IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 14-19

Web Server Log Analysis

Sangeeta Vhatkar, Harsh Rawal, Shubham More, Vishal Patel
Department of Information Technology, Thakur College of Engineering and Technology, Mumbai, India

Abstract: *Web server log analysis is a novel and peculiar field constantly formed and changed by the convergence of various Web technologies. Due to its interdisciplinary character, the diversity of issues it addresses, and the variety and number of Web applications, it is the subject of many distinctive and diverse research methodologies. [1] About Log Files - Current software application often produce (or can be configured to produce) some auxiliary text files known as log files. The main problem with log files being the vast amount of unpreprocessed data which needs to be processed for analysis. Such files are used during various stages of software development, mainly for debugging and profiling purposes Use of log files helps testing by making debugging easier. It allows to follow the logic of the program, at high level, without having to run it in debug mode. Nowadays, log files are commonly used also at customer's installations for the purpose of permanent software monitoring and/or fine-tuning. Log files became a standard part of large application and are essential in operating systems, computer networks and distributed systems.*

Keywords: *web, log file, server.*

I. Introduction

Web-based applications have become prevalent in the ubiquitously-connected world. The recent increase in demand for huge data storage and high processing speed has led to the era of cloud computing, yet, most cloud-based systems and applications are accessed through the Internet web. Web security therefore has remained as top priority for Internet and Cloud-Services Providers (ISP/CSP). In order to improve ISP/CSP user confidence and to protect web-based systems and applications, strong, effective security mechanisms must be deployed. Many architecture designs have been proposed and many cryptographic algorithms implemented; yet intruders continue to gain access to web based application, to steal confidential information, and to make unwanted modifications including recent intrusions made to Target, the Home Depot, BlueCross Insurance, and many hospital and government information systems. Two major techniques have been used in log analysis: pattern matching and machine learning. While the pattern matching method may work dynamically, only known patterns can be recognized, yet new types of injections may be created when only small changes are made to existing patterns Machine learning also has its limitation, since classification in machine learning algorithms works with probabilities, it may not be able to correctly classify SQL injections that combine groups of words each was classified with high probability as non-SQL injection. Most existing log analysis methods for SQL injection detection are based on either pattern matching or machine learning. Log files containing approximately 20000 Log entries or more are taken into account as the example dataset in our project. Entries such as the IP addresses, category of Operating system, Origin of IP address and the threat concerned with the related IP are processed. Log files are often the only way how to identify and locate an error in software, because log file analysis is not affected by any time-based issues known as probe effect. This is the opposite to an analysis of a running program, when the analytical process can interfere with time– critical or resource–critical conditions within the analyzed program. Log files are often very large and can have complex structure. Although the process of generating log files is quite elementary and unexacting, log file analysis could be a tremendous task that requires enormous computational resources, long time and enlightened procedures. This often leads to a common situation, when log files are continuously generated and occupy valuable space on storage devices, but nobody uses them and utilizes enclosed information. The proposed work till date has substantial outcomes such as the log file provided as the input is mined for the appropriate data and kept ready for further use. Therefore, with the growing speed of data in the web, a framework is needed to process and analyze the data for vulnerabilities. Here we elucidate the analysis of log files using pattern matching framework which incorporates the major preprocessing task and session identification algorithm to handle vast amount of log data. From the results it is concluded that processing a huge file in distributed fashion reduces the time and data transfer cost, without moving the data. The scope of this research is Proactive monitoring - Move from reactive to proactive real-time log monitoring and view app performance, system behavior, and unusual activity across the stack. [2] Monitor key resources and metrics, and eliminate small issues before they turn into big problems. Troubleshooting - Trace issues down to their root cause by analyzing them in the context of the entire stack. To observe how components interact, identify correlations, and share findings with experts across team boundaries to resolve problems fast.

Data analysis and optimization - Analyze and visualize your data to answer key questions, track SLA compliance, and identify anomalies. Our project automatically recognizes common log formats and gives you a structured summary of all your parsed logs. Team collaboration and integration - Building and running complex systems requires tight coordination between development, operations, and product.[3]

II. Methodology

Web Server Log analysis system is a kind of application which will help to detect and avoid the attacks which might happen on the server by using the log files of the server itself, this log files can be used to accurately give us an idea so as to where and how the attack is been initiated and is taking place.

