RESEARCH ARTICLE OPEN ACCESS

Big Data Security Analytic Solution using Splunk

P.Charishma, K.Venkatesh
M.Tech, Information Security and Cyber Forensics, SRM University, Chennai
M.Tech , Assistant Professor (Sr.G), Department of Information Technology, SRM University, Chennai

Abstract—
Over the past decade, usage of online applications is experiencing remarkable growth. One of the main reasons for the success of web application is its "Ease of Access" and availability on internet. The simplicity of the HTTP protocol makes it easy to steal and spoof identity. The business liability associated with protecting online information has increased significantly and this is an issue that must be addressed. According to SANSTop20, 2013 list the number one targeted server side vulnerability are Web Applications. So, this has made detecting and preventing attacks on web applications a top priority for IT companies. In this paper, a rational solution is brought to detect events on web application and provides Security intelligence, log management and extensible reporting by analyzing web server logs.

Keywords—*Web application, Web-server logs, splunk, SIEM, Security Intelligence, Log Management, Splunk Forwarder*

I. INTRODUCTION

Advances in web technologies coupled with a changing business environment, mean that web applications are becoming more prevalent in corporate, public and Government services today. Although web applications can provide convenience and efficiency, there are also a number of new security threats, which could potentially pose significant risks to an organization's information technology infrastructure if not handled properly. The simplicity of HTTP Protocol makes it easy for the attackers to spoof the identity and hack into the web application. There are many methods to hack into the web application like Cross-site scripting, Bruteforce, Bufferoveflow, SQL injection etc. There are Different types of Event detection mechanisms like SIEM (Security Information and Event Management), Analyzing the Web Traffic, Analyzing Infrastructure Logs etc. In this paper, a solution has been brought to provide a solution to detect and prevent attacks on web applications by analyzing the web server logs.

Why should you bother analyzing log files instead of using a network intrusion detection system? There might be several reasons for this:

1. The HTTP traffic may be SSL encrypted (HTTPS);
2. There may be no NIDS (hard to deploy; another zone of attack)
3. High traffic load makes it difficult to analyze network traffic (in real time).
4. NIDS are designed to work on the TCP/IP level, and thus they may not be as effective on the HTTP layer;
5. IDS evasion techniques (HTTP, encoding, fragmenting, etc). In this we are analyzing the web server logs using a Analytics tool called splunk. It takes the logs from the Web server and analyzes them according to the patterns and provides Security intelligence and log management to the organizations.
6.

II. RELATED WORK:

Though there are many mechanisms SIEM (Security information and Event Management) has been evolved as a Powerful mechanism for incident response and security intelligence. SIEM technology provides real-time analysis of security alerts generated by network hardware and applications. SIEM is sold as software, appliances or managed services, and are also used to log security data and generate reports for compliance purposes.

It describes the product capabilities of gathering, analyzing and presenting information from network and security devices applications; vulnerability management and policy compliance tools; operating system, database and application logs; and external threat data. A key focus is to monitor and help manage user and service privileges, directory services and other system configuration changes; as well as providing log auditing and review and incident response.

III. SYSTEM ARCHITECTURE

The system architecture will be the overall proposed work of the System with its modules as presented in the fig 1.The following are the tasks to be done to provide Security intelligence to the Organization.

Module 1: In this module, the logs of the web server will be collected and will be transferred to Splunk using splunk forwarders.

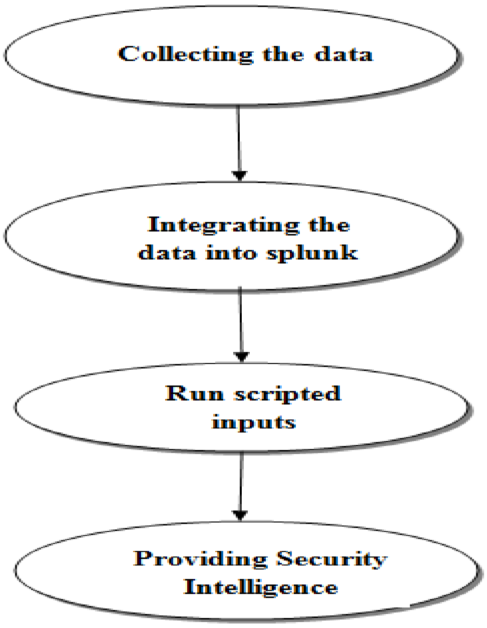
Collecting the data
Integrating the data into splunk
Run scripted inputs
Providing Security Intelligence

