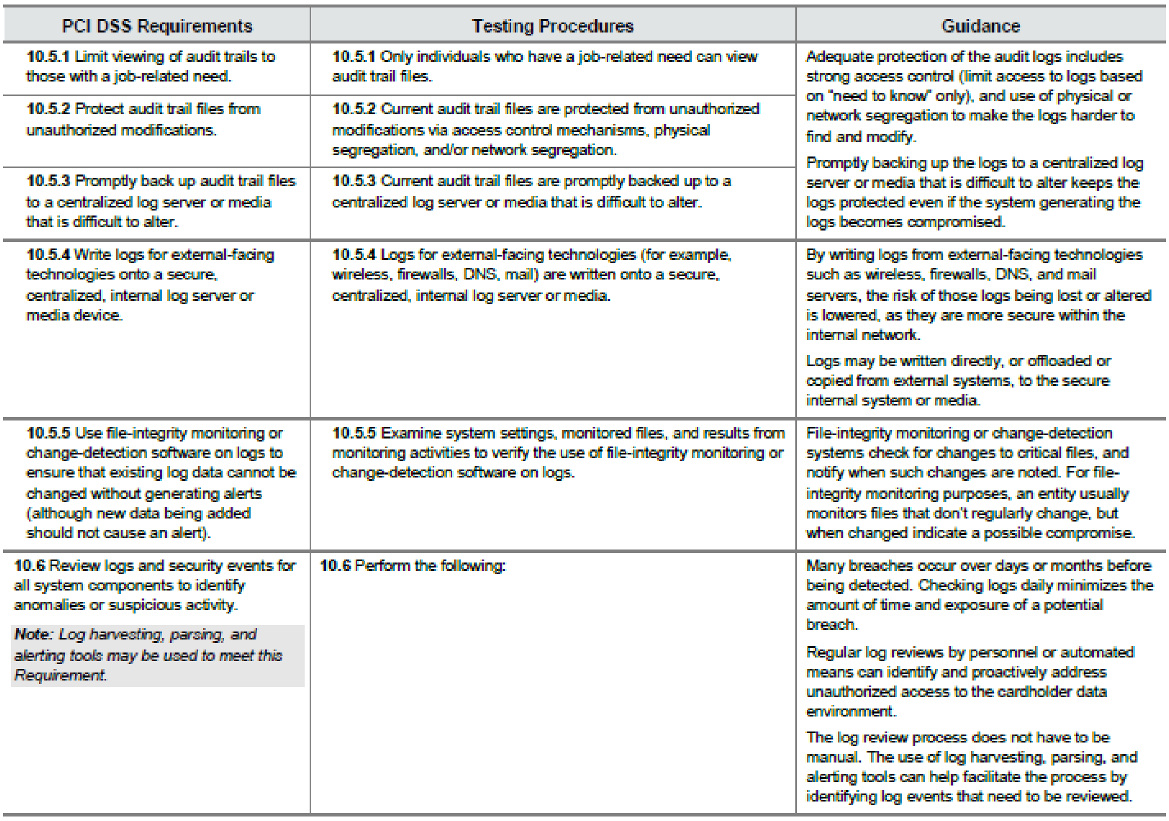
PCI DSS Requirements	Testing Procedures	Guidance
10.3 Record at least the following audit trail entries for all system components for each event:	**10.3** Through interviews and observation of audit logs, for each auditable event (from 10.2), perform the following:	By recording these details for the auditable events at 10.2, a potential compromise can be quickly identified, and with sufficient detail to know who, what, where, when, and how.
10.3.1 User identification	**10.3.1** Verify user identification is included in log entries.	
10.3.2 Type of event	**10.3.2** Verify type of event is included in log entries.	
10.3.3 Date and time	**10.3.3** Verify date and time stamp is included in log entries.	
10.3.4 Success or failure indication	**10.3.4** Verify success or failure indication is included in log entries.	
10.3.5 Origination of event	**10.3.5** Verify origination of event is included in log entries.	
10.3.6 Identity or name of affected data, system component, or resource.	**10.3.6** Verify identity or name of affected data, system component, or resources is included in log entries.	
10.4 Using time-synchronization technology, synchronize all critical system clocks and times and ensure that the following is implemented for acquiring, distributing, and storing time. ***Note:*** *One example of time synchronization technology is Network Time Protocol (NTP).*	**10.4** Examine configuration standards and processes to verify that time-synchronization technology is implemented and kept current per PCI DSS Requirements 6.1 and 6.2.	Time synchronization technology is used to synchronize clocks on multiple systems. When clocks are not properly synchronized, it can be difficult, if not impossible, to compare log files from different systems and establish an exact sequence of event (crucial for forensic analysis in the event of a breach). For post-incident forensics teams, the accuracy and consistency of time across all systems and the time of each activity is critical in determining how the systems were compromised.
10.4.1 Critical systems have the correct and consistent time.	**10.4.1.a** Examine the process for acquiring, distributing and storing the correct time within the organization to verify that: • Only the designated central time server(s) receives time signals from external sources, and time signals from external sources are based on International Atomic Time or UTC. • Where there is more than one designated time server, the time servers peer with one another to keep accurate time, • Systems receive time information only from designated central time server(s).	

