Volume: 12 Issue: IV Month of publication: April 2024

DOI: https://doi.org/10.22214/ijraset.2024.59587

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue IV Apr 2024- Available at www.ijraset.com

Log Analyzer and Anomaly Reporting Framework for Proactive System Monitoring.

Reddyvari Venkateswara Reddy[1], Prince Kumar[2], G. Manoj[3], E. Jaswanth[4], M. Eswar Sai Raj Kumar[5]
[1]Head Of The Department, [2]Assistant Professor, [3, 4, 5]Students, Department of CSE(Cybersecurity), CMR College of Engineering and Technology, Hyderabad, Telangana

Abstract: In an era where organizations increasingly rely on intricate software applications, cloud services, and interconnected networks, the significance of analyses cannot be overstated. These tools serve as vigilant custodians of digital footprints, meticulously dissecting the voluminous records encapsulated in log files to extract valuable insights and detect anomalies. As such, log analyses emerge as linchpins in deciphering the meaning behind recorded events, enabling organizations to obtain a more comprehensive understanding of their digital infrastructures and enhance their security posture. This comprehensive exploration aims to untangle the core attributes of log analyses, bringing clarity to their parsing capabilities, the art of information extraction, and the nuanced algorithms that facilitate the conversion of raw logs into actionable insights. Furthermore, against the backdrop of a dynamically evolving cyber threat landscape, the role of log analyses extends beyond conventional diagnostics. These tools have become instrumental in the proactively orchestrated defense against cyber adversaries, empowering organizations to detect and mitigate threats in real-time. Through an in-depth analysis of log analyses and their evolving functionalities, this paper seeks to provide a comprehensive understanding of their Integral role in modern cybersecurity and system management. By elucidating the significance and impact of log analyses, organizations can leverage these tools to fortify their defenses, mitigate risks, and make informed, data-driven decisions in an increasingly complex digital environment.

Keywords: Log analyzer, Anomaly Detection, Log files, Security performance, Events analyzing, Potential reports.

I. INTRODUCTION

In an era where organizations increasingly rely on intricate software applications, cloud services, and interconnected networks, the significance of log analyses cannot be overstated. These tools serve as vigilant custodians of digital footprints, meticulously dissecting the voluminous records encapsulated in log files to extract valuable insights and detect anomalies. As such, log analyses emerge as linchpins in deciphering the meaning behind recorded events, enabling organizations to acquire a more comprehensive understanding of their digital infrastructures additionally, improve and bolster their security posture.

This comprehensive exploration aims to unravel the core attributes of log analyses, shedding light on their parsing capabilities, the art of information extraction, and the nuanced algorithms that facilitate the conversion of raw logs into actionable insights. Furthermore, amidst the setting of a dynamically evolving cyber threat landscape, the function of log analyses extends beyond conventional diagnostics. These tools have become instrumental in the proactively orchestrated defense against cyber adversaries, empowering organizations to detect and mitigate threats in real- time. Through an in-depth analysis of log analyses and their evolving functionalities, this paper seeks to furnish assistance comprehensive understanding of their pivotal role in modern cybersecurity and system management. By elucidating the significance and impact of log analyses, organizations can leverage these tools to fortify their defenses, mitigate risks, and make informed, data-driven decisions in an increasingly complex digital environment.

II. LITERATURE REVIEW

Logging and Log Management is the Authoritative Guide to Understanding the Concepts Surrounding Logging and Log Management, authored by Kevin Schmidt, Chris Phillips, and Anton Chauvin, offers an exhaustive analysis of the fundamental principles and practices associated with logging and log management within the realm of information technology. The book addresses the growing importance of effectively managing log information for diverse uses, encompassing system operations, security, compliance, and incident response.

One of the key themes explored in the literature is the significance of log analysis in detecting and mitigating malicious activities. The authors emphasize the role of logs as valuable sources of information for identifying security incidents and breaches. Through case studies and practical examples, they illustrate how logging technologies, such as syslog-ng, can be deployed in real- world environments to collect and analyses log data effectively.

