International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-4, Issue-6, June 2018
ISSN: 2395-3470
www.ijseas.com

A Proposed Risk Management System Using Event Log Correlation Technique for Host-Based Proactive Security

Hany Abdel-Ghany Gouda[1], Nevine Makram Labib[2]
[1] Senior DBA, Micro, Small and Medium Enterprise Development Agency (MSME) UNDP-Project, Giza, Egypt
[2] Professor of Computer Science, Computer and Information System Department, Sadat Academy for Management Science, Cairo, Egypt

Abstract

Every minute of any Information System's lifetime equals dozens of recorded events; some are normal, and others are abnormal. A huge amount of events exemplifies the main challenge, which is collecting those events from different log files, correlating, and analyzing them on real-time, to gain comprehensive vision and full realization of each event, and take rapid active-response against any risk threatening the functioning of the system.

In such a context, this research has developed a risk management system and applied it inside IT Sector of Micro, Small and Medium Enterprise Development Agency, utilizing a local server for hosting LAMP software bundle, and a total of 700 connected users and 7 system administrators. The event log correlation technique was used to manage the data flow through OSSEC open source tool. All results of access control, operation and activities proved the ability to early threat detection and handling using security measures.

Keywords: *Information System Security, Risk Management, Event Log Management and Correlation, Proactive Security, Host-based IPS, Attacks Pattern.*

1. Introduction

Today's key drivers of information system security are globalization directives and the constant state of rapid changes that induce a dramatic increase in security risks such as new threats and vulnerabilities discovered every day, generating a steeper challenge [1].

The major challenge for organizations is to enhance monitoring and controlling of those risks to meet the security needs, confidentiality, integration, and availability. Accordingly, risk management is one of the most important methods to address security issues, which covers the full spectrum of crisis; before, during and after. Hence, there is a need to develop a system that achieves full perception through collecting and correlating information about events of these crises in order to detect and prevent them [2].

This study primarily focuses on security risk management and related topics, in order to form a comprehensive vision and full understanding of all Host-Based events, and thus improve the efficiency and effectiveness of the monitoring and controlling mechanisms that contribute to mitigating the effects of crises.

2. Information System Security and Risk Management

Realization of the relationship between information system security and risk management must be the approach of the expert in charge, in terms of adopted method to control all types and resources of information systems and protection from unauthorized usage, and understanding the vulnerability, threat, exploit and methodology of mitigation. These components combine to establish initial evaluation and recommended action plan for risk management [3].

2.1 Security Standards

There are many internationally recognized information security standards that are closely related and interdependent. Each one has processes, steps and phases as guidelines of effective security practices, such as ISO/EC 27005 and NIST SP800-30 [4].

2.2 Security Objectives

There should be well-defined needs that commensurate with risks for a specific system and state a clearly achievable objective to protect information and critical elements. Security objectives work toward protecting information from risks that threaten each element within their scope, such as networks, operating systems, Databases, web application, mail, barrier that prevents the assault through the provision of necessary

tools, and means available to protect information from internal or external threats through main objectives [5].

A) **Confidentiality:** authorized restrictions on information access while protecting personal privacy and proprietary information.
B) **Integrity:** guarding information against modification or destruction, while ensuring information non-repudiation and authenticity.
C) **Availability:** Ensuring timely access to and use of information systems.

Attack	Threat to Objective	Attributive
Brute force	Confidentiality	Automated software, it's used a trial-and-error attempt to guess a system's password.
Spoofing/Sniffing		Capture the data packets when transmitting.
Backdoors		Workaround, custom-designed to bypasses normal authentication entry, allow and enabled to altering an existing issue in the system.
Injection	Integrity	Manipulate within back-end repository of application, whether operating system (OS) or database (DB) to get information for OS: read, write, execute or delete for DB: view, insert, modify or delete
Buffer Overflow		Manipulate a server's memory, accessing or overwriting data in any part of memory. causing errors or crashes.
Privilege Escalation		Where a user has account on specific system and trying to elevate his privileges, motivated to gain complete control over a system
Denial of Service	Availability	Launched a huge number of service requests exceed the ability to disrupt handling with it.
Flood		System resource overload, overwhelm the victim server with legitimate requests to replies.

Fig. 2.1 Samples of Attacks Against Security Objectives [6]

2.3 Security Metrics

The major concern in the information security field is what cannot be measured cannot be managed. A framework to help specify goals and objectives should refer to using measurable factors to provide information, fully understand security issues and design an effective solution [7].

SMART metrics, are the measurable security factors as determined by the National Institute of Standards and Technology (NIST), designed to aid system administrator in measuring an enterprise's security posture.

A) **Specific:** Measure a specific variable, which means define security goals that address a particular set of problems and not all possible problems. For example:
   - What is to be secured?
   - What is the objective secured?
   - What risks we try to provide security against?
B) **Measurable:** Must be able to answer how will you know if the goal is accomplished or not? Focusing on determining metrics for a specific topic so that you can measure and track their achievement.
C) **Actionable/Achievable:** Define the scope of coverage, should not measure variables which cannot be acted upon.
D) **Relevant:** Only events from systems in scope need to be collected through filtering and aggregation. Should avoid any information that fails to improve the security.
E) **Timely:** Time frame is always critical for arrival, follow-up, and response per event. It can also make a goal unachievable while delay or ignoring that frame could constitute a threat.

2.4 Security Architecture

It is a unified security model that describes how to apply the security control that addresses the necessities and potential risks involved in a system environment [8].

A) **Negative Security:** A security model, also known as the blacklist, that defines what is disallowed, while implicitly allowing everything else. You know what is bad and you block only the bad stuff.
B) **Positive Security:** A security model known as the whitelist that defines what is allowed and rejects everything else. It denies access to everything, and only allows access to specific authorized resources or functions.

2.5 Security Management Methods

Risk issues are common in any system environment. This may occur due to misconfiguration, default installation, incompatibility between layers in system architecture, and insider/exterior violation. Such methods improve security and mitigate the impact of risk through reactive or proactive approaches [9].

