International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249 – 8958 (Online), Volume-9 Issue-5, June 2020

Log Analyzer for Update Manager

Shiva Sagar Shetty, Poonam G, Sadhana Ashar

Abstract*: Performing System updates is a very tricky and complicated task. You have to update new software on the same old hardware. This may introduce many errors in the system. For example a windows update on a single PC is generally prone to failures. Further updating a server is more challenging job. As a server can support many Operating systems their components, firmware's and software's etc, there are many possibilities that a server update may initiate many errors in the system. Thus, this paper proposes a tool that can go through all those thousands of log files generated during an update of a server and finds out where exactly the update went wrong. The paper describes about building a tool to perform log analysis from the scratch. It describes about developing parsers at the backend, building a user interface at the frontend and deploying the tool on to network. The proposed work supports building a Docker for the backend. Docker is a platform as a service product that uses OS-level virtualization to deliver software in packages called containers. The tool takes compressed log file as input. The tool analyses the structure of log file and then parses the contents of the log file according to its structure. Finally the proposed tool generates a report in JSON (JavaScript Object Notation) as well as CSV (Comma-Separated Values) format and that report consists of all the required fields to identify and classify the error that was responsible for failure during system update. This report is sent to concerned software development team to fix the defect that occurred in the update.

Keywords: *Log Analyzer, Powerful parsers, well-built backend, Web App framework.*

I. INTRODUCTION

UM (Update Manger) is a smart and efficient tool to keep the system software, firmware and drivers of servers up to date and secure [5]. For example Dell servers have Dell EMC Server Update Utility, HPE has Smart Update Manager and Cisco servers have Cisco UCS Manager. Update Manager helps to minimize the time consuming, error prone and expensive updates. Log file is a file that records either events that occur in an operating system or when software runs. Log file is generated when our Update Manager runs. This paper deals with building a tool called UM log analyzer. UM log analyzer takes input of logs that are generated by running UM on servers. It analyzes these logs and then gives a JSON and a CSV file as an output summarizing the update done by UM.

This paper also deals with building a framework around UM log analyzer so that it can be deployed in the network. To deploy a tool on a network we need a frontend, a middleware and a backend [5]. For frontend, any frontend web framework can be used such as AngularJS, React or Angular 8. These are recommended because of their dynamic and independent structure. The proposed tool is installed on client side (end users browser) which in turn reduces the load at the server side. For middleware, Gunicorn and Nginx frameworks are used which are described in the later sections of this paper. For backend, either Flask or Django frameworks can be used. Both of these are python based web framework. Flask is easy to learn framework and it is used to work with simpler applications. Django framework is usually used for heavy applications. Using any of these frameworks help to efficiently manage and operate servers at the backend.

The proposed tool supports the use of Docker container as all the dependency required by the tool [6] can be supported by Docker container. The advantage of the Docker container is that it can run on any operating system. The system or the machine on which Docker container is deployed does not need any extra software to support this tool. Hence, Docker container consists of all the dependency software such as python script executer, zip extractor etc. This is also further used by the UM log analyzer to process the log files.

II. LITERATURE REVIEW

Biplob and et. al. in their paper [1] have explained about Log Lens application which is an unsupervised machine learning based techniques to discover patterns. They use parsers to get required data and then analyze this using unsupervised ML. Model learns from the training dataset. It reduces the effort of detecting the errors by 1290x times, which will greatly affect the productivity. This application requires no user involvement.

Dileep V1 and et. al. in their paper [2] talks about web usage mining application for predictive analytics in the businesses. This is proven application of log analyzer in today's business. By logging the Web Users data the application tries to understand the customer. Application uses Hadoop (Big Data software) and many other high performance requirement applications which can be a performance bottleneck.

Dileepa Jayathilake and et. al. in their paper [3] proposes a very fast log analyzer. This is achieved by taking into account of the structural design of the logs. It also uses NoSQL databases. Limitation of this paper is that it works only for Structured Logs.

Revised Manuscript Received on May 15, 2020.
* Correspondence Author
Shiva Sagar Shetty*, Computer Science Department, RVCE, Bengaluru, India. Email: shivasagarshetty25@gmail.com
Dr. Poonam G, Computer Science Department, RVCE, Bengaluru, India. Email: poonumghuli@rvce.edu.in
Sadhana Ashar, ISSDC, HPE, Bengaluru, India. E-mail: sadhana.ashar@hpe.com

Max Landauer in his paper [4] propose an application of log analyzer to overcome security concerns. Dynamic anomaly detection is used to detect flaws in cyber security. The system works on real time logs. Limitation of this paper is, it does not work 100% correctly gives false alarms around 2% of the time.