Figure A2.3. PCI DSS Security standard mandate 10 (PCI Security Standards Council, 2013)



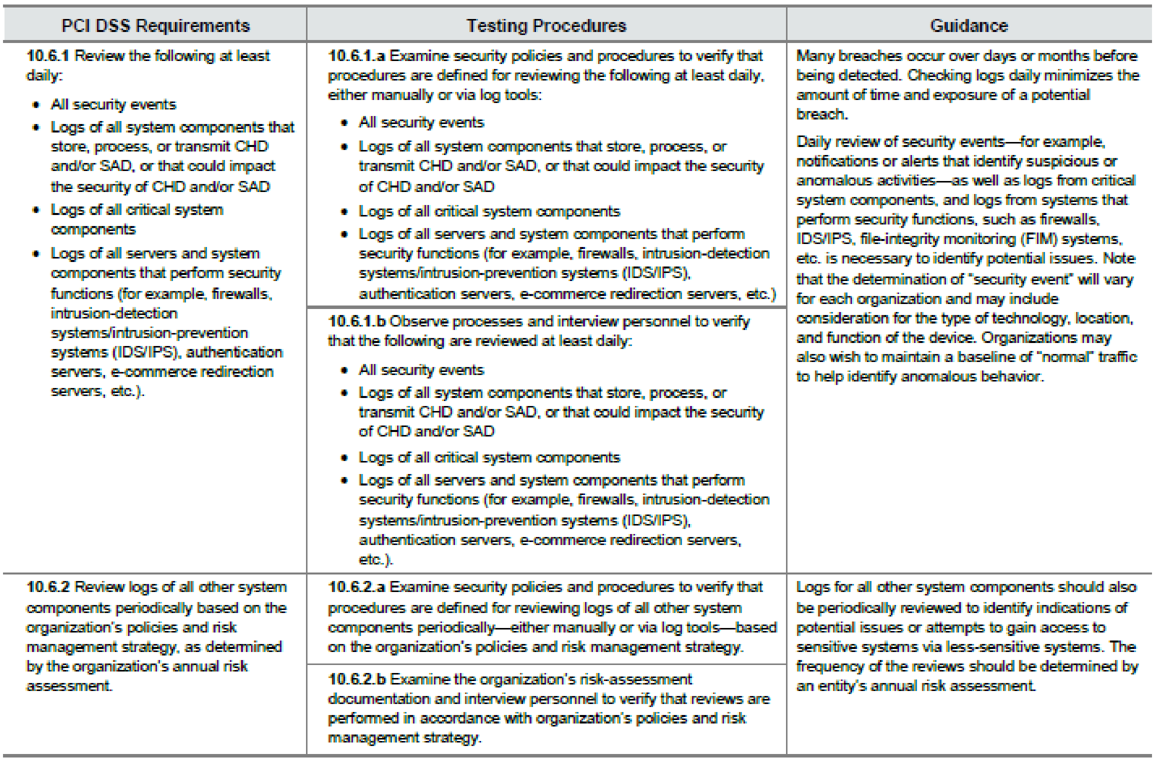
PCI DSS Requirements	Testing Procedures	Guidance
	10.4.1.b Observe the time-related system-parameter settings for a sample of system components to verify: • Only the designated central time server(s) receives time signals from external sources, and time signals from external sources are based on International Atomic Time or UTC. • Where there is more than one designated time server, the designated central time server(s) peer with one another to keep accurate time. • Systems receive time only from designated central time server(s).	
10.4.2 Time data is protected.	**10.4.2.a** Examine system configurations and time-synchronization settings to verify that access to time data is restricted to only personnel with a business need to access time data.	
	10.4.2.b Examine system configurations, time synchronization settings and logs, and processes to verify that any changes to time settings on critical systems are logged, monitored, and reviewed.	
10.4.3 Time settings are received from industry-accepted time sources.	**10.4.3** Examine systems configurations to verify that the time server(s) accept time updates from specific, industry-accepted external sources (to prevent a malicious individual from changing the clock). Optionally, those updates can be encrypted with a symmetric key, and access control lists can be created that specify the IP addresses of client machines that will be provided with the time updates (to prevent unauthorized use of internal time servers).	
10.5 Secure audit trails so they cannot be altered.	**10.5** Interview system administrators and examine system configurations and permissions to verify that audit trails are secured so that they cannot be altered as follows:	Often a malicious individual who has entered the network will attempt to edit the audit logs in order to hide their activity. Without adequate protection of audit logs, their completeness, accuracy, and integrity cannot be guaranteed, and the audit logs can be rendered useless as an investigation tool after a compromise.