A) **Reactive Approach:** Response to the risks that have already passed through security incidents. Determines and analyzes the causes that have

allowed the incidents to occur for preventing possible repetition.

B) **Proactive Approach:** Characterized by being adaptive, flexible and creatively intellectual. Defines explicit metrics based on closed loop strategy that combines a mixed method of past, present, and future prediction to avoid risks.

2.6 Types of Control

They including controlling according to the time that they act and relative to security events. These techniques use all the monitoring capabilities cited before, during and after incidents to mitigate the risk [3], [10]. And can be classified as:

A) **Detective control:** Designed to detect violations which have already occurred or attempted, or may be on their way, and go through to assure correction instantly.

B) **Preventive control:** Designed to Expect violation attempts and inhibit them.

2.7 Risk Assessment

It is the first phase process of the risk management methodology to determine the Specific adverse event, context, extent and impact of the potential threat that may happen in the future against information system security, reaching result and decision recommendation [3], [11]. The process can be divided into the following steps:

A) **Characterization:** The initial step is to define the scope of the effort. Resource boundaries of the information system security to provide all information about the multiple-layers of system architecture for (Well defined prior to applying any methodology, drawing an overview of the system's processing environment, a preparatory step to security control selection).

B) **Identification:** An iterative process of examining to determine and state what could cause a potential risk, which has a repository of historical data that is updated regularly through life-cycle, identifying new risks. For example, the following must be identified: vulnerabilities, threats, existing planned-security measures and consequences.

C) **Analysis:** The objective is to analyze the controls that have been established then implemented, or are planned for implementation. In an efficient and systematic manner, reflects changes in a system's control environment to reduce the likelihood of a threat's exploiting system vulnerability.

D) **Determination of Likelihood and Impact:** Likelihood "may" occur, that is a potential vulnerability that could be exploited by a specific threat-source that can be described. Impact is the adverse event resulting from successful exploitation of a vulnerability, in terms of breach or degradation of any combination of security goals; Confidentiality, Integrity and Availability. Each likelihood impact is determined and assessed for severity as (Very High, High, Medium, and Low).

2.8 Risk Control

It is the second phase process of the risk management methodology. It involves immediate action to achieve risk-reduction of an acceptable level and implements controls with minimal impact on system capabilities, by recommending output from the first phase of risk assessment process. Controls can be classified according to risk-indicator relative to a security incident and the time that took action [3], [11].
Finally, it is an organized set of principles and rules that drive action in a particular field and can be achieved through the following:

A) **Response:** An automation patch; reactive measures that provide an action plan executed against a risk indicator to avoid the chance that the system architecture could become interrupted, or bring it back up as soon as possible.

B) **Monitoring:** Guarantees the automation of response. It will be implemented as planned, tracking new risks arising during execution that were not previously identified, evaluating their effectiveness as expected or should be developed.

C) **Re-evaluation:** The effectiveness of any countermeasures is evaluated based on the results of monitoring. Actions will be taken to improve, change or keep up with the current

plans. The final step to ensure that the immediate response previously taken removed any risk or further harm to the system and that the problem has not recurred again.

3. Event Log Management and Correlation

The logging mechanism is a text message providing contextual information about the event during operation that explains what is happening. It is generated by multi-layers, e.g. UPS, router, switch, storage unit, operating system, database, middleware and application. These entities across the different layers compose a system architecture.

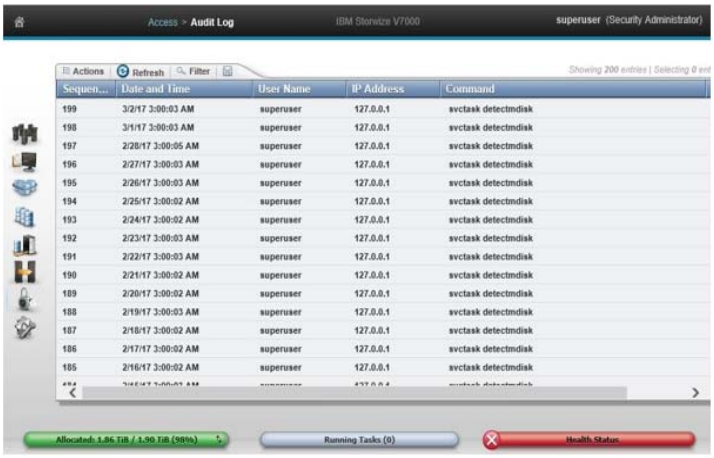

Fig. 3.1 Audit Log for Storage Device

```
Feb 26 11:27:12 sfdapp seahorse-daemon[6420]: init gpgme version 1.1.8
Feb 26 11:27:54 sfdapp gnome-session[6327]: WARNING: Unable to determine session:
Unable to lookup session information for process '6327'
Feb 26 11:27:54 sfdapp gnome-session[6327]: WARNING: Unable to determine session:
Unable to lookup session information for process '6327'
Feb 26 11:27:55 sfdapp gnome-session[6327]: WARNING: Unable to determine session:
Unable to lookup session information for process '6327'
Feb 26 11:48:36 sfdapp gnome-session[30461]: WARNING: GSIdleMonitor: IDLETIME
counter not found
Feb 26 11:48:36 sfdapp gnome-session[30461]: WARNING: Unable to determine session:
Unable to lookup session information for process '30461'
Feb 26 11:48:38 sfdapp seahorse-daemon[30537]: DNS-SD initialization failed: Daemon not
running
Feb 26 11:48:38 sfdapp seahorse-daemon[30537]: init gpgme version 1.1.8
Feb 26 11:48:43 sfdapp pulseaudio[30678]: pid.c: Daemon already running.
```