Above literature review speaks about various applications where log analyzer is used. A few advantages discussed in the above literature can be incorporated in UM log analyzer. For example UM log analyzer can be made faster by taking advantage of the structured pattern of the logs as mentioned in paper [2]. Also, the end users of UM log analyzer are not layman but software developers. Hence, the proposed tool is fast and accurate rather than heavy and complex. This tool is easily accessible inside the network where it is deployed for security concerns through web browser. It also supports concurrent access by multiple users. The further sections in this paper discuss the incorporation and realization of above mentioned features in the proposed tool.

III. PROPOSED APPROACH

UM log analyzer is built with the aim to increase the productivity of the software developers, who are responsible for fixing the errors that occur during software update. Software developers spend a lot of time checking log files to figure out where the error occurred. In order to overcome this problem the proposed tool helps software developers to figure out the exact errors effectively and efficiently.

The objectives of this tool are:

- To provide easy web enabled access to the end user.
- To provide hassle free concurrent access to the tool by multiple users
- To maintain data consistency across the framework.
- To ensure availability of the tool to users.
- To enable report generation in JSON and CSV format by processing and analyzing the user provided log files.

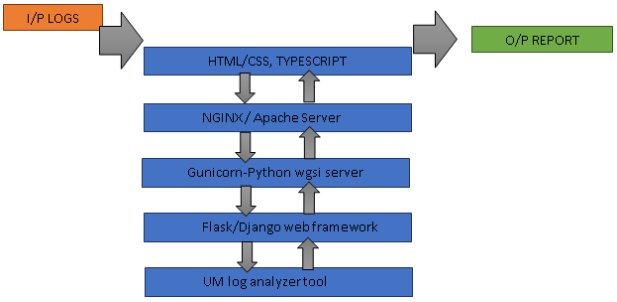

Fig. 1.Architectural Diagram of the Web Framework.

In order to satisfy the above objectives, a proper web enabled framework is built around the UM log analyzer. The figure 1 shows the architectural diagram of the proposed framework. The proposed architecture consists of following components described as follows:

- HTML/CSS, typescript: Provides a very dynamic and flexible frontend design. We can choose Angular JS, React or Angular 8 for this.
- Nginx / Apache server: These are web servers. These are used to host your frontend consisting HTML/CSS and scripts as a web application. Nginx is a web server which can also be used as a reverse proxy, load balancer, mail proxy and HTTP cache [4].
- Gunicorn: It is a middleware. It is used mainly to balance the load at the server. Gunicorn is pre fork worker model. It will create many sub processes called workers so that multiple end users can send request to the server at the same time.
- Flask/ Django web framework: This is the backend of web framework. It will be running 24*7. It performs the following tasks: Getting request from the frontend, receiving the logs as input, running the UM log analyzer tool, generating the output, sending the output as the response back to the frontend.
- UM log analyzer: This is a tool coded in python and runs powerful parsers on the input logs to generate the report. Figure 2 shows the descriptive flow diagram of UM log analyzer.

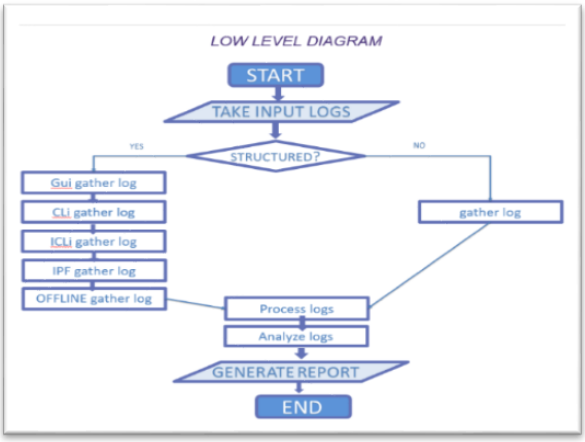

Fig. 2.Descriptive diagram of UM Log Analyzer

UM log analyzer uses of the structure of the logs. Structure of the logs refers to the way in which log files are arranged in a directory. It creates multiple threads to parse parallelly into different folders in a structured log. If logs are unstructured it parses the log records one by one.