The literature review highlights the diverse topics covered in the book, ranging from the basics of log data to advanced analysis techniques and visualization methods. It discusses the significance of log storage technologies, statistical analysis, and log data mining in deriving valuable insights from extensive volumes of log data. Moreover, the book addresses the challenges associated with logging laws, compliance regulations, and attacks against logging systems, underscoring the need for robust log management procedures and security measures.

Furthermore, the literature review emphasizes the target audience for the book, which includes systems administrators, security engineers, application developers, and managers interested in enhancing their understanding of logging and log management practices. By providing practical guidance on selecting log analysis systems, planning deployments, and ensuring data normalization and correlation, the book aims to equip readers with the necessary knowledge and skills to effectively manage log data in diverse organizational settings. Logging and Log Management offers a comprehensive resource for information technology professionals seeking to deepen their understanding of logging and log management concepts. Through its thorough scope, pragmatic perspectives,and real-world examples the book serves as an authoritative guide for individuals involved in log analysis, security operations, and compliance efforts within organizations.

III. EXISTING SOLUTIONS

A. Splunk

While Splunk offers powerful real-time insights and search capabilities for log data, its licensing costs can be prohibitivefor small to medium-sized organizations. Additionally, managing and scaling Splunk deployments may require significant expertise and resource.

B. ELK Stack (Elasticsearch, Logstash, Kibana)

Although ELK Stack provides an open-source solution for log management, its setup and configuration can be complex, especially for users without a strong technical background. Additionally, managing large volumes of log data in Elasticsearch may require careful resource allocation and optimization to prevent performance issues.

1) *Gray log:* While Gray log offers centralized log collection and analysis with an open-source platform, its user interface and reporting capabilities may be less intuitive compared to commercial solutions. Additionally, the maintenance and support of graylog deployments may require in-house expertise or reliance on community forums for assistance.
2) *LogRhythm:* While LogRhythm provides advanced analytics and automated response capabilities for security event management,its commercial licensing model may be cost-prohibitive for certain organizations, particularly smaller ones or non-profits. Additionally, the implementation and customization of LogRhythm deployments may require specialized training or consultingservices.
3) *Sumo Logic:* Although Sumo Logic offers a cloud-based log management platform with real-time insights, its pricing model based on data ingestion volume may result in unexpected costs for organizations with fluctuating log data volumes. Additionally, the dependency on cloud infrastructure may elicit apprehensions regarding data confidentiality and compliance for some users.
4) *Apache Flume:* While Apache Flume provides a scalable and distributed log collection system, its configuration and management may require expertise in Apache Hadoop technologies. Additionally, Flume's reliance on Hadoop ecosystem components may introduce dependencies and compatibility issues with other systems and tools.
5) *Logy:* While Logy offers a cloud-based log management service with dynamic field extraction and trend analysis, its reliance on internet connectivity for log data ingestion may pose challenges for organizations with strict security or compliance requirements. Additionally, the absence of on-premises deployment options may constrain its appropriateness for certain environments.

6) *Paper Trail:* Although Paper trail provides a cloud-hosted log management service with live tailing and custom alerts, its reliance on third-party infrastructure may introduce concerns about data ownership and security. Additionally, the lack of advanced analytics features compared to other solutions could potentially diminish its effectiveness in organizations requiring in-depth loganalysis capabilities.

IV. PROPOSED METHOD

Operating on Linux ensures compatibility with a wide range of systems and environments, making the log analyzer accessible to abroad user base within the Linux community. Leveraging Python scripting allows for extensive customization and flexibility in log analysis. Developers can easily tailor the analyzer to suit specific requirements andintegrate additional functionalities as needed.

Integrating Flask enables the creation of a user-friendly web interface for accessing and interacting with the log analyzer.This facilitates ease of use and accessibility, as users can access the analyzer through a web browser without the need for additional software installations. With Flask's capabilities for handling real- time data streams, the log analyzer can provide instantaneous analysis and visualization of log data as it is generated. This allows for timely detection and response to system events and anomalies. Matplotlib's powerful visualization capabilities enable the creation of interactive and dynamic charts, graphs, and plots to represent log data visually. Users can gain deeper insights into log data trends, patterns, and anomalies through intuitive visualization tools.