Fig. 3.2 Event Log for General Message

```
180305 19:40:24 mysqld_safe Starting mysqld daemon with databases from /var/lib/mysql
161030 14:34:06 [Note] /usr/libexec/mysqld: Normal shutdown
161031 10:00:11  InnoDB: Starting shutdown...
161031 10:00:14  InnoDB: Shutdown completed; log sequence number 0 44233
170711 17:13:57  InnoDB: Database was not shut down normally!
161031 10:00:14 [Note] /usr/libexec/mysqld: Shutdown complete
161101 12:57:07 [ERROR] Cannot find or open table score_test/crd_score from the
internal data dictionary of InnoDB though the
161103  9:18:10 [Warning] '--log' is deprecated and will be removed in a future release.
Please use "--general_log'/'--general_log_file" instead.
```

Fig. 3.3 MySQL DB Event Log

```
192.168.34.17 - - [21/Feb/2018:15:08:08 +0200] "POST
/ben/apps/direct_cont/ben_cont_dp.php?PRG_CODE=4&PRJ_CODE=791&GOVT_CODE=02&B
EN_CODE=2467795& "http://192.168.161.85/ben/index.php?dir=direct_cont&file=cont_list"

192.168.34.17 - - [21/Feb/2018:15:08:14 +0200] "POST
/ben/apps/direct_cont/ben_cont_dp.php?PRG_CODE=4&PRJ_CODE=791&GOVT_CODE=02&B
EN_CODE=2467795& "http://192.168.161.85/ben/index.php?dir=direct_cont&file=cont_list"

192.168.34.17 - - [21/Feb/2018:15:08:15 +0200] "POST
/ben/apps/direct_cont/ben_cont_dp.php?PRG_CODE=4&PRJ_CODE=791&GOVT_CODE=02&B
EN_CODE=2467795& "http://192.168.161.85/ben/index.php?dir=direct_cont&file=cont_list"

92.168.34.17 - - [21/Feb/2018:15:08:22 +0200] "POST
/ben/apps/direct_cont/ben_cont_dp.php?PRG_CODE=4&PRJ_CODE=791&GOVT_CODE=02&B
EN_CODE=2467795& "http://192.168.161.85/ben/index.php?dir=direct_cont&file=cont_list"

92.168.34.17 - - [21/Feb/2018:15:08:22 +0200] "POST
/ben/apps/direct_cont/ben_cont_dp.php?PRG_CODE=4&PRJ_CODE=791&GOVT_CODE=02&B
EN_CODE=2467795& "http://192.168.161.85/ben/index.php?dir=direct_cont&file=cont_list"

192.168.34.17 - - [21/Feb/2018:15:08:37 +0200] "POST
/ben/apps/direct_cont/ben_cont_dp.php?PRG_CODE=4&PRJ_CODE=791&GOVT_CODE=02&B
EN_CODE=2467795& "http://192.168.161.85/ben/index.php?dir=direct_cont&file=cont_list"

192.168.34.17 - - [21/Feb/2018:15:08:37 +0200] "POST
/ben/apps/direct_cont/ben_cont_dp.php?PRG_CODE=4&PRJ_CODE=791&GOVT_CODE=02&B
EN_CODE=2467795& "http://192.168.161.85/ben/index.php?dir=direct_cont&file=cont_list"
```

Fig. 3.4 Access Log for HTTP Web-Server

```
[root@sfdapp etc]# tail -f /var/log/cron
Mar  8 10:01:01 sfdapp run-parts(/etc/cron.hourly)[7967]: finished 0anacron
Mar  8 10:01:01 sfdapp run-parts(/etc/cron.hourly)[7958]: starting mcelog.cron
Mar  8 10:01:01 sfdapp run-parts(/etc/cron.hourly)[7982]: finished mcelog.cron
Mar  8 10:05:01 sfdapp CROND[11662]: (root) CMD (LANG=C LC_ALL=C
/usr/bin/mrtg /etc/mrtg/mrtg.cfg --lock-file /var/lock/mrtg/mrtg_1 --confcache-file
/var/lib/mr
(1) Mar  8 10:06:38 sfdapp crontab[13074]: (root) BEGIN EDIT (root)
(2) Mar  8 10:06:43 sfdapp crontab[13074]: (root) REPLACE (root)
Mar  8 10:06:43 sfdapp crontab[13074]: (root) END EDIT (root)
Mar  8 10:07:01 sfdapp crond[2687]: (root) RELOAD (/var/spool/cron/root)
```

Fig. 3.5 Event Log for Cron Job

As such, the event logs are separate and collected from multiple sources of system architecture. They are extremely valuable while correlating with each other, helping to recognize past, present and next of system behavior. Hence, granting the ability to avoid service failures. And they are a primary source of information for identifying the System threats and problems that occur in the System at any point of time [12].

Event log provides a fully Integrated vision, and can be argued as diverse sensors. It provides broader support to monitoring, troubleshooting, predicting risk and mitigating sophisticated attacks.

Gain Security Intelligence of event log management which meets all critical capabilities such as the ability to collect, store, index, search, filter, correlate, analyze and report on real-time to forensics issue, determine the root cause of security incidents and thwart potential threats against of information system.

3.1 Challenges

Some of the challenges include knowing what to look for in a huge amount of events can be the hardest

challenge, additionally gaining useful information and drawing comprehensive vision across domain.
The main challenges in log management are: multiple layers of event log sources, recording similar events differently, inconsistent standard of formats and types, generating a large amount of volumes and types, not all activities and procedures generate logs, constant state of change, and categorization and correlation.

3.2 Correlation Technique

It has become one of the most important security techniques. The objective is to overcome the limitations of templates and traditional security systems that focus on specific problems rather than event correlation for an entire system architecture, which can create customized correlation rules to define actually in precise terms, the relationship between pre-defined events sequence caused by abnormal behavior of procedures and activities, and provide the matching analysis to detect and reduce the potential threats [13].

3.3 Rule-Based Correlation

A pattern recognition that depends on pre-existing knowledge to determines the scenario that an attack must follow. Rule-based correlation approach pre-defines criteria handling occurrence issue, condition, severity, and actions. It will be triggered automatically when a particular event is generated. The trigger is an early warning signal which indicates in advance if a risk is likely to occur.