Fig1:Modules of Proposed System

Module 2: In this module, The data will be mounted to splunk and creating jobs,Triggers on splunk.
Module 3: In this module, on the data mounted on to the splunk run the preconfigured scripts and categorize the events as Bruteforce, Bufferoveflow, Cross-site scripting etc.
Module 4: In thi module,the alerts and reports will Be generated and provides the security intelligence.

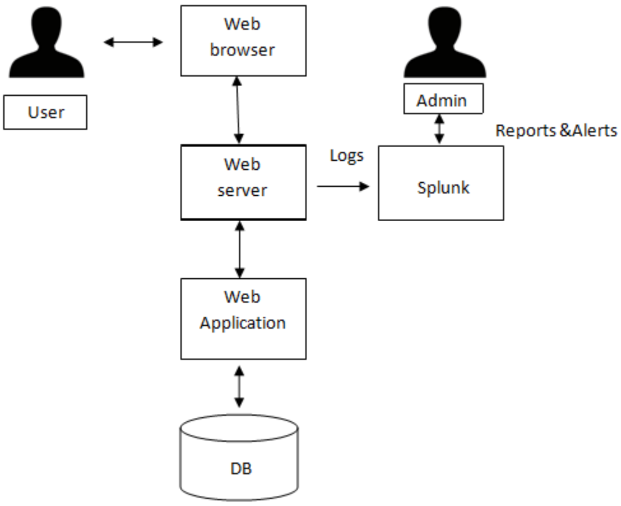

Fig2: System Architecture

The system architecture consists of user, web browser, Web server, Web application , splunk and admin. The user makes a request through the web browser to the web server and the web server will serve request and gives access to the web application .The logs of the web server will be given to Data analytics tool called splunk by using the Forwarder. As soon as the logs received by splunk it triggers the job and runs some script over splunk and it analyze the logs and categorize the events according to the patterns and if any event occurred it will generate an alert or report. The alerts can be directly sent to user's mail id using splunk.

Event Detection Mechanism:
As the logs are received by Splunk, it will detect the events by analyzing the logs. The CLF log file contains a separate line for each HTTP request. A line is composed of several tokens separated by spaces: Attacks on Web Applications can be detected by host, ident, authuser date request status bytes values. If a token does not have a value, then it is represented by a hyphen ().Tokens has these meanings:

1. **Host:** The fully qualified domain name of the client, or its IP address.
2. **Ident:** If the Identity Check directive is enabled and the client machine runs ident, then this is the identity information reported by the client.
3. **Authuser:** If the requested URL required a successful Basic HTTP authentication, then the user name is the value of this token.
4. **Date:** The date and time of the request.
5. **Request:** The request line from the client, enclosed in double quotes (").
6. **Status:** The three digit HTTP status code returned to the client.
7. **Bytes:** The number of bytes in the object returned to the client, excluding all HTTP headers.

A request may contain additional data like the referrer and the user agent string. Let us consider an example of log entry (in the Combined Log Format [Apache Combined Log Format, 2007].

```
127.0.0.1    frank
[10/Oct/2007:13:55:36 0700]
"GET /index.html HTTP/1.0" 200 2326
"http://www.example.com/links.html"
"Mozilla/4.0 (compatible; MSIE
7.0; Windows NT 5.1; .NET CLR
1.1.4322)"
```

From the above server log, if two or more logs have the same session id and different Agent and different timestamp means that will be categorized as Cookie hijacking.
A request may contain additional data like the referrer and the user agent string. Let us consider an example

of log entry (in the Combined Log Format [Apache Combined Log Format, 2007].