Python's scalability and performance make it well-suited for handling large volumes of log data efficiently. Combined with Flask's lightweight architecture, the log analyzer can scale to accommodate growing data volumes while maintaining optimal performance.

Running on Linux simplifies the deployment process, as Linux- systems that are rooted in a foundation are extensively utilized and well-supported in various environments. Users can deploy the log analyzer seamlessly on their Linux servers or workstations without encountering compatibility issues.

Being based on open-source technologies like Python, Flask, and Matplotlib fosters a vibrant community of developers, enthusiasts, and contributors. Users can leverage community resources, forums, and documentation for support, collaboration, and knowledge sharing. The log analyzer can effortlessly blend with other Python-based tools, libraries, and frameworks, in addition to existing systems and infrastructures within the Linux environment. This interoperability enhances the analyzer’s versatility and utility for diverse use cases. Operating on Linux offers inherent security benefits, such as robust user permissions, access controls, and security features. Paired with robust coding protocols and dedication to best practices, the log analyzer can help organizations maintain the privacy, authenticity, and accessibility of their log data.

V. PROBLEM DEFINITION

In today's intricate and ever-evolving digital landscapes, organizations face a formidable challenge presented by the generation of extensive and heterogeneous log files from diverse components within computer systems. These logs serve as repositories of invaluable information regarding system activities, errors, and security events.

Manual analysis of these logs compounds the problem, as it is inherently time-consuming, error-prone, and inefficient. Human analysts often find it challenging to match the speed of the relentless influx of log data, leading to take more time in identifying critical events, diagnosing system issues, and addressing security breaches. Moreover, the subjective nature of manual analysis introduces the risk of overlooking important insights or misinterpreting data, potentially exposing organizations to operational disruptions and security vulnerabilities.

Consequently, there is an urgent need for automated and intelligent solutions to address the complexities inherent in log management and analysis. Such solutions should offer capabilities for efficient log collection, normalization, aggregation, and analysis across heterogeneous environments. Moreover, they should leverage advanced techniques like machine learning, abnormality recognition, and pattern recognition to optimize the precision and efficiency of logging, enabling organizations to derive actionable insights in real-time.

VI. LIMITATIONS

Limitations inherent in log analyses include constraints related to the specificity of log file formats and security vulnerabilities. While log analyses excel in parsing specified patterns within log files, their effectiveness may diminish when encountering diverse or non-standardized formats, limiting their applicability across heterogeneous systems. Moreover, certain log analyses may exhibit vulnerabilities that compromise the security of log data, potentially leading to unauthorized access or data breaches. These security shortcomings can undermine the reliability and trustworthiness of log analysis results, raising concerns about the confidentiality and integrity of sensitive information contained within log files.

As such, addressing these limitations is paramount to ensuring the robustness and effectiveness of log analyzer implementations in safeguarding system integrity and data privacy.