Classifying attack patterns and comparing them with information sent by notification alert service on the system architecture, attempts to identify behaviors indicative of threats. Almost common forms today are those in which an attacker performs a vulnerability scan of the system to determine attack type to exploit the vulnerability. An effective rule-based would have a comprehensive vision against the vulnerability scan and exploit, especially from insider attackers (perhaps skipping/stopping firewall and traditional security level).

3.4 Event Log Correlation Approach

Correlating events from log files of multiple layers, achieving effective action about detection of threats. It also reduces potential risks due to the knowledge repository from various sources [14]. It can be represented in mathematical expression as below:

$$Threat\ detection = \{event1, event2, event3 \ldots x\}$$

There are two different approaches based on log correlation for attack detection [12], [15].

A) Top-Down Approach: This approach starts from the "Top" of an attack event as a big picture and works "Down" toward the origin status. It helps understand details of content structure. It is a backward-looking method that doesn't differentiate between high and low severity of events. Known attacks are analyzed to determine attack signatures from many event logs.
In simple terms, forensics analyses of the risk by seeking out the source logs and aggregating the impact of operational failures. It measures the variances in variables that are not explained. As such, this approach is simple and relies mainly on historical data.

B) Bottom-Up Approach: This approach starts from the source log "Bottom". It is a forward-looking method and does not rely on historical data. It classifies and differentiates between high and low severity events. Real-time analysis of anomalies from multiple layers of event logs correlated to detect a Potential attack "Up".
Furthermore, Log parser. When log analysis as a subsequence of correlation is performed manually by a scripting language comes to your aid.

4. Host-Based Intrusion Prevention System

4.1 Host-Based Protection Mechanism

While no single mechanism offers adequate protection for the whole system, defense in depth is a commonly used approach to prevention, detection, and response in information system security that utilizes multiple "layers" to address these principles and protect system architecture. In other words, a potential threat, if not

stopped by a first layer, will be stopped by a second layer, the leap from passive detection to active response [16].
A Host-based proactive security is the basis upon which to build a defense strategy. Where should one defensive measure fail, there are other defensive measures in place that continue to provide protection.

4.2 Intrusion Prevention System

Increasing the value and complexities of a system architecture of the servers, always makes them vulnerable to threats. Hence, it is significant to consider a systematic, efficient and automated mechanism. An attempt to detect and prevent intrusion attacks. That monitoring system behavior collects information from a variety of system layers, correlates them with each other and it has indicators/sensors to recognize patterns of abnormal system activities in the past, present and expectation of the future.

Both techniques have functionality of defense tools; an intrusion detection system for monitoring and intrusion prevention system for controlling [17].

This is a proactive approach; threat prevention capabilities that examine anomaly-based submitted from intrusion detection technique and configure immediate active response should be used for each type of alert. It is an active system that averts vulnerability exploits and reduces threats, while being a prevention method, a sample of active response and preventive countermeasure that can be used on host-based to reduce threats. And that can be categorized according to "system architecture profile" which was modified.

5. Development of A Proposed System for Host-Based Proactive Security

Proactive security is at the forefront of information system security. It predicts the risk before occurring and even if it happens, the response is timely to recover. The following steps have been used to build risk management system that depends on intrusion prevention capabilities on single host (where it will be using the Integration of log-based correlation and anomaly-based approach at the phase of intrusion detection technique, followed by writing new own rules and shell scripts as a rapidly active response at the phase of intrusion prevention technique) and able to efficiently detect known and unknown attacks. An applied study on Information Technology Sector, Enterprise Development Agency (MSME).

5.1 System Implementation Schema

Collecting information about events and correlating them to identify and distinguish activities/violations of three types of attacks: command execution, path traversal, and privilege escalation, then reporting. The following subsections will demonstrate the implementation steps. Improving the quality of Host-based Intrusion Prevention System (HIPS) is effective when increasing the sensitivity of true positive alarms and reducing false positive alarms. Below, you can see the log-based correlation vs. Proactive Method

5.2 Physical Deployment Model

Host-Based Intrusion Detection System (HIDS), considered as initialization part of a proactive security architecture. Security tools that gather information about details of the activity occurring on Host, and detect vulnerabilities. The term "Host" refers to server platform, which means there is a system architecture including several layers: the kernel of the operating system at the bottom base, file systems structure, packages, demons, ports, security setting, configuration files, and applications in the top layer.

Fig. 5.1 Deployment Based

5.3 Installation Design

Single Host “Local”, implementing structure to collect, process, and monitor data of an event of suspicious activities occurring through the entire server, rather than passing the output to another host.

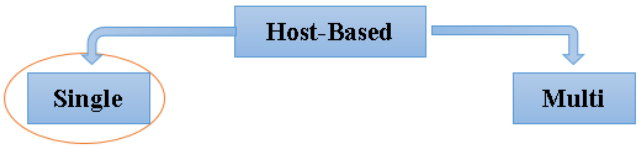

Fig. 5.2 Design, Structure Based

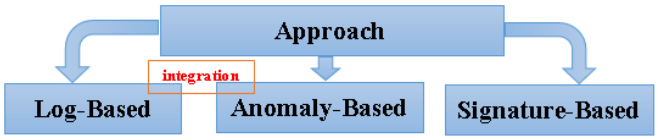

Fig. 5.3 Detection Approach

5.4 Platform Specifications

A) IBM server System X3650 M4, 16 Core, 16GB of Memory.
B) LAMP Software Bundle: Linux 6.6_64bit, Apache HTTP Web-Server, MySQL DB, and PHP Web-Development.