A. PHASE 1: (Planning, Analysis, Design, coding)

- Planning: This involves how to analyze the log files from the obtained data. To use the available database to detect what kind of attack takes place on the server at the present moment.
- Analysis: Critical study, analysis and review of feasibility for proposed solution. If a genuine user is an attacker (unauthorized person) a warning will be issued to the server manager (authorized person) then the preferred action can be taken by him.
- Coding: Open source algorithms can be implemented using various coding techniques including java, PHP etc. Design: The application will get an access to the log files of the server which will then indicate what kind of attack is been executed and what action can be taken.

B. PHASE 2: (Integration, Testing, Deployment)

- Integration – Integrating of various proposed modules such as Log detection module whether it detects attacks from various destinations.
- Testing –Exhaustive testing using test cases to check the integration and fixing bugs for proposed solution. Perform alpha testing after completion of prototype.
- Deployment – Give the completed prototype to available server for security purpose in the active servers.

We present the application of the proposed methodology for analysing of the web log files. In this Paper, we have developed an expert system to assist the web designer and web administrator to improve their website by determining occurred link connections in the website. Firstly, we have obtained access log files which are recorded in web server of the First University. The obtained log files were analysed by proposed web usage mining methodology in SAS software. We present an overview of the tasks for each step and discuss the challenges involved. The architecture consist of three main tasks for performing web usage mining: pre-processing, pattern discovery and pattern analysis. An important task in web usage mining application is the creation of a suitable pre-processed usage data set. This process is usually complex and critical to the successful extraction of useful from the log files in web usage mining. Purpose of the pre-processing is to offer a structural, reliable and integrated data source for pattern Data cleaning is the first step performed in the preprocessing of web usage mining. In the raw logs, not all the log entries are valid for pattern analysis. We only want to keep the entries that carry relevant information. Therefore, the data cleaning step is used to eliminate the irrelevant entries from the access log files. Transaction identification is to create meaningful clusters of references for each user. Each user session is considered either as a single transaction consisting of many page references or a set of many single-page reference transaction. A session can be described as the group of activities performed by a user from the moment he entered the website to the moment he left it. Therefore, session identification is the process of segmenting the access log of each user into individual access session .

III. Flow Diagram

A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, and plan, improve and communicate often complex processes in clear, easy-to-understand diagrams. Figure 1 shows the flow chart for Web server log analysis using pattern

