SCALABLE LOG ANALYTICS AND VISUALISATION

Abdullah Abdullah | Vasigaran Senthilkumar | Ilias Merentitis
aabdull | vsen | iliasme @kth.se

GROUP 10
30 October 2022

ABSTRACT

In today’s world of big data and distributed systems, every organisation ranging from small scale to large scale have multiple systems (microservices based) and infrastructure running on different cloud providers like AWS, Azure etc. For running businesses reliably and effectively, organisations need to know about the performance of their systems and infrastructure from time to time. Over time, several strategies have been adopted to inspect system performances. One of them is analysing system and application level logs (web server, mobile requests etc) and applying predictive analytics to the log data. The amount of log data depends on systems and the organisation's infrastructure and is usually massive.

1. INTRODUCTION

Thanks to big data, distributed computing and open-source analytics frameworks like SPARK, performing scalable log analytics on massive log data sets is now doable. Our task for the project is to take advantage of the technologies mentioned and analyse production web server logs from one of the biggest organisations in the world “NASA” and present a useful graphical analysis of the server logs like **frequency of HTTP status codes, number of daily requests per host, number of error responses, listing the endpoints with most errors received time series analysis of errors.**

2. TOOLS

The tools used for the basis of the project are the Python language together with PySpark, written in the Jupyter Notebook. Apache Spark and Spark SQL will be used for data processing and exploratory analysis.

In order to visualise the graphical analysis of the logs, a Javascript GUI is also built using Javascript chart libraries (d3.js).

3. DATA

For our project, we have used two open-source datasets [1] provided by NASA. The two datasets contain two months’ worth of HTTP requests to the NASA Kennedy Space Center WWW server in Florida, which amount to around 3.46 million of HTTP requests (logs). The logs are in ASCII format with one line per request. Each request contains information such as **host, timestamp, request, HTTP reply code, and bytes in the reply**. The datasets contain requests from July 1st 1995 till 31st August 1995.

4. METHODOLOGY AND RESULTS

4.1 SETUP

The first step was retrieving the NASA open-source datasets [1]. Once the data was downloaded, we loaded them into the development environment, where all the necessary data preprocessing took place.

4.2 DATA PREPARATION AND CLEANING

During the data preprocessing phase, our goal was to extract structured attributes from the dataset and meaningful information from each log message. To achieve that, data cleaning and parsing had to be conducted. By using regular expressions we could transform entries from the log into a more understandable structure from where we extracted information and insights. For each distinct extracted entry a corresponding column was created, in which all the values matching that entry would be appended sequentially. Doing this for all entries meant that at the end of the process we had successfully converted the log into a data frame from which we could work easier.

Another necessary step was handling null values. Per entry, if the entire row is null then it is dropped, else if the row is relevant we replace the null values with zero. For example, from our datasets, we got around 33 thousand null values for content size (response size) which had to be replaced with zero.

4.3 DATA ANALYSIS

The next step was to do exploratory data analysis and extract useful data. We used SparkSQL functions like groupBy, sort, max, min etc to analyse our data. Some of the useful data retrieved for our results are as follows:

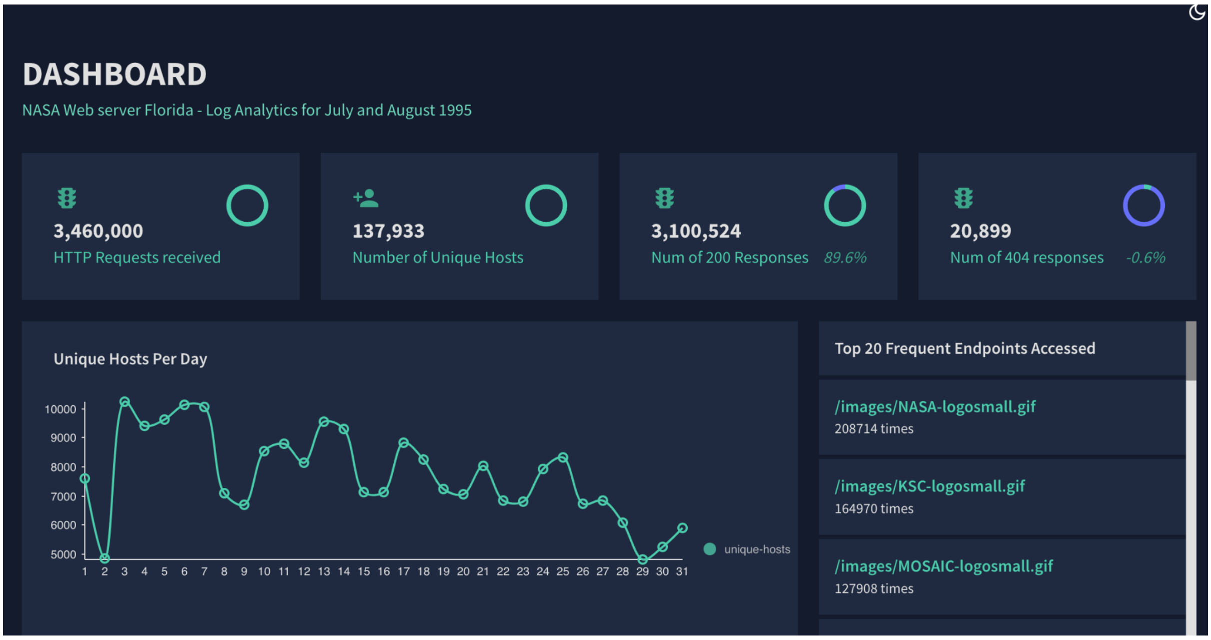
- Total Number of 200 (Okay) and 404 (Not Found) responses out of all the requests (log entries) which are around 3.46 million.
- A total number of unique hosts.
- Unique Hosts per day.
- Top 20 frequent endpoints accessed.
- HTTP Status code occurrences.
- Average daily requests per host for the whole month.
- Top ten error endpoints.
- Top twenty 404 response code endpoints and the hosts.
- Top five days where the 404 errors occurred the most.

All these analyses are presented through graphs using a front-end web app.

4.4 DATA VISUALISATION

After doing the analysis we exported the data into our frontend app (a react app) where we show all the data through graphs like Line charts, Pie charts and Bar charts along with some stats like how many requests were successful and how many returned errors.

Some of the results of the data analysis are shown in Figures 1, 2 and 3 which are screenshots from our react app.



DASHBOARD

NASA Web server Florida - Log Analytics for July and August 1995

3,460,000
HTTP Requests received

137,933
Number of Unique Hosts

3,100,524
Num of 200 Responses 89.6%

20,899
Num of 404 responses -0.6%

Unique Hosts Per Day

