Proactive montoring of server logs to prevent instant shutdown of the server using ElasticSearch

Prajakta Sonawane[1], Kanchan Pingale[2], Manasi Gawali[3]
Pune Institute Of Computer Technology, Dhankawadi, Pune

Abstract— A server is defined as a computer program which manages access to a centralized resource or a service in a network. As the definition is stated, a server is the heart of the any organization to work smoothly. But sometimes unexpected shutdowns of the server can cause various problems in the organization which may lead to unhealthy situations. The aim of our project is to avoid this instantaneous shutdown of the servers through elastic search. The server log files are parsed to make the files more humanly readable and then the concept of elastic search comes into play. The runtime data which is being generated is processed , parsed and stored into the clusters of elastic search. The processed data is analyzed and user is notified through dashboard notification or email notification whether the system server will shutdown or not hence saving the time.

Keywords— Data analysis, eastalert, elasticsearch, kibana, logstash, prediction

I. INTRODUCTION

Organizations run multiple servers to deliver business pre-eminent services for their end users. Some of them include database servers, core app servers, caching servers, web servers, etc. The performance of each of these servers is very pivotal because even if one of the servers fail, then it influences the delivery of business services. Apart from the panic and flurry of activity to get things back online, a good number of employees would be twiddling their thumbs and bosses would be wondering how much this is going to cost the company. Data loss is the focal problem and recovery of the same is a time-consuming process which requires various algorithms to be incorporated. Therefore it is indispensable to know any performance issues proactively so that they are identified at the early stage and fixed before they turn big and hamper business.

Data analysis is cleansing ,inspecting and modeling of data and has become crucial these days due to the increasing risks and unexpected outcomes as in case of servers. Lot heard about of server crash due to the reasons such as n number of users connected to the server, many users using services at the same time provided by server etc .This can be fatal and may lead to major problems. The unexpected shutting down of server has become a common issue and needs to be looked upon. In such a case, logs of various servers like IAS server , SQL server, apache server is taken and parsed so that can be converted to a generic JSON format from where later dumped into a search engine named elastic search. As Elasticsearch is flexible, consistent and is able to achieve fast search responses. Instead of searching directly, it searches an index. Later by using analysis algorithms detects threats if any and notifies. Notifications can be either Dashboard notifications or email notifications.

II. PROPOSED METHOOLOGY

At present , there are hundreds of millioms of computers, laptops or any electronic devices in not only extravagant but small companies as well generating huge amount of logs. The generation of such huge amounts, we can say, nearly about thousands of billions of information is served by the log files on everyday basis. To predict the business value of any organization it is necessary to store and analyse these log files. Our Oracle databases may not be helpful in storing and analysing such bllions of log events that are being generated. If Oracle databases are taken into consideration then this humungous data must be stored in Oracle clusters which are lowers in performance after scaling and some of the SQL features may be lost.

The amount of information generated on the internet is being proliferated. Traditonal centralized search engine needs to retrieve such huge amount of information quickly which is getting more and more difficult day by day, hence the earch engine system should have distributed processing capabilities, according to the need to deal with the increasing information, constantly scaling up the system to improve the system's ability to process information. Therefore, developing a system to take care of this situation which has distributive capabalities turns out to be more useful. To search, retrieve and analyse the log data in real time, we need such a system which would not put pressure on the Oracle clusters. It has been observed with experience by major companies like Yahoo, Facebook, etc. traditional relational database systems cannot handle big data. This gave rise to the concept of Lucene index.

Lucene is a full text search ,open source library. It is currently the most popular Java based information retrieval lbrary. Lucene is a full text , open source , distributed and RESTful search engine. Elasticsearch is based on lucene indices. Designed for cloud computing, it is stable, fast, easy to install and a reliable search engine. The main property of Elasticsearch is that it is possible to search data or retrieve it in near real time. It supports JSON based data over HTTP index. It is implemented in Java yet there are many clients including PHP, Ruby, Pearl, Scala, .NET, Python, JavaScript, Erlang and Closure. Django, Couchbase and SearchBox are integrated into ElasticSearch. MongoDB, CounchDB, RabbitMQ, RSS, Sofa, JDBC, FileSystem, Dropbox, ActiveMQ, LDAP, Amazon, SQS, St9, OAI and Twitter can be imported into Solr directly.