5.5 Data Source

Information about system architecture layers is gathered from:

- System boot log (/var/log/boot.log).
- PAM Modules (/var/log/secure).
- Apache web server (access_log and error_log).
- General messages (/var/log/messages).
- Jobs are scheduled (/var/log/cron).
- Audit system (/var/log/audit/audit.log).
- MySQL DB (/var/log/mysqld.log).
- Linux Partition scheme (/etc/fstab).
- Linux Kernel setting (/etc/sysctl.conf).
- Run levels.
- Users and groups (passwd and group).
- Package Management (/var/ossec/sysarch).
- Daemons (/var/ossec/sysarch/)
- Ports (/var/ossec/sysarch/)
- Server utilization (/var/ossec/sysarch/)
- World-writable files (/var/ossec/sysarch/).

5.6 Detection-Based Approach

Integration feeding of log-based and anomaly-based approach will bring the advantages of both. Rules are predefined for normal and abnormal behaviors which are created by event log correlation technique to detect an intruder act of known attacks. Intrusion detection will alert directly in real-time to the occurrence of events of a critical issue, then log-analysis tool evaluates the relationships among those events and any others that may be occurring in the system architecture, and it has the ability to detect unknown attacks.

5.7 Per-Host Detection Method

The following scenario describes common security issues regarding a combination of attacks in one attack "relation one to many", while attempting to exploit the system architecture, there are always steps as part of an attack used by an intruder with the intent to escalate and maintain persistent access. This includes but is not limited to command execution, path traversal, and privilege escalation "related to each other's".
A path traversal attack gains access to application source code, configuration files and directories that are stored outside the document root path"/var/www/html" by manipulating variables that reference files with "../" to display the file "/etc/passwd" which lists the users, then execute commands trying to crack password of lower user account and modify that file to get rights of root as privilege escalation attack. At this point, the intruder can control the whole system. Therefore, the detection method to reduce an impact of attacks depends on continuous per-host vulnerability scanning to recognize the suspicious activities during pre-processing stages and prevent threats before exploits.

Attacks	Solutions to reduce impact (composite rule based on event log correlation)
Command Execution **Path Traversal** **Privilege Escalation**	vulnerability scanning to identify: - Config system structure, partition scheme, kernel, run levels, and packages. - Run time enforcement, used in a whitelist box for specific (superuser, on time, source IP) to prevent use of any non-sanctioned commands. - No Non-Root Accounts Have UID, no accounts have empty passwords, and files not owned by any user or group. - Find (recently modified files, backdoor PHP shell script on a host, and world-writable file). - Check Daemons/ports which are running/stopped opened/closed. - Config structures of web server, and PHP. - Config events add/delete/modify for users/groups. - Login failure, sshd communication accepted/bad/fatal, and su good/bad. - All application processes run with the minimal privileges required. - Crontab jobs

Fig. 5.4 Per-Host Detection Method

As such, correlating information from multiple event logs helps in achieving effective response when detecting the attacks.

Finally, does this solve all security issues? Detection method does not cover all security issues, but plays an important role in the phase of system architecture security.

5.8 OSSEC Tool

The Complementary security layer approach focuses on maintaining appropriate operations within security measures. Implementing security at each level is very important and selecting a specific tool from most popular open source security tools vary based on requirements. The selection is based on feature set, simplicity and activeness among the community development for enhancing the tools better and better to match requirements.

Open Source HIDS Security has extensive configuration options, which can modify the source code to add new capabilities. It allows customizing alert rules and writing script code that takes actions in response to security alerts.

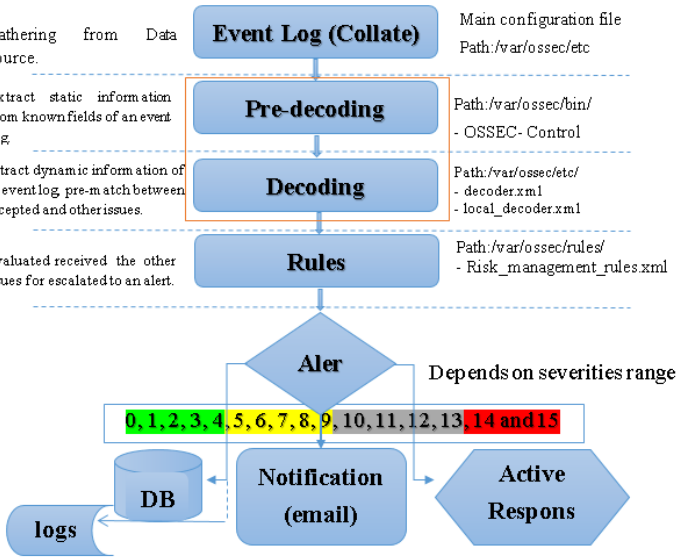

Fig. 5.5 Analysis Process of OSSEC (Event Flow Diagram)

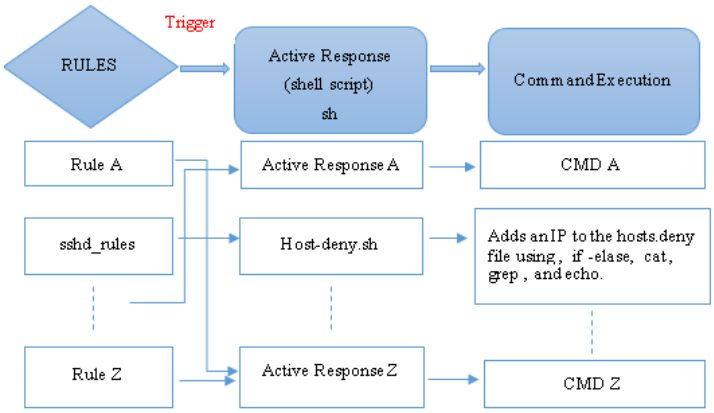

Fig. 5.6 Active-Response Diagram

5.9 Set Correlation Rules

The event log correlation technique through rule-based as XML objects, which are supposed to match the received pattern. Here is a complete example of syntax, where the rules have been divided into sub-routines:

```
<group name="$RMS1,">
  <rule id="100100" level="0" noalert="1">
    <decoded_as>pam</decoded_as>
    <description>RMS100_100. PAM_UNIX_RULES
     </description>
  </rule>
<rule id="100103" level="5">
    <if_sid>100100</if_sid>
    <match>authentication failure; logname=</match>
    <description>RMS100_100_SUB
     User_login_failed.</description>
    <group>authentication_failed,</group>
  </rule>
<rule id="100151" level="10" frequency="5"
timeframe="120">
    <if_matched_sid>100103</if_matched_sid>
    <same_source_ip />
    <description>RMS100_100_SUB 5/times failed
    logins during 2 minutes from same_srcip.</description>
    <group>authentication_failures,</group>
  </rule>
</group> <!-- "RMS1" Done... -->
```