127.0.0.1 frank [10/Oct/2007:13:55:36 0700] "GET /index.html HTTP/1.0" 200 2326 "http://www.example.com/links.html " Centos"

If the requests are flooding from the same ip to the web server then it is categorized as Buffer overflow. If the logs contain login failed error for many times then it is categorized as Bruteforce attack.

Providing Security Intelligence:
Splunk Provides Security intelligence, so that the Organization's administrator can come to know the Vulnerabilities of their own web application and also can be aware of the events happened from time to time.

The newest security strategy against advanced attackers divides an attack into stages called the "kill chain" Defenders can use the kill chain as a way to break down and analyze events, with the goal of stopping attackers as early as possible.

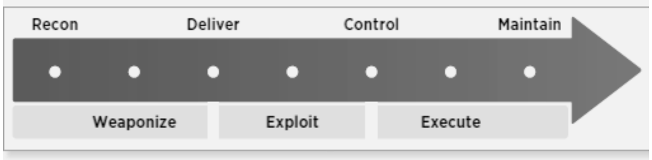

Fig: The stages of kill chain

Splunk reads data from a *source*, such as a file or port, on a *host* (e.g. "my machine"), classifies that *source* into a *Sourcetype* (e.g., "syslog", "access combined", "apache error", ...), then extracts timestamps, breaks up the source into individual events (*e.g., log events, alerts, …*), which can be a single-line or multiple lines, and writes each event into an *index* on disk, for later retrieval with a *search*. *Search-time Processing:* When a *search* starts, matching indexed *events* are retrieved from disk, *fields* (e.g., code=404, user=David...) are extracted from the *event*'s text, and the *event* is classified by matching against *event type* definitions (e.g., 'error', 'login' ...). The *events* returned from a search can then be powerfully transformed using Splunk's *search language* to generate *reports* that live on *dashboards*

Events:
An *event* is a single entry of data. In the context of log file, this is an event in a Web activity log:

173.26.34.223 - - [01/Jul/2009:12:05:27 -0700] "GET /trade/app?action=logout HTTP/1.1" 200 2953

More specifically, an event is a set of values associated with a timestamp. While many events are short and only take up a line or two, others can be long, such as a whole text document, a config file, or whole java stack trace. Splunk uses line breaking rules to determine how it breaks these events up for display in the search results.

Indexes:
When you add data to Splunk, Splunk processes it, breaking the data into individual events, timestamps them, and then stores them in an *index*, so that it can be later searched and analyzed. By default, data you feed to Splunk is stored in the "main" *index*, but you can create and specify other *indexes* for Splunk to use for different data inputs.

Event Types:
Eventtypes are cross-referenced searches that categorize *events* at search time. For example, if you have defined an *eventtype* called "problem" that has a search definition of "error OR warn OR fatal OR fail", any time you do a search where a result contains error, warn, fatal, or fail, the event will have an eventtype field/value with eventtype=problem. So, for example, if you were searching for "login", the logins that had problems would get annotated with Event type=problem.
Eventtypes are essentially dynamic tags that get attached to an *event* if it matches the search definition of the *eventtype*.

Reports:
Search results with formatting information (e.g., as a table or chart) are informally referred to as *reports*, and multiple *reports* can be placed on a common page, called a *dashboard*.

Permissions/Users/Roles:
Saved Splunk objects, such as *saved searches*, *eventtypes*, *reports*, and *tags*, enrich your data, making it easier to search and understand. These objects have *permissions* and can be kept private or shared with other users, via roles (e.g., "admin", "power", "user"). A *role* is a set of capabilities that you can define, like whether or not someone is allowed to add data or edit a report. Splunk with a Free License does not support user authentication.

Transactions:
A *transaction* is a set of events grouped into one event for easier analysis. For example, given that a customer shopping at an online store would generate web access events with each click that each share a SessionID, it could be convenient to group all of his events together into one *transaction*. Grouped into one *transaction event*, it's easier to generate statistics like how long shoppers shopped, how many items

they bought, which shoppers bought items and then returned them, etc.