VII. SCOPE OF THE PROJECT

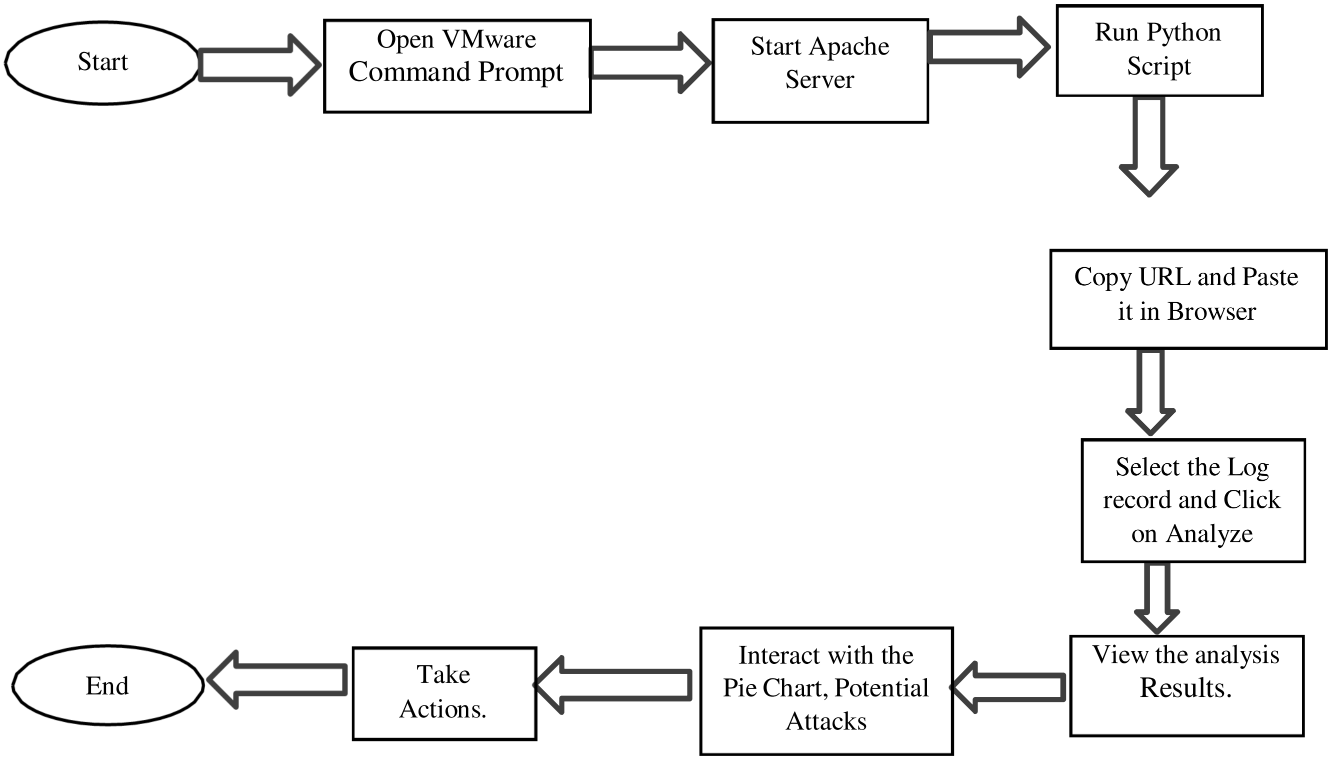
1) *Log Data Retrieval:* The log analyzer will connect to an Apache server running on a Linux system to retrieve log files. The tool will use Python's requests module to establish a connection and fetch log information obtained from the server.
2) *Data Parsing and Analysis:* Upon retrieving the log data, the analyzer will parse and analyses it using Python scripting. It will extract relevant information such as HTTP requests, response codes, client IP addresses, and timestamps from the log entries.
3) *Event Categorization:* The analyzed log data will be categorized into various kinds of events based on predefined criteria. Events may include successful requests, server errors, client errors, access attempts, and more.
4) *Flask Integration:* The log analyzer will be integrated with Flask, a lightweight web framework for Python, to craft an interface for engaging with the tool. Flask will handle HTTP requests and responses, providing a web-based interface for users to record log files and view the analysis results.
5) *Visualization with Matplotlib:* Matplotlib, a Python plotting library, will be familiar with generate a pie chart representation of the categorized events. The pie chart will visually depict the distribution and proportions of different types of events present in the log data.
6) *User Interface Development:* The Flask-based on the user interface enables users to interact seamlessly, granting them the ability to upload log files, initiate the analysis process, and view the resulting pie chart visualization. The interface will be formulated to be intuitive and user-friendly, providing clear instructions and feedback to the user.
7) *Scalability and Performance Optimization:* The log analyzer will be optimized for scalability and performance to handle log files of varying sizes efficiently. Python's multiprocessing capabilities may be leveraged to parallelize data processing tasks and improve overall performance.