- **rule_id 100100:** Configured for PAM_MODULES group, with ignore alert.
- **rule_id 100103:** Configured to alert depends on rule_id “100100”, and when matching pattern.
- **rule_id 100151:** Configured to alert depends on rule_id “100103” within two factors, first one is

the frequency to specify how many times of pattern must occur before the rule trigger an alert, and the second is the timeframe to recurrence and look for the pattern.

Following is a list of all allowed options used to correlate and reduce false positive alarms:

A) If_matched_sid: Specify which rule wants the composite rule to look and match.
B) If_matched_group: Specify which group type wants the composite rule to look and match.
C) Same_id: Decode ID be the same.
D) Same_source_ip: Decode source IP must be the same.
E) Same_source_port: Decode Source port must be the same.
F) Same_dst_port: Decode Destination port must be the same.
G) Same_location: Specify the hostname or agent must be the same.
H) Same_user: Decode user_name must be the same.

6. Risk Management System Test Boundaries

System architecture generates event log entries to record as much information about everything that is happening on the Host as possible.

The following list summarizes some of the information that provides a way to track risk-relevant:

Sensitivity files.
Date and time, and type of an event.
Association of user ID with relevant-event.
Password (accepted and failed).
Authentication mechanisms, e.g. SSH, SU, and SUDO.
Unauthorized usage.
Successful and Failure evidence such as login, conversation, and execute.
Session (open and closed).
Source IP, program name, and destination path.
Risky words (attack, bad, Corrupted, denied, error, fail, fatal, illegal).

6.1 Events Volume Results Analysis

The analytics provide comprehensive visualizations and reporting of data-collected, data-sources and severities levels. Graphics A, B, and C display the event volume and the event totals for the selected timeframe.

A) Data collected "Host-Based" in the last 24 hours. (Sun, Feb 04, 2018 8:00am – Mon, Feb 05, 2018 8:00am); Events: 120309; Types: All

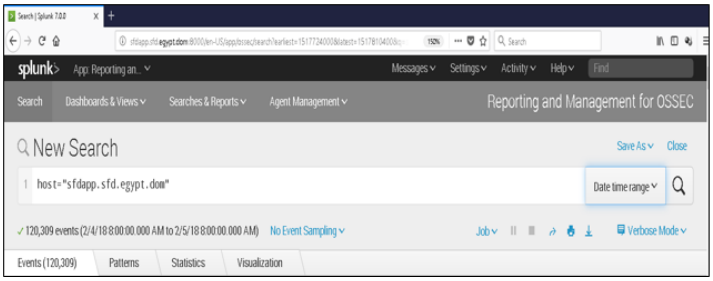

Fig. 6.1 Large Number of Events /day

Part of the above events, out of official working hours (Sun, Feb 04, 2018 4:00pm till Mon, Feb 05, 2018 8:00am); Events: 3797; Types: All

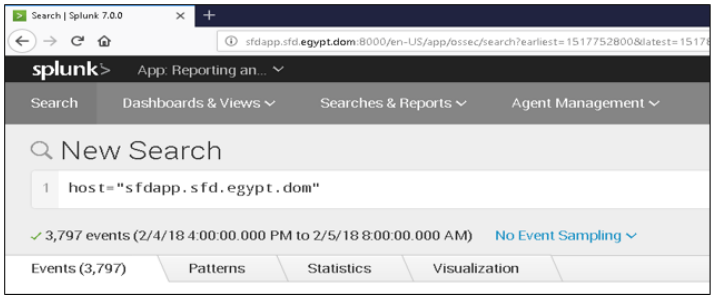

Fig. 6.2 Part of the day

B) Data-Source reporting, top 4 values of event log = 119410 events from 120309 "approximately 99.25%"

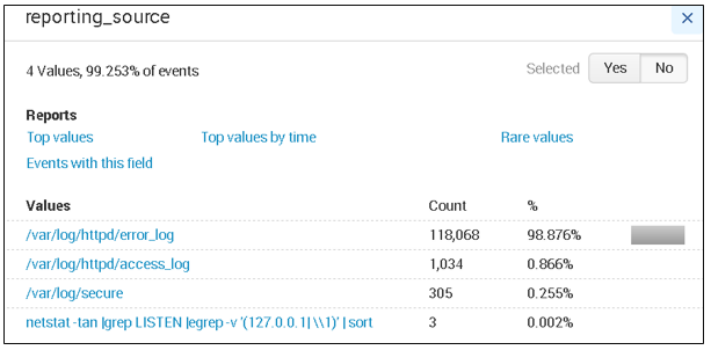

Fig. 6.3 Source of Top Event / day

C) Severity, top 7 values of severity = 120208 "approximately 99.20%"

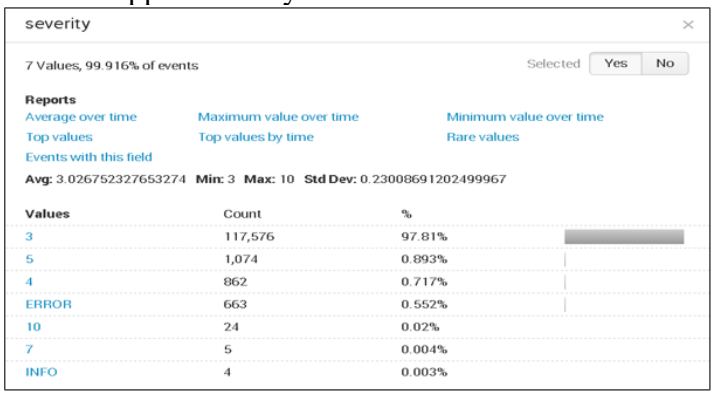

Fig. 6.4 Source of Top Event / day

6.2 Results Analysis of Access Control Monitoring

6.2.1 Global System Activities: Customized rule_id ‘101002’ for Risky Words: Found matching risky words as defined before like “failed and failure”, alert by mail triggered when the event is detected.