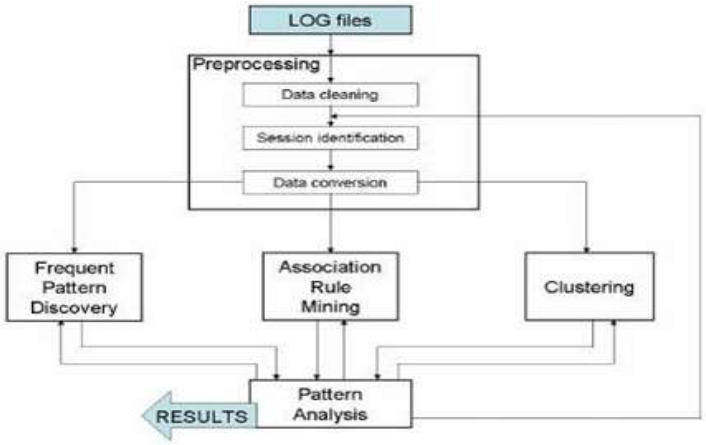

Fig. 1: Web server log analysis using pattern

In the preprocessing phase the data have to be collected from the different places it is stored (client side, server side, proxy servers). After identifying the users, the click-streams of each user has to be split into sessions. In general, the timeout for determining a session is set to 30 minute. The pattern discovery phase means applying data mining techniques on the preprocessed log data. [3][4] It can be frequent pattern mining, association rule mining or clustering. In this paper we are dealing only with the task of clustering web usage log. In web usage mining there are two types of clusters to be discovered: usage clusters and page clusters. The aim of clustering users is to establish groups of users having similar browsing behavior. The users can be clustered based on several information. In the one hand, the user can be requested filling out a form regarding their interests, for example when registering on the web portal. The clustering of the users can be accomplished based on the forms. On the other hand, the clustering can be made based on the information gained from the log data collected during the user was navigating through the portal. Different types of user data can be collected using these methods, for example (I) characteristics of the user (age, gender, etc.), (ii) preferences and interests of the user, (iii) user's behavior pattern. The aim of clustering web pages is to have groups of pages that have similar content. This information can be useful for s arch engines or for applications that create dynamic index pages. The last step of the whole web usage mining process is to analyze the patterns found during the pattern discovery step. The irrelevant patterns have to be filtered out, and the resulted patterns or clusters have to be validated. Some visualization techniques can help this process for the user.[5][6]
Pre-processing, Pattern discovery and Pattern Analysis is shown in Figure 2. Block diagram of Interesting Rules, Pattern's and statistics

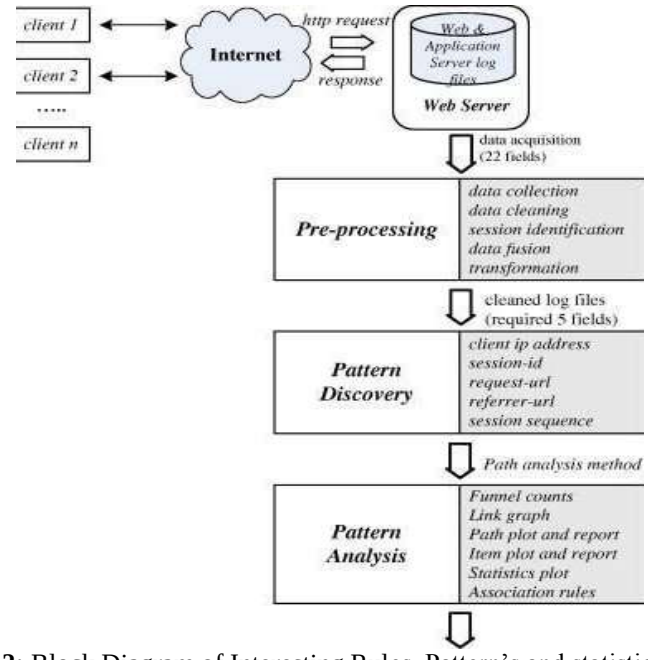

Fig.2: Block Diagram of Interesting Rules, Pattern's and statistics [7]

Data collection and pre-processing:

An important task in web usage mining application is the creation of a suitable pre-processed usage data set. This process is usually complex and critical to the successful extraction of useful from the log files in web usage mining. Purpose of the pre-processing is to offer a structural, reliable and integrated data source for pattern.

Data cleaning:

Data cleaning is the first step performed in the preprocessing of web usage mining. In the raw logs, not all the log entries are valid for pattern analysis. We only want to keep the entries that carry relevant information. Therefore, the data cleaning step is used to eliminate the irrelevant entries from the access log files.

Transaction identification:

The aim of transaction identification is to create meaningful clusters of references for each user. Cooley et al. (1999a) propose a general model for transaction identification. In their model, each user session is considered either as a single transaction consisting of many page references or a set of many single-page reference transaction.

Session identification:

A session can be described as the group of activities performed by a user from the moment he entered the website to the moment he left it. Therefore, session identification is the process of segmenting the access log of each user into

A session can be described as the group of activities performed by a user from the moment he entered the website to the moment he left it. Therefore, session identification is the process of segmenting the access log of each user into individual access sessions.

Pattern discovery:

In order to extract patterns of usage from web log files are used data mining techniques for web usage mining.[4][6] Pattern discovery is the key process of the web mining, which includes the algorithms and techniques from several research areas, such as data mining, machine learning, statistics and pattern recognition. The techniques such as statistical analysis, association rules, clustering, classification, sequential pattern and dependency modelling are used to discover rules and patterns.