IV. RESULT AND DISCUSSION

The paper describes the successful realization and deployment of the proposed tool. After deployment of the complete framework we are able to achieve the following.

- Availability: The system would be up and running at all time. The tool is deployed on a server which is always running.
- Speed: The system takes advantage of the structure of logs. To test this, logs of sizes between 100-150 MB were taken and processed using log analyzer and the proposed tool provided the following results as shown in Table 1. Time taken for processing structured logs is less compared to unstructured logs.

Retrieval Number: E9507069520/2020©BEIESP
DOI: 10.35940/ijeat.E9507.069520
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Table- II: Time taken to process Structured and Unstructured logs

TIME TAKEN FOR STRUCTURED logs (in secs)	TIME TAKEN FOR UNSTRUCTURED logs (in secs)
26.061	28.774

Ease of access: End users can access it by just a click of a button. On their browser.

- Security: Separating frontend and backend will be a added security bonus. Also the most of resource utilization is on client side due to frontend.
- Multi-User: Up to five end users can access the tool simultaneously due to the support from gunicorn.

V. CONCLUSION

This paper explains in detail the development and deployment of a web enabled log analyzer framework. The tool is accessed by multiple users concurrently and is available all the time. End user can provide the log files as input to the proposed tool and receive the output in the form of JSON report. The Limitation of the system is that it performs static log analysis and unable to perform real time analysis. With increased progress in machine learning, predictive log analysis is seen as the future path. Incorporation of machine learning to the proposed framework will allow it to process logs dynamically and also provide solutions for the errors occurred.

Log Analyzer is proven to be very powerful tool from the past. Log analyzer is seen as futuristic using tool because of the rise in log processing machine learning algorithms and powerful CPU's. These log analyzers make the tedious task of analyzing or reading the huge stacks of logs easier. This in turn helps to improve the productivity of the team effectively. Moreover customers are more selective than ever when it comes to the applications they use. With such a large pool of alternatives at their disposal, they can easily turn to one of your competitors if they're not satisfied with your product or service. You must deliver an excellent user experience that, looking beyond functionalities, boils down to a stable and efficient application with regular updates.

REFERENCES

1. Biplob Debnath, NEC Labs. America, Inc., Princeton, NJ, USA, 'Log Lens: A Real-Time Log Analysis System', *IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, July 2018.
2. Dileep V ,Nehru Institute of Technology, Coimbatore, 'A Study on Log Parser Analysis and Error Detection using Big Data', *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*,2018.
3. Dileepa Jayathilake, 99X Research, 99X Technology, Colombo, Sri Lanka,'Towards structured log analysis '*Ninth International Conference on Computer Science and Software Engineering (JCSSE)* June 2012.
4. Max Landauer ,Austrian Institute of Technology, Austria, Dynamic log file analysis: An unsupervised cluster evolution approach for anomaly detection, *Science Direct Journal*, 2018
5. Carey Wodehouse, "A Beginner's Guide to web Development," [Online] Available: 'https://upwork.com' ,2018.
6. O.S Tezer, "Deploy Python WSGI Apps Using Gunicorn HTTP Server Behind Nginx," [Online] Available: 'https://www.digitalocean.com/community/tutorials/how-to-deploy-python-wsgi-apps-using-gunicorn-http-server-behind-nginx/', Dec.2013.
7. Babak Bashari Rad, Harrison John Bhatti, Mohammad Ahmadi, "An Introduction to Docker and Analysis of its Performance", *'IJCSNS International Journal of Computer Science and Network Security'*. March 2017.

AUTHORS PROFILE

Shiva Sagar Shetty, is currently a student of R.V college of engineering in Bengaluru. He has published three research papers in IEEE Xplore through various conference held by IEEE in the fields of Network on Chips and Network Buffer Optimization. He is currently an intern at Hewlett Packard Enterprise.

Dr. Poonam Ghuli, has been working as Associate Professor in the Department of Computer Science and Engineering over the past fifteen years at RVCE, Bengaluru, India. She has a couple of papers published in reputed Journals and conferences. She is working in the area of Data Analytics. She is actively involved in many research and consultancy work supported by renowned companies such Citrix, Cisco, Samsung etc.

Sadhana Ashar, Working with Hewlett Packard Enterprise for the last 10 years in the Server Domain and Server Update Technologies. Graduated from the PES Institutions with a degree in Information Science