ELK Stack, where ‘ELK’ is the acronym for three open source projects: Elasticsearch, Logstash, and Kibana.

A. Intoduction to ElasticSearch, Logstash, Kibana, Filebeat and ElastAlert

Elasticsearch is a distributed data search engine. It is mainly based upon Apache Lucene. It can fetch and retrieve complex data which are being represented using JSON objects. It can store data in JSON format as well as retrieve it in the same format. It is based on the concept of inverted index. The terminologies used for storing the data in the Elasticsearch instance are documents, shards, replicas, indices, cluster and nodes. The smallest unit of elasticsearch is the Document and the largest unit is an Index. An index can have multiple shards. A shard in Elasticsearch is a Lucene Index. The maximum number of documents in a Lucene index is fixed and depends on the version of Lucene used. As of Lucene 5843 the limit is 2147483519. When we create an ES index, we can provide the number of shards it can have. Each shard in itself is an independent index and can be hosted on any node of the Elasticsearch Cluster. In Elasticsearch, all data in every field is indexed by default. That is, every _eld has a dedicated inverted index for fast retrieval. Elasticsearch refreshes its index in a default time interval of 1 second which helps in achieving Near Real Time search. This interval can be changed by modifying its value in elasticsearch.yml setting file in /config folder. Elasticsearch is mainly designed keeping cloud environment in mind. Elasticsearch automatically detects all the nodes in the network having same cluster name. Cluster name, node name, number of shards and replicas as well as other settings can be done in elasticsearch.yml file. When a document is indexed, several inverted indices gets created. Each inverted index contains the field name, corresponding value of the field in given document and a pointer to the document.

Logstash is a server-side data processing pipeline that ingests data from multiple sources simultaneously, transforms it, and then sends it to a "stash" like Elasticsearch. It ease overall processing independent of the data source, format, or schema. Kibana is an open source analytics and visualization platform designed to work with Elasticsearch. Kibana is used to search, view, and interact with data stored in Elasticsearch indices. We can easily perform advanced data analysis and visualize your data in a variety of charts, tables, and maps.Kibana makes it easy to understand large volumes of data. Its simple, browser-based interface enables you to quickly create and share dynamic dashboards that display changes to Elasticsearch queries in real time.Kibana lets users visualize data with charts and graphs in Elasticsearch. Filebeat is part of the Elastic Stack, i.e it works seamlessly with Logstash, Elasticsearch, and Kibana. It is a lightweight shipper for logs. Elastalert is open sourced, developed by github used as an alerting tool .ElastAlert is developed to automatically query and analyze the log data in our Elasticsearch clusters and generate alerts based on easy-to-write rules.

B. Dataset Requirements

The datasets required for the proper execution of the project is the log files from various web servers and database servers such as:

Database servers:
1. MySQL
2. MongoDB

Web servers:
1. Apache
2. Tomcat
3. IIS server

C. Functional Requirements

Functional requirement describes what the system should do.
Accordingly, the functional requirements for this system are
1. Parses the log files of different servers
2. Create one log format for all the server logs
3. Create the clusters and put it in elastic search
4. Search for threats through elastic search
5. Show analysis
6. Predict and notify server shutdown

D. Non-functional Requirements

Non functional requirements is any requirement which specifies how the system functions. Accordingly, the non functional requirements for this system are
1. Parsing the log files efficiently in the required file format
2. Searching the right data at the right time .
3. The searching must be done fast.
4. Replication and sharding must be done properly.
5. Effectivity, efficiency, compatibility and security are the major non-functional requirements

III. SYSTEM ARCHITECTURE

In Fig. 1 log files from different servers like database servers and web servers are taken into consideration. The log files are parsed into JSON like documents which are fed to elastic search. The JSON formatted parsed log files are fed to elastic search where they would be stored in the form of clusters and indexed because of which it would be easier to search and analyzing of the data would be simpler. The online threats would be detected using the data collected and analyzed in the analysis dashboard which would be notified to the user and server shutdown would be avoided.