IV. Result and discussion

Fig.3: Attack patterns

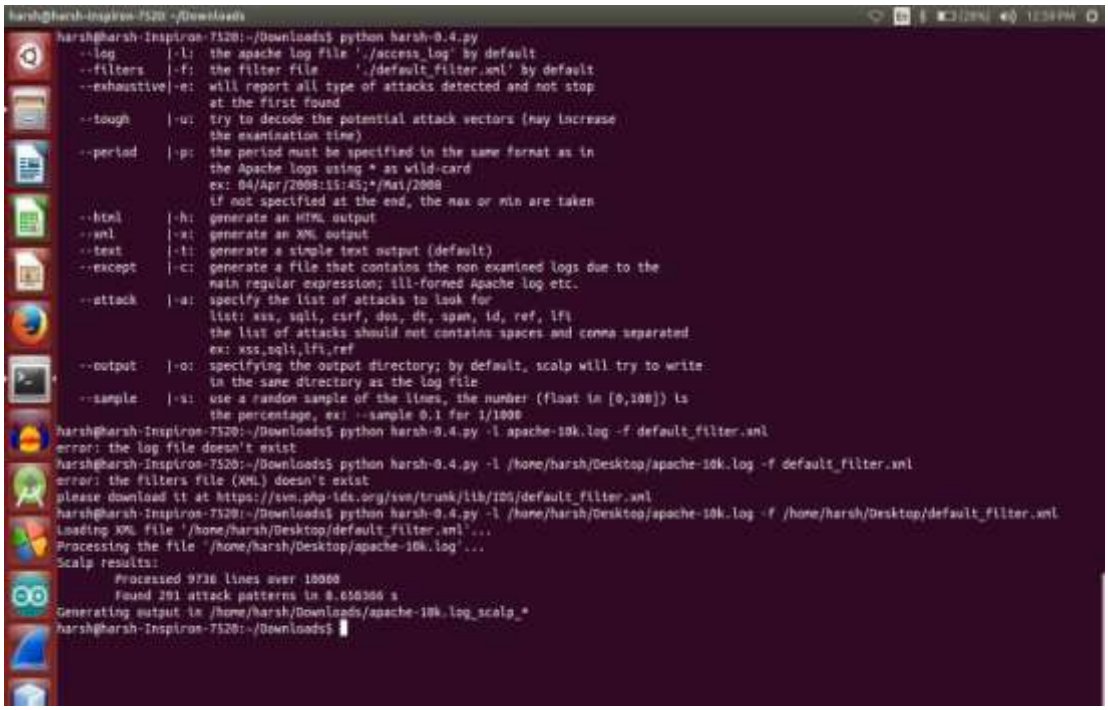

Fig.4: Attack Patterns

This application will provide the security to any server using the log files associated with the server. This can accurately detect the attack that takes place also the details related to the attacker such as the ip of attacker, type of attack and other attacker attributes. We have to plan that stored data i.e. attack log type in database (Trained database) needs to match with the database which also contains the same log data as trained database i.e., test database. If the trained database matched with the test database following action will be performed first, if user is authorized person access will be granted and secondly if user is an unauthorized person then threat will be detected. Design by using Log Analysis Algorithm all the connection followed by using Database to detect what kind of threat or attack can be executed, and Warning is associated to Dashboard output and it is used for indication purpose and after all the interconnection is done the server is used for keeping it protected from threats. Various modules for coding will be created like:

1. Application code having log analysis algorithm developed over proper code.
2. Dashboard output for warning the server manager.
3. Possible action that can be taken by the server manager.

For integration, constructing the various modules of the proposed solution to integrate them into a prototype. The integrated prototype will be tested exhaustively within the test cases to validate and verify the prototype's functioning (unit /performance testing) and perform integration testing, system testing and stress testing. The benefits of Analysis on after deploying the proposed prototype include:

1. Proper and efficient way for securing server.
2. This system can be used in various multiple servers.
3. The cost is efficient.
4. The owner has full control on his/her server.
5. Attacker can be identified easily.