6.2.2 PAM_UNIX Modules:

A) Customized rule_id ‘100303’ for sudo: Found severity level ‘4’ correlated with rule_id ’100300’ and matching for the PAM module used, to run splunk program with root privilege, then alert by mail triggered and send a notification when the event is detected.
B) Customized rule_id ‘106715 and 100101’ for sshd: About rule_id ‘106715’, severity level ‘3’ correlated with rule_id ’106700’ and matching word “Accepted” for the PAM module used, and ignore the notification. And rule_id ’100101’ severity ‘3’ correlated with rul_id ‘100100’ and matching ‘session opened for user’, and no action.
C) Customized rule_id ‘100101’: Found session opened issue while logging as hany and switch to superuser root by using ‘su’ command as PAM module.
D) Customized rule_id ‘100103’: Found Authentication failure issue, user hany trying to execute command through sudo for updating the system and entering wrong password
E) Customized rule_id ‘100305’: Found unauthorized issue, while matching the word ‘user NOT in sudoers’ user hany trying to add a new user through sudo command. The alert mail triggered and send a notification when the event is detected.
F) Customized rules for multiple authentication failure: While a particular user is using Remote Desktop Connection, trying to log and entering wrong password multiple time.
The results of this situation were:
- A number of ‘9’ events were recorded, each one of them related to specific issue. Constructing correlation role based on sequence of events and their association order by rule_id ‘100100, 100103, 106700, 106716, 100501 and 100502’, and matching many words of access control messages where it defined before as 'multiple authentication failures' with a high level of severity’10’
- The active response, alert mail triggered to send a notification when the event is detected, and blocked that IP Address.

6.2.3 HTTP Event Logs: Detected a user login to the beneficiary’s system and upload of a high amount of data ‘POST request’ through Excel file. The construction of the correlation role based on sequence of events and their association order by rule_id ‘105000, 105130 and 105133’, if matched calling the next where it defined before as 'high amount of POST requests in a small period of time'.

In this situation, the action taken was notifying by mail only, because it is a legal process in our Agency where data files are uploaded. Secondly; this case explains the pattern of the threat.

6.3 Results Analysis of Operations and Activities Monitoring

Efficiently provide a level to detect malicious and other abuses of legitimate access, that occurs via start/stop daemons, opened/closed port, and manipulation in cron jobs.

The result of the Risk Management System (RMS) has become a regular and comprehensive enhancement of Host-based proactive security to be under control and can be managed. Scenarios have been implemented during a combined legal use, anomaly and misuse to evaluate "RMS". And demonstrate the ability of the system to detect and prevent all situations in real-time.