Fields:

Fields are searchable name/value pairings in *event* data. As Splunk processes *events* at index time and search time, it automatically extracts fields. At index time, Splunk extracts a small set of default *fields* for each *event*, including *host*, *source*, and

***Sourcetype*:** At search time, Splunk extracts what can be a wide range of fields from the event data, including user- defined patterns as well as obvious field name/value pairs such as user_id=jdoe.

Tags:

Tags are aliases to *field* values. For example, if there are two host names that refer to the same computer, you could give both of those host values the same tag (e.g.,"hall9000"), and then if you search for that tag (e.g., "hal9000"), Splunk will return *events* involving both host name values.

Forwarder:

A *forwarder* is a version of Splunk that allows you to send data to a central Splunk indexer or group of *indexers*. An *indexer* provides indexing capability for local and remote data

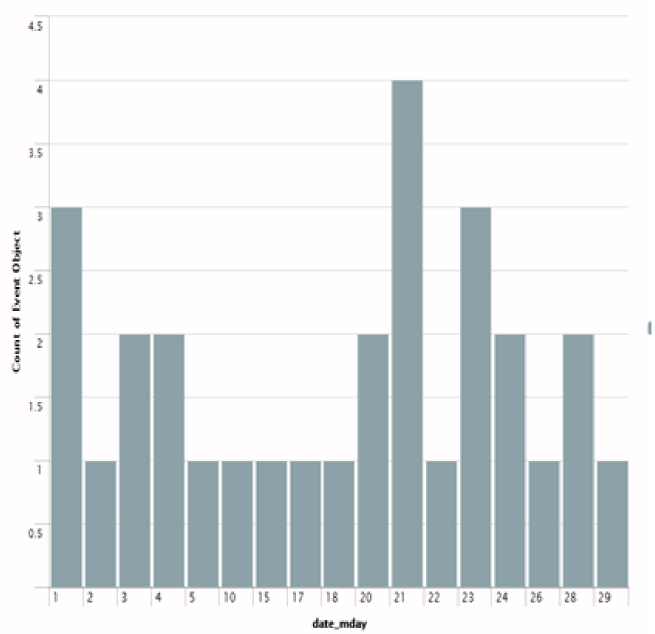

Fig 3: Graphical Representation of Event Occurrence in splunk

IV. CONCLUSION

As Internet usage and online applications are experiencing spectacular growth. Worldwide, there are over a billion Internet users at present. The simplicity of the HTTP protocol makes it easy to steal and spoof identity and to attack web applications become an easy target. This paper gives a scenario to detect the attacks on web applications through logs and provides security compliance to the user.

V. REFERENCES

[1.] Baker, Wade, Alex Hutton, C. David Hylender, Christopher Novak, Christopher Porter, Bryan Sartin, Peter Tippett, and J. Andrew Valentine. Data Breach Investigations Report. Technical report, Verizon Business RISK Team, 2009.

[2.] Hasan, Masum, Binay Sugla, and Ramesh Viswanathan. "*A Conceptual Framework for Network Management Event Correlation and Filtering Systems*". Integrated Network Management, 233-246, 1999.

[3.] Myers, J., Grimaila, M.R., and Mills, R.F., "*Towards Insider Threat Detection using Web Server Logs*," Proceedings of the Cyber Security and Information Intelligence Research Workshop (CSIIRW 2009), Oak Ridge National Laboratory, Oak Ridge, TN, April 13-15, 2009.

[4.] https://www.splunk.com/en_us/solutions/solution-areas/big-data.html.

Author's Profile:

P.charishma received B.Tech Degree from Audisankara Institute of Technology, Gudur, A.P. Pursuing M.Tech in Information Security and cyber Forensics in SRM University, Chennai, India. Areas of Interest: Threat Intelligence, Information Security

Mr.K.VENKATESH, Assistant Professor (Sr.G), Department of Information Technology, SRM University. Pursued M.Tech in Computer Science from SRM University in 2010. Areas of Interest : Data mining