International Journal of Research in Engineering, Science and Management
Volume-1, Issue-12, December-2018
www.ijresm.com | ISSN (Online): 2581-5792

Automated Monitoring and Service Recovery Using Real-Time Log Analysis

Aarti Hatkar[1], Asawari Hanchate[2], Pornima Divekar[3], Krutika Latkar[4]
[1,2,3,4]Student, Dept. of Computer Engg., JSPM's Imperial College of Engineering and Research, Pune, India

***Abstract*: In today's modern era due to extensive use of digital application log file analysis has become a necessary task to track system operations or user behavior and acquire important knowledge base on it. Log files or logs in computing are the files for keeping record of the events that occur in operating system. Log files contain large amount of valuable information about the system operation status, usage, user behavior analysis etc. The main objective of proposed system is to design an application for server-log analysis and use the big data tools to get the result which will be useful for system administrator to take proper decision. The widely used big-data tool – Apache Spark makes it possible to provide a unified platform with Spark streaming and in memory computing capacity in order to process logs in high available, stable and efficient way. Statistic results generated in the form of charts, bar graphs and pie-charts are displayed on the dashboard system. This will ultimately help in greater expansion of business and a company that will have such data to its disposal and ready to use.**

***Keywords*: Hadoop, Server Logs, Apache Spark Python Flask, Kafka, Flume**

1. Introduction

Today in world of big data and complexity of data increases with fast pace Here; log analysis is a savior and science seeking to make sense out of computer-generated complex, unreadable logs (also called as records). The process of creating such records is called Log Processing. Logs provide us with essential information on how our system is behaving. However, the content and format of logs varies according to the different services and among different components of the same system.
Reasons to perform log analysis are as follows:

- Purpose of security policies
- Compliance with audit or regulation
- System troubleshooting
- Forensics analyzation (during investigations or in response to subpoena)
- Capturing online user behavior

Logs are generated by network devices, operating systems, and applications. Logs may be directed to files and stored on disk. Log messages should be interpreted with respect to the internal state of its source (e.g., application) and implement the security or operations related events (e.g., a user login, or systems error).

A. *Objective*

The objective of this project is to monitor the real-time streaming of server logs and provide recovery services. The server is been created with the help of Python Flask and then analyzation is done with Hadoop tools like Spark, Kafka, Flume. Result generated is been stored in database and displayed on website

2. Existing system

Costly monitoring tools, big display screens and telephones line. Tool displays pop-up alert, upon the event (i.e. when a link or server, application goes up, or down) Monitoring Associate (person) manually needs to notice, evaluate and call the Technical Engineer to check and remediate. Technical engineer then will intervene to see in details. Limitation of the existing system:

- Possibility of human errors.
- Cost of operating is high.
- Application tool dependency and licensing cost

3. Proposed system

All services will emit logs to files, which be shared to common location Server will pick files and use real-time stream to connect and process log files and hold result. Dashboard will consume process result and will Show on dashboards the best way to solve and implement server log analysis. Spark platform allows us to store the files on the disk cheaply. Log data can be used to monitor the company server improves our business and enhances the intelligent functioning. The implementation of certain projects like log processing can be done using Spark streaming.

A. *Apache spark*

Apache Spark is a fast ad general purpose cluster computing tool. It provides a new abstraction called as RDD (Resilient Distributed Dataset), which helps in fault tolerance and keeping data in memory. This results into the significant speed of the log analysis. It works much faster as compared to the traditional large-scale-data set frameworks. Spark has some additional key features as:

- Provides APIs in Scala, Java, and Python, with support for other languages like R

- Integrates with the Hadoop ecosystem and data sources (HDFS, Amazon S3, Hive, HBase etc.)
- Can run on clusters managed by Hadoop and can also run standalone.
- Spark introduces the concept of an RDD (Resilient Distributed Dataset), a fault-tolerant, distributed collection of objects. An RDD can contain any type of object and is created by loading an external data items.

RDDs support two types of operations:

- Transformations are operations (Map, filter, join, union, and so on) that are performed on an RDD and which generates a new RDD containing the result.
- Actions are operations (Reduce, count) that return a value after running a computation operation on an RDD.

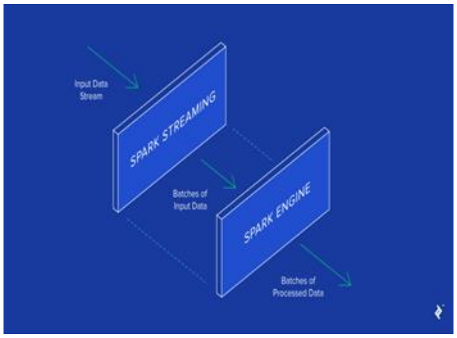

Fig. 1. Apache Spark Streaming

The whole process can be divided into the sections to understand the flow of log.