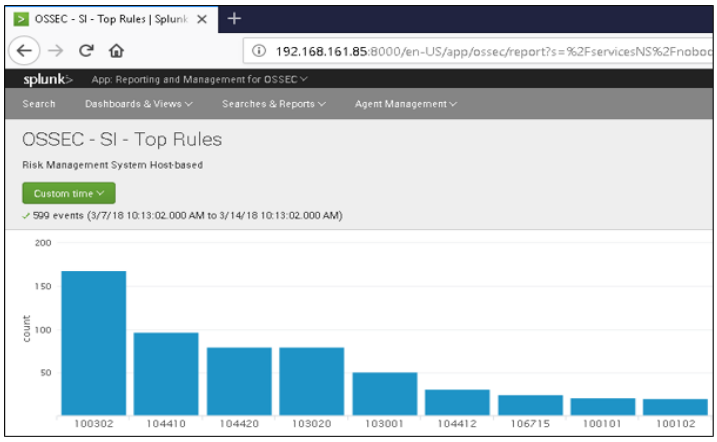

Fig. 6.5 Top 9 Rules During One Week

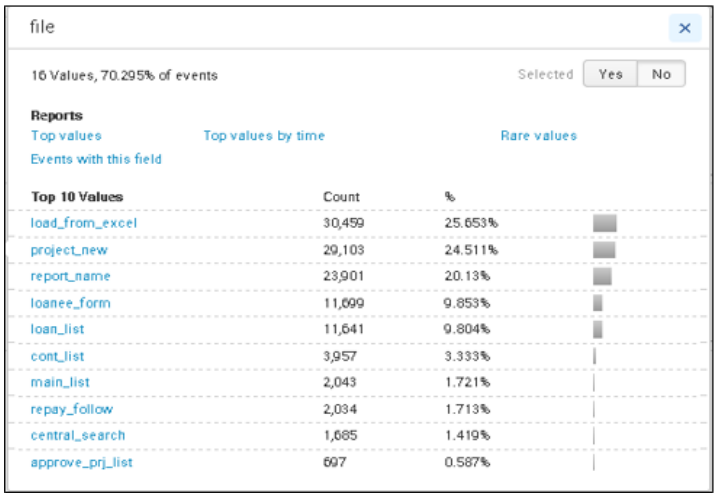

Fig. 6.6 File Opened by Web Application System During One Day

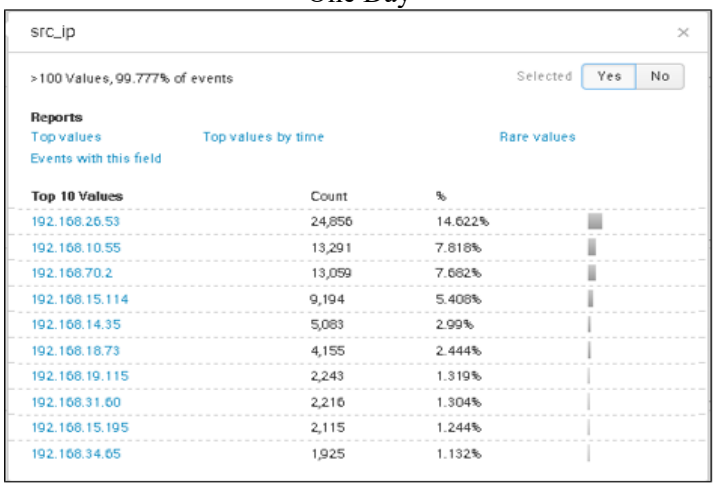

Fig. 6.7 Source IP Address from Regional Offices During One Day

7 Conclusion and Future Work

7.1 Conclusion:

With incidents and potential threats exploding inside the information system sectors, whether intentionally or accidentally, to violate security policies or attempt attacks, it is difficult to detect risks without a comprehensive and constant vision, and one cannot be sure that the system has not been violated before. So, the need for a mechanism is very important to control and manage the overflow of security data available in your hands, which guarantees rapid active response to them, depending on their characteristics.

There are no templates or packages for a 100% complete protecting solution. The IT security specialists and researchers need to be proactive and take a lot of procedures regarding managing risks, in order to address related threats and vulnerabilities before any significant damage can occur. Build own protecting solution is important to verify and ensure that the multiple levels of security are adequate and secure. Therefore, that taking into consideration the proposed risk management system, especially after achieving desired results, would be sufficient for Host to be highly secure.

7.2 Future Work

As recommendations, determine strengths, limits and, in some cases, risks, determine data source which collecting information about events, build your intrusion detection and prevention capabilities, event correlation technique helps produce a manageable level of alerts, posting Implementation schema and drawing the attacks pattern, and suitable tool from open source, the selection based on feature set, simplicity and active among the community development for enhancing the tools better and to match requirements.
As future work, integrate the control procedures, generate custom logs, correlation rules and shell scripts as rapid active-response for each expected event and may be affecting the profile of the system architecture.

References

[1] A. Johnson, K. Dempsey, R. Ross, S. Gupta, and D. Bailey, "Guide for Security-Focused Configuration Management of Information Systems", Computer Security Division, Information Technology Laboratory (National Institute of Standards and Technology), NIST Special Publication 800-128, August 2011.

[2] G. Locke, and D. Patrick, "Guide for Applying Framework to Federal Information Systems, A Security Life Cycle Approach", Computer Security Division, Information Technology Laboratory (National Institute of Standards and Technology), NIST Special Publication 800-37 Revision 1, February 2010.

[3] G. Stonebumer, A. Goguen, and A. Feringa, "Risk Management Guide for Information Technology Systems", Computer Security Division, Information Technology Laboratory (National Institute of Standards and Technology), NIST Special Publication 800-30, July 2002.

[4] C. Raspotnig, "Requirements for Safe and Secure Information Systems", Dissertation for the Degree of Philosophiae Doctor, Department of Information Science and Media Studies, University of Bergen, Norway, 2014.

[5] M. Nieles, K. Dempsey, and V. Pillitteri, "An Introduction to Information Security", Computer Security Division, Information Technology Laboratory (National Institute of Standards and Technology), NIST Special Publication 800-12 Revision 1, June 2017.

[6] Common Attack Pattern Enumeration and Classification, A community Resource for Identifying and Understanding Attacks [Online]. Available: https://capec.mitre.org/index.html *[Accessed: Apr. 28, 2018].*

[7] E. Chew, M. Swanson, K. Stine, N. Bartol, A. Brown, and W. Robinson, "Performance Measurement Guide for Information Security", Computer Security Division, Information Technology Laboratory (National Institute of Standards and Technology), NIST Special Publication 800-55 Revision 1, July 2008.

[8] M. Swpana, and G. Kumar, "Contending Malware Threat Using Hybrid Security Model", International Research Journal of Engineering and Technology (IRJET), Volume 04, Issue 12, e-ISSN: 2395-0056, p-ISSN: 2395-0072, 2017.

[9] E. STROIE, and A. RUSU, "Security Risk Management – Approach and Methodology", Internatica Economica, Volume 15, no 1/2011, 2011.

[10] D. Richards, A. Oliphant, and C. Grand, "Global Technology Control Guide – Information Technology Controls", The Institute of Internal Auditors, USA, March 2005.

[11] IBM Corporation, Software Group, Thought Leadership White Paper, "Managing Security Risks and Vulnerabilities", USA, January 2014.

[12] C. Abad, J. Taylor, C. Sengul, , and W. Yurick, "Log Correlation for Intrusion Detection: A Proof of Concept", Department of Computer Science, University of Illinois at Urbana-Champaign, National Center for Supercomputing Applications (NCSA), Science Applications International Corporation (SAIC).

[13] A. Brathen, "Correlating IDS Alerts with System Logs by Means of a Network-Centric SIEM Solution", Master's Thesis, Master of Science in Information Technology, 30 ECTS, Department of Computer Science and Media Technology, Gjovik University College, Norway, 2011.

[14] NETFORENSICS White Paper, "Event Correlation Matters: Practical, Automated Solutions for Protecting Critical Data", (2009). www.netforensics.com

[15] B. Deokar, "Intrusion Detection System using Log Files and Reinforcement Learning", International Journal of Computer Applications, Vol 45 – No.19, May 2012, India, Mumbai, 2012.

[16] N. Krawetz, Introduction to Network Security, Boston, Massachusetts: 1st edition, 2007.

[17] J. Beale, A. Baker, and J. Esler, Snort IDS and IPS Toolkit, Syngress Publishing, Inc, 2007.