VIII. IMPLEMENTATION

1) *Open VMware:* Begin by launching VMware, a virtualization platform, to access the virtual machine environment where the log analyzer will operate.
2) *Start the Apache Server:* Within the VMware environment, start the Apache server to initiate log files encompass data pertaining to the generation of information on web server activities.
3) *Run the Python Script:* Execute the Python script responsible for collecting, parsing, and analyzing the log records generated by the Apache server. The script employs various algorithms and techniques to extract meaningful insights from the log data.
4) *Copy the URL and Paste in the Browser:* Upon completion of the analysis, the Python script provides a URL that directs users toa web-based interface or dashboard displaying the analysis results. Copy this URL and paste it into the web browser to access the interface.
5) *Open the Log Records and Click on Analysis:* Within the web-based interface, navigate to the log record section and initiatethe analysis process. This action may involve clicking on a button or link labelled "Analysis" to trigger the analysis of thelog data.

6) *View Analysis Results:* Upon initiating the analysis, the web- based interface presents users with comprehensive insights derived from the log data. The primary visualization tool used to represent the analysis results is a pie chart, illustrating the distribution of various events captured in the log files. Additionally, the interface highlights potential attacks detected within the log data, providing users with actionable information to mitigate security risks.
7) *Interact with the Pie Chart:* Users can interact with the pie chart to explore different event categories and their corresponding proportions. Hovering over specific segments of the pie chart may reveal additional details or contextual information about the events.
8) *Analyze Potential Attacks:* Additionally, pie chart, the interface displays a catalog of potential attacks identified within the log data. Each attack entry is accompanied by relevant details, such as attack type, severity level, and recommended actions for mitigation.
9) *Take Action:* Armed with insights from the analysis results, users can grasp suitable actions to address security vulnerabilities, optimize system performance, and enhance overall resilience against potential threats.

IX. FLOW CHART

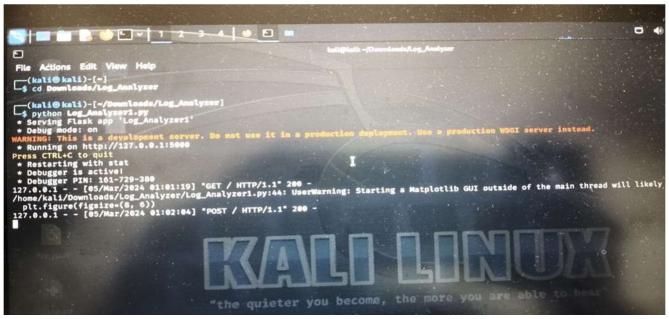

Start
Open VMware Command Prompt
Start Apache Server
Run Python Script
Copy URL and Paste it in Browser
Select the Log record and Click on Analyze
View the analysis Results.
Interact with the Pie Chart, Potential Attacks
Take Actions.
End

X. CONCLUSION

In conclusion, the log analyzer project presents a robust and efficient solution for organizations to effectively analyze log data within Linux environments. By leveraging Python scripting, Flask, and Matplotlib library, the project offers a multifunctional device that effortlessly blends with Apache servers, providing users with actionable insights and visualization capabilities. Through the steps of opening VMware, starting the Apache server, running the Python script, and interacting with the analysis results, users can derive valuable insights from log data and identify possible security risks. Despite the challenges posed by diverse log formats and complex data structures, the log analyzer empowers organizations to optimize system performance, enhance security posture, and make informed decisions based on real-time data analysis. Moving forward, continued research and development in log analysis methodologies will further advance the capabilities of the log analyzer, enabling organizations to stay ahead of evolving cybersecurity threats and effectively manage their digital infrastructures.

XI. RESULTS ANS DISCUSSIONS

Begin by launching the Command Prompt on your system.