- Introduction to Apache Spark
- Importing data
- Exporting data
- Log analyser application work

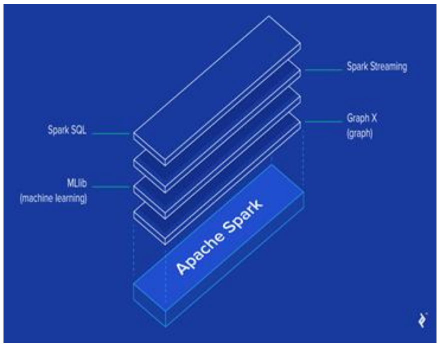

Fig. 2. Apache Spark

B. Spark streaming

Apache Spark is the best way to solve and implement server log analysis. Spark platform allows us to store the files on the disk cheaply. Log data can be used to monitor the company server, improves our business and enhances the intelligent functioning. The implementation of certain projects like log processing can be done using Spark streaming. Spark Streaming supports real time processing of streaming data, such as web server log files (e.g. Apache Flume and HDFS/S3), social media like Twitter, and various messaging queues application like Kafka. Spark Streaming receives the input data streams and divides the data into different batches. Next step, they gets processed by the Spark engine and generate final results in batches. The Spark Streaming API closely resembles with the Spark Core, makes it easy for programmers to work in the both batch and streaming data.

C. Architectural view

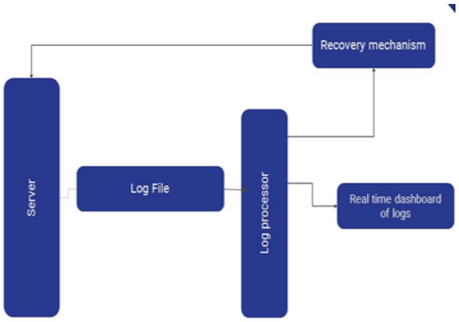

Fig. 3. Proposed system mechanism

D. Components

1) kafka

Apache Kafka is a tool that allows handling large volume of data with using hardware set. It is used to create subscription based messaging functionality. It can process many events per day (Up to 1 Trillion events a day) and process them.

2) Flume

Apache Flume is an excellent tool in collecting data then aggregate, transform volume of data into the file system. Flume automatically maintains a steady flow of data.

3) Flask python

It is a framework for creating web applications. It is Python based micro framework. Website (Front end) It Includes dashboard, chart, statistics.

4) Database

Mongo DB used as a database here. All the daily log records are stored here after processing done on them. These are no more complex form of logs but the meaningful data deposited over the database after streaming.

4. Application

The Spark technology can be used in various fields and has different applications according to the type of requirement. It is highly used in: Memory data mining on Hive data (Conviva), Predictive analysis (Quantified), City traffic prediction (Mobile Millennium), Twitter spam classification (Monarchy), Collaborative filtering via matrix factorization. In the game industry, processing and discovering patterns from the potential fire hose of real-time in-game events and being able to respond to them immediately. In the e-commerce industry, real-time transaction information can be passed to a streaming clustering algorithm like k-means or collaborative filtering like ALS.

Results such as customer. Comments or product reviews, and used to constantly improve and adapt recommendations over time. In the finance or security industry, to find a fraud or intrusion detection system .It could achieve top results by processing huge amounts of archived logs.

5. Conclusion

This project helps in understanding requirement of the respective organization and performs log analysis to sense out the useful and meaningful information out of the complex logs. Spark helps to simplify the challenging and intensive task of processing high volumes of real-time data, both structured and unstructured. Spark brings Big Data processing to the ease and efficient data analysis. The whole idea behind this project meets the key features like enhanced log mining, evaluating logs for useful on formation to provide it to the customer avoids security issues and improve business revenue with it.

References

[1] H. Andrews, “Theory and practice of log_le analysis." Technical Report524, Department of Computer Science, University of Western Ontario, May 1998.

[2] Lamport, “Time, Clocks, and the Ordering of Events in a Distributed System." Communications of the ACM, Vol. 21, No. 7, July 1978

[3] M. J. Guzdial, “Deriving software usage patterns from log _les." Georgia Institute of Technology. GVU Center Technical Report. Report

[4] Tec-Ed, Inc., “Assessing Web Site Usability from Server Log Files WhitePaper." http://citeseer.nj.nec.com/290488.html

[5] Osmar R. Zaane, Man Xin, and Jiawei Han, “Discovering web access patternsand trends by applying OLAP and data mining technology on web logs." In Proc. Advances in Digital Libraries ADL'98, pp. 19-29, Santa Barbara,CA, USA, April 1998.

[6] J. Valdman, “SOFA Approach to Design of Flexible Manufacturing Systems.” Information Systems Modeling Conference (ISM’00), Roznov pod Radhostem, 2000.

[7] A. Andreasson, P. Brada, J. Valdman: “Component-based Software Decomposition of Flexible Manufacturing Systems”. First International Carpatian Control Conference (ICCC’2000), High Tatras, Slovak Republic, 2000.

[8] J. Rovner, J. Valdman: “SOFA Review – Experiences From Implementation.” Information Systems Modeling Conference (ISM’01), Hradec nad Moravici, 2001.

[9] J. Valdman: “MAN at The University of West Bohemia.” Invited speech at First Austrian International Network Academy Conference (AINAC), Inssbruck, Austria, 2001.

[10] [DCSE–1] J. Valdman: “Means of Parallelization in Higher Programming Languages.” A study report, Pilsen 2000.

[11] [DCSE–2] J. Valdman: “www. Proxy & Cache.” A study report, Pilsen 2000.

[12] [DCSE–3] P. Hanc, J. Valdman: “SOFA Techreport.” Technical report of the Department of Computer Science And Engineering, University of West Bohemia, Pilsen 2001.