V. Conclusion

The normal process of log information retrieval includes store process and then visualize, however, the proposed platform does preprocess-store and directly visualize, thereby decreasing overall latency. Making use of the proposed platform, real time data analytics can be made possible on large data sets, thus facilitating prompt insights into the data. The platform can take in any type of log files, which gives it a generic capability to analyze different logs simultaneously. The platform can also be extended as a base module to serve other applications for decision making on the basis of real-time analytics. The clusters, which are created on the fly depending on the patterns identified, help shortening the search time. Thus the searching is done in a single cluster instead of the whole data store. This tremendously reduces the lag between request and response. As an added functionality the platform can be extended to serve any type of data other than logs, if the data matches

the regular expression in system. Thus, the proposed solution will be implemented.

Instead of tracking the behavior of overall users (interested or not interested) in order to redesign the web site to support users. The data mining techniques like Association, Clustering, and Classification can be applied only on to the group of interested regular users to find frequently accessed patterns which results in less time consumption and less memory utilization with high accuracy and performance.

References

[1]. R. Kosala, H. Blockeel, Web mining research: a survey, SIGKDD: SIGKDD Explorations: newsletter of the special interest group (SIG) on Knowledge discovery & data mining, ACM 2 (1), 1–15, 2000

[2]. R. Cooley, B. Mobasher, and J. Srivastava, "Data preparation for mining World Wide Web Browsing patterns," Knowledge and Information Systems, Vol. 1, No. 1, 1999, pp. 5-32

[3]. Navin Kumar Tyagi, A.K. Solanki and Sanjay Tyagi: "An Algorithmic Approach to Data Preprocessing in Web Usage Mining". International Journal of Information Technology And Knowledge Management,Volume 2, No. 2, July-December 2010, pp 279-283

[4]. Youquan He, "Decentralized Association Rule Mining on Web Using Rough Set Theory". Journalof Communication and Computer,Volume 2, No.7, Jul. 2005, (Serial No.8) ISSN1548- 7709, USA.

[5]. G. Castellano, A. M. Fanelli, M. A. Torsello. "Log Data Preparation for Mining WebUsage Patterns". IADIS International Conference Applied Computing 2007, pg 371-378.

[6]. M. Agosti, G.M. Di Nunzio and A. Niero "From Web Log Analysis to Web User Pro-Filing" In DELOS Conference 2007. Working Notes. Pisa, Italy, 2007, pp 121–132.

[7]. B. Berendt, B. Mobasher, M. Nakagawa, M. Spiliopoulou "The Impact of Site Structure and User Environment on Session Reconstruction in Web Usage Analysis", WEBKDD 2002, LNAI 2703, pp 159-179, 2003.

[8]. C. W. Cleverdon "The Cranfield Tests on Index Languages Devices". In Readings in Information Retrieval, Morgan Kaufmann Publisher, Inc., San Francisco, California, pp.47– 60, 1997.

[9]. F.M. Facca, P.L. Lanzi "Mining interesting knowledge from Weblogs: a survey", Data and Knowledge Engineering Vol. 53, No. 3, June 2005, pp 225-241.

[10]. D. Nicholas, P. Huntington, A. Watkinson "Scholarly journal usage: the results of Deep log analysis", Journal of Documentation Vol. 61 No. 2, 2005.

[11]. Jan Valdman, Log File Analysis, University of West Bohemia in Pilsen, July 2001.

[12]. http://studyres.com/doc/4204560/a-result-evolution-approach-for- webusage- mining-using-fu...

[13]. https://mafiadoc.com/extraction-of-interesting-patterns-throughassociation- rule-resul-da_59c712131723ddf68040fa71.html

[14]. Santhosh Pininti, Sindh Usha and Teng-Sheng Moh "Detecting Web Attacks Using Multi-Stage Log Analysis."

[15]. K. R. Suneetha "Identifying User Behavior by Analyzing Web Server Access Log File"

[16]. Amruta Ambre, Narendra Shekokar "Insider threat Detection using Log analysis and Event Correlation."

[17]. Kurt Thearling "An Introduction to Data Mining", Computer Society of India Communications, Oct 2006.