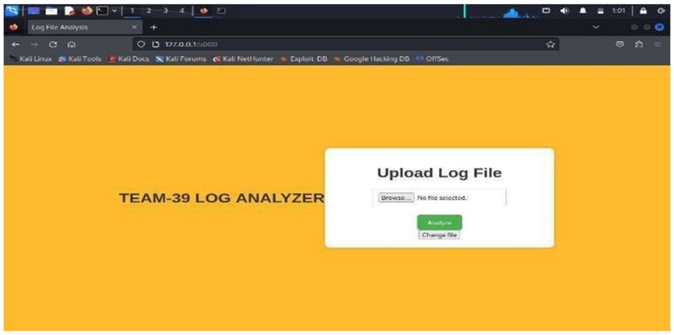
```
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ cd Downloads/Log_Analyzer

┌──(kali㉿kali)-[~/Downloads/Log_Analyzer]
└─$ python Log_Analyzer1.py
 * Serving Flask app 'Log_Analyzer1'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 161-729-380
127.0.0.1 - - [05/Mar/2024 01:01:19] "GET / HTTP/1.1" 200 -
/home/kali/Downloads/Log_Analyzer/Log_Analyzer1.py:44: UserWarning: Starting a Matplotlib GUI outside of the main thread will likely
  plt.figure(figsize=(8, 6))
127.0.0.1 - - [05/Mar/2024 01:02:04] "POST / HTTP/1.1" 200 -
```

kali@kali: ~/Downloads/Log_Analyzer
KALI LINUX
"the quieter you become, the more you are able to hear"

Fig 1. Starting the Apache Server

After the Python script completes its analysis, it will provide a URL as output in the Command Prompt. Use your mouse to highlight the URL, then right-click on it and select "Copy" from the context menu.

Open your preferred web browser (e.g., Google Chrome, Mozilla Firefox) and click on the address bar to activate it. Right- click inside the address bar and select "Paste" from the context menu to paste the copied URL. Press Enter to navigate to the provided URL.

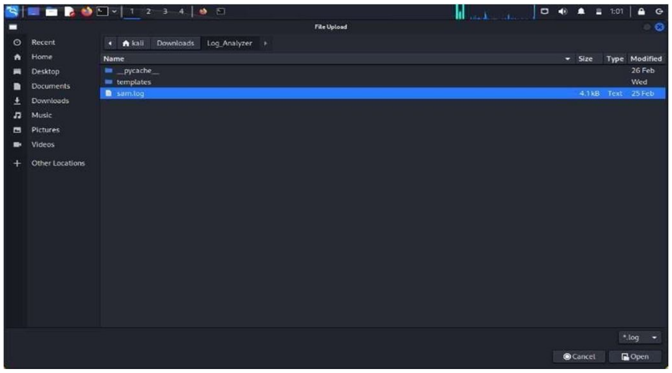


TEAM-39 LOG ANALYZER

Upload Log File

Fig 2.A Web Interface to Upload a File

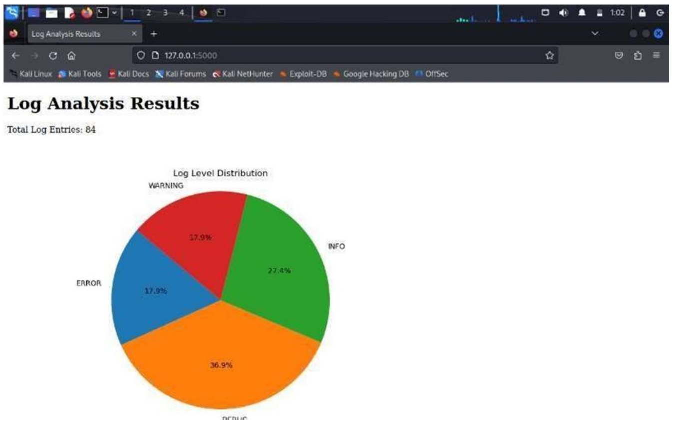
Once the web page loads, you will be presented with options to select the log file you wish to analyses. This may involve browsing your local file system to locate the desired log file or selecting from list of available log files, depending on the interface design.