Figure A2.4. PCI DSS Security standard mandate 10 (PCI Security Standards Council, 2013)



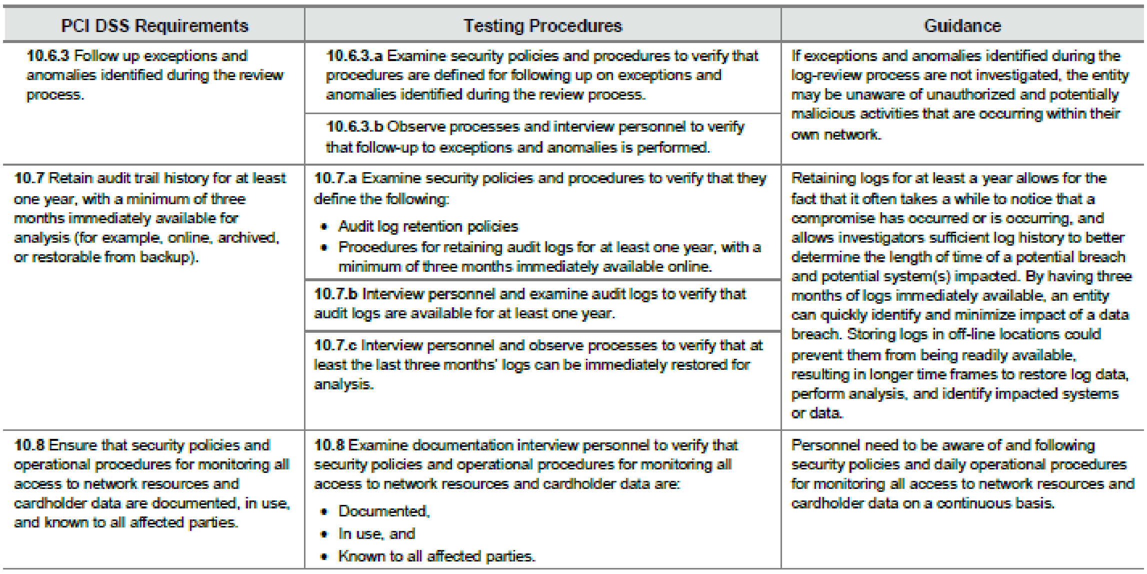
PCI DSS Requirements	Testing Procedures	Guidance
10.5.1 Limit viewing of audit trails to those with a job-related need.	**10.5.1** Only individuals who have a job-related need can view audit trail files.	Adequate protection of the audit logs includes strong access control (limit access to logs based on "need to know" only), and use of physical or network segregation to make the logs harder to find and modify. Promptly backing up the logs to a centralized log server or media that is difficult to alter keeps the logs protected even if the system generating the logs becomes compromised.
10.5.2 Protect audit trail files from unauthorized modifications.	**10.5.2** Current audit trail files are protected from unauthorized modifications via access control mechanisms, physical segregation, and/or network segregation.	
10.5.3 Promptly back up audit trail files to a centralized log server or media that is difficult to alter.	**10.5.3** Current audit trail files are promptly backed up to a centralized log server or media that is difficult to alter.	
10.5.4 Write logs for external-facing technologies onto a secure, centralized, internal log server or media device.	**10.5.4** Logs for external-facing technologies (for example, wireless, firewalls, DNS, mail) are written onto a secure, centralized, internal log server or media.	By writing logs from external-facing technologies such as wireless, firewalls, DNS, and mail servers, the risk of those logs being lost or altered is lowered, as they are more secure within the internal network. Logs may be written directly, or offloaded or copied from external systems, to the secure internal system or media.
10.5.5 Use file-integrity monitoring or change-detection software on logs to ensure that existing log data cannot be changed without generating alerts (although new data being added should not cause an alert).	**10.5.5** Examine system settings, monitored files, and results from monitoring activities to verify the use of file-integrity monitoring or change-detection software on logs.	File-integrity monitoring or change-detection systems check for changes to critical files, and notify when such changes are noted. For file-integrity monitoring purposes, an entity usually monitors files that don't regularly change, but when changed indicate a possible compromise.
10.6 Review logs and security events for all system components to identify anomalies or suspicious activity. ***Note:*** *Log harvesting, parsing, and alerting tools may be used to meet this Requirement.*	**10.6** Perform the following:	Many breaches occur over days or months before being detected. Checking logs daily minimizes the amount of time and exposure of a potential breach. Regular log reviews by personnel or automated means can identify and proactively address unauthorized access to the cardholder data environment. The log review process does not have to be manual. The use of log harvesting, parsing, and alerting tools can help facilitate the process by identifying log events that need to be reviewed.