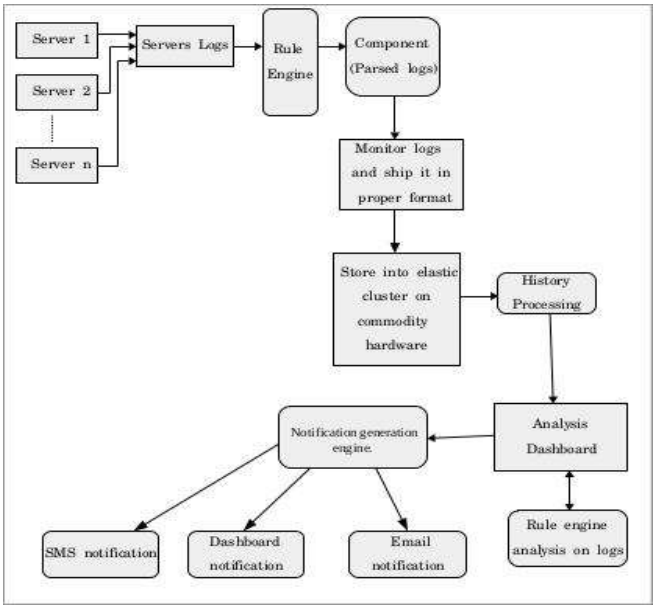

Fig. 1 System Architecture

IV. ALGORITHMS USED

There are two algorithms used while developing the system.

A. Parser

1) Define the log file to be parsed.
2) The dictionary with the attributes of the log files prepared.
3) Remove the unnecessary lines of the logs such as comments.
4) Parse each line of the log files according to the attributes of the logs.
5) Store the parsed logs in JSON format in a file.
6)Since the log can be of variable size and can be huge , a folder structure is prepared according to the date and time for efficiency.

B. Elastic Search Algorithm

1) The logs in the JSON format are fed to elastic search.
2) Basic index structure is considered.
3) The documents are split in a dictionary format with words, its frequency and which document it belongs to as attributes.
4) After some simple text processing (lowercasing, removing punctuation and splitting words), we can construct the "inverted index".
5) The inverted index maps terms to documents (and possibly positions in the documents) containing the term.
6) Since the terms in the dictionary are sorted, we can quickly find a term, and subsequently its occurrences in the postings-structure.
7) End.

V. IMPLEMENTATION

The modules are split up into

1) Parser

The log files from different servers like database servers and web servers are taken into consideration. Using python, the log files are parsed into JSON like documents which are fed to elastic search.

2) ElasticSearch

The JSON formatted parsed log files are fed to elastic search where they would be stored in the form of clusters and indexed because of which it would be easier to search and analyzing of the data would be simpler.

3) Notification Dashboard

The online threats would be detected using the data collected and analyzed in the analysis dashboard which would be notified to the user by writing rules using ElastAlert and server shutdown would be avoided.

A. Preparation of Data

The data that we have considered throughout the building of the project are.

1) Apache Log files

Unparsed log snippet:

127.0.0.1 - - [20/Sep/2017:20:34:43 +0530] "GET / HTTP/1.1" 200 871 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:49.0) Gecko/20100101 Firefox/49.0

2) IIS log files
Unparsed log snippet:
#Software: Microsoft Internet Information Services 8.5
#Version: 1.0
2017-02-03 20:04:07 10.0.0.4 GET / - 80 - 120.138.116.242
Mozilla/5.0+(Windows+NT+6.3;+WOW64;+rv:51.0) +Gecko/20100101+Firefox/51.0 - 200 0 0 62

B. Processing of Data

The types of servers logs taken are , IIS and Apache server logs .These logs are studied thoroughly and the attributes are considered. As IIS server logs has 15 atttributes which include date, time, cs-method, cs-uri-stem, cs-username, c-ip, cs(User-Agent), cs(Referer), sc-status, s-ip, sc-substatus, sc-win32-status, time-taken, cs-uri-query, s-portand apache server logs include date, time, cs-method, cs-uri-stem, cs-username, c-ip, cs(User-Agent), cs(Referer), sc-status, user-identity, response-size.The common attributes from both the logs are taken into consideration .Further, parsing of logs is done using python language in JSON format .The parsed logs are then forwarded to filebeat. Filebeat which is a text shipper forwards the parsed logs to logstash .Logstash enriches and transforms the data to the elasticsearch where alerting is done . Alerting can be done using elastalert . Alerts are sent through gmail SMTP , SMS etc. Alerts consists of information such as source of the log file , error type and hostname .