Name	Size	Type	Modified
__pycache__			26 Feb
templates			Wed
sam.log	4.1 kB	Text	25 Feb

Fig 3. Selecting the Log File

After selecting the log file, initiate the analysis process by clicking on the appropriate button or link on the web page. This action triggers the examination of the selected log file, utilizing the algorithms and techniques implemented in the Python script.

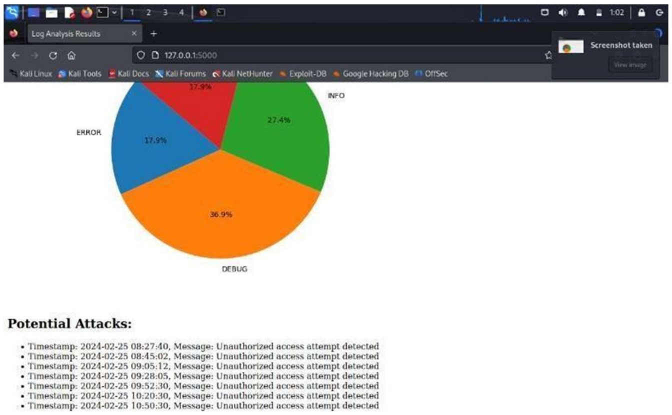


Log Analysis Results

Total Log Entries: 84

Fig 4. Event Analyzing.

Once the examination is complete, the web page will display the results derived from a given log file analysis. This may include visualizations, summary statistics, and actionable insights derived from the principles of log data. Take the time to review and check the results to gain invaluable perspective into system activities, errors, and security events.



Potential Attacks:

- Timestamp: 2024-02-25 08:27:40, Message: Unauthorized access attempt detected
- Timestamp: 2024-02-25 08:45:02, Message: Unauthorized access attempt detected
- Timestamp: 2024-02-25 09:05:12, Message: Unauthorized access attempt detected
- Timestamp: 2024-02-25 09:28:05, Message: Unauthorized access attempt detected
- Timestamp: 2024-02-25 09:52:30, Message: Unauthorized access attempt detected
- Timestamp: 2024-02-25 10:20:30, Message: Unauthorized access attempt detected
- Timestamp: 2024-02-25 10:50:30, Message: Unauthorized access attempt detected

Fig 5. Detecting Potential Attacks.

REFERENCES

[1] Flink, Scalable Stream and Batch Data Processing, Aug. 2017, [online] Available: https://flink.apache.org/.

[2] F. Yang, J. Li and J. Cheng, "Husky: Towards a more efficient and expressive distributed computing framework", VLDB Endowment, vol. 9, no. 5, pp. 420-431, 2016.

[3] Log Parser, Aug. 2017, [online] Available: https://www.elastic.co/products/logstash.

[4] Turn Machine Data into Answers, Aug. 2017, [online] Available: https://www.splunk.com.

[5] S. He, J. Zhu, P. He and M. R. Lyu, "Experience report: System log analysis for anomaly detection", 2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE), pp. 207-218,2016.

[6] Log Analysis for Software-defined Data Canters, Feb. 2015, [online] Available: https://blog.logentries.com/2015/02/log- analysis-for-software-defined-data-centers/.

[7] Operator Rounds Software, Aug. 2017, [online] Available: https://plantlog.com/. E. Analyzer, An IT Compliance and Log Management Software for SIEM, Aug. 2017, [online] Available: https://www.manageengine.com/products/eventlog/.

[8] C. C. Michael and A. Ghosh, "Simple state-based approaches to program-based anomaly detection", ACM Transactions on data and System Security, vol. 5, no. 3, Aug. 2002.

[9] Log Analysis Tool Kit, Aug. 2017, [online] Available: http://www.cert.org/digital.

[10] S. Alspaugh, B. Chen, J. Lin, A. Ganapathi, M. Hearst and R. Katz, "analyzing log analysis: An empirical study of user logmining", LISA14, pp. 62-77, 2014.