Figure A2.5. PCI DSS Security standard mandate 10 (PCI Security Standards Council, 2013)



PCI DSS Requirements	Testing Procedures	Guidance
10.6.1 Review the following at least daily: • All security events • Logs of all system components that store, process, or transmit CHD and/or SAD, or that could impact the security of CHD and/or SAD • Logs of all critical system components • Logs of all servers and system components that perform security functions (for example, firewalls, intrusion-detection systems/intrusion-prevention systems (IDS/IPS), authentication servers, e-commerce redirection servers, etc.).	**10.6.1.a** Examine security policies and procedures to verify that procedures are defined for reviewing the following at least daily, either manually or via log tools: • All security events • Logs of all system components that store, process, or transmit CHD and/or SAD, or that could impact the security of CHD and/or SAD • Logs of all critical system components • Logs of all servers and system components that perform security functions (for example, firewalls, intrusion-detection systems/intrusion-prevention systems (IDS/IPS), authentication servers, e-commerce redirection servers, etc.)	Many breaches occur over days or months before being detected. Checking logs daily minimizes the amount of time and exposure of a potential breach. Daily review of security events—for example, notifications or alerts that identify suspicious or anomalous activities—as well as logs from critical system components, and logs from systems that perform security functions, such as firewalls, IDS/IPS, file-integrity monitoring (FIM) systems, etc. is necessary to identify potential issues. Note that the determination of "security event" will vary for each organization and may include consideration for the type of technology, location, and function of the device. Organizations may also wish to maintain a baseline of "normal" traffic to help identify anomalous behavior.
	10.6.1.b Observe processes and interview personnel to verify that the following are reviewed at least daily: • All security events • Logs of all system components that store, process, or transmit CHD and/or SAD, or that could impact the security of CHD and/or SAD • Logs of all critical system components • Logs of all servers and system components that perform security functions (for example, firewalls, intrusion-detection systems/intrusion-prevention systems (IDS/IPS), authentication servers, e-commerce redirection servers, etc.).	
10.6.2 Review logs of all other system components periodically based on the organization's policies and risk management strategy, as determined by the organization's annual risk assessment.	**10.6.2.a** Examine security policies and procedures to verify that procedures are defined for reviewing logs of all other system components periodically—either manually or via log tools—based on the organization's policies and risk management strategy.	Logs for all other system components should also be periodically reviewed to identify indications of potential issues or attempts to gain access to sensitive systems via less-sensitive systems. The frequency of the reviews should be determined by an entity's annual risk assessment.
	10.6.2.b Examine the organization's risk-assessment documentation and interview personnel to verify that reviews are performed in accordance with organization's policies and risk management strategy.	

Figure A2.6. PCI DSS Security standard mandate 10 (PCI Security Standards Council, 2013)



PCI DSS Requirements	Testing Procedures	Guidance
10.6.3 Follow up exceptions and anomalies identified during the review process.	**10.6.3.a** Examine security policies and procedures to verify that procedures are defined for following up on exceptions and anomalies identified during the review process. **10.6.3.b** Observe processes and interview personnel to verify that follow-up to exceptions and anomalies is performed.	If exceptions and anomalies identified during the log-review process are not investigated, the entity may be unaware of unauthorized and potentially malicious activities that are occurring within their own network.
10.7 Retain audit trail history for at least one year, with a minimum of three months immediately available for analysis (for example, online, archived, or restorable from backup).	**10.7.a** Examine security policies and procedures to verify that they define the following: • Audit log retention policies • Procedures for retaining audit logs for at least one year, with a minimum of three months immediately available online. **10.7.b** Interview personnel and examine audit logs to verify that audit logs are available for at least one year. **10.7.c** Interview personnel and observe processes to verify that at least the last three months' logs can be immediately restored for analysis.	Retaining logs for at least a year allows for the fact that it often takes a while to notice that a compromise has occurred or is occurring, and allows investigators sufficient log history to better determine the length of time of a potential breach and potential system(s) impacted. By having three months of logs immediately available, an entity can quickly identify and minimize impact of a data breach. Storing logs in off-line locations could prevent them from being readily available, resulting in longer time frames to restore log data, perform analysis, and identify impacted systems or data.
10.8 Ensure that security policies and operational procedures for monitoring all access to network resources and cardholder data are documented, in use, and known to all affected parties.	**10.8** Examine documentation interview personnel to verify that security policies and operational procedures for monitoring all access to network resources and cardholder data are: • Documented, • In use, and • Known to all affected parties.	Personnel need to be aware of and following security policies and daily operational procedures for monitoring all access to network resources and cardholder data on a continuous basis.

Figure A2.7. PCI DSS Security standard mandate 10 (PCI Security Standards Council, 2013)