C. Output Snapshots

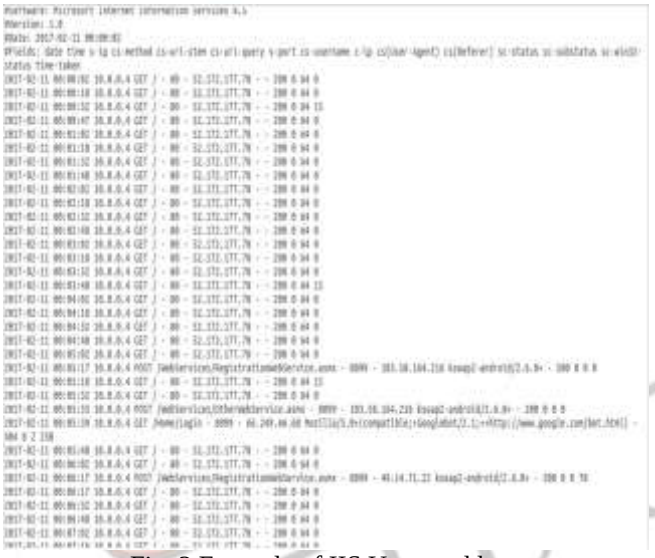

Fig. 2 Example of IIS Unparsed logs

Fig. 3 Example of IIS Parsed Logs

Fig. 4 Example of Kibana Dashboard

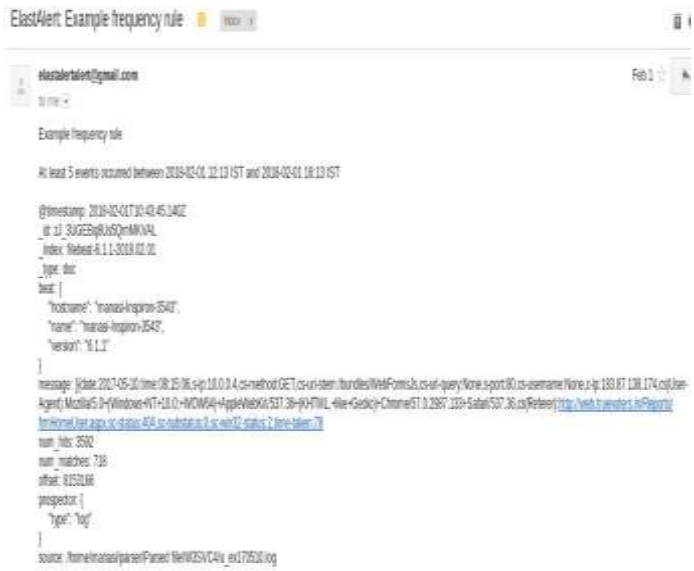

Fig. 5 Example of an email notification

VI. CONCLUSIONS

Thus, developing a system that will prevent unexpected shutting down of system is helpful for all the organizations for their effective functioning. The re-work that needs to be done due to the server failure would be saved hence saving the most precious resource that we all always lack- TIME.

REFERENCES

[1] O. Kononenko, O. Baysal, R. Holmes and M. Godfrey, "Mining modern repositories with elasticsearch", Proceedings of the 11th Working Conference on Mining Software Repositories - MSR 2014, 2014.
[2] Risto Vaarandi, "A Data Clustering Algorithm for Mining Patterns From Event Logs," in Proceedings of the 2003 IEEE Workshop on IP Operations and Management, pp. 119-126.
[3] Pragya Gupta1, Sreeja Nair, Survey Paper on Elastic Search, Paper ID: NOV152583 , Volume 5 Issue 1, January 2016.
[4] Jun Bai. Feasibility analysis of big log data real time search based on hbase and elasticsearch. In Natural Computation (ICNC), 2013 Ninth International Conference on, pages 1166{1170. IEEE, 2013.
[5] Yicheng Zheng, Feng Deng, Qingmeng Zhu, and Yong Deng. Cloud storage and search for mass spatio-temporal data through proxmox ve and elasticsearch cluster. In Cloud Computing and Intelligence Systems (CCIS), 2014 IEEE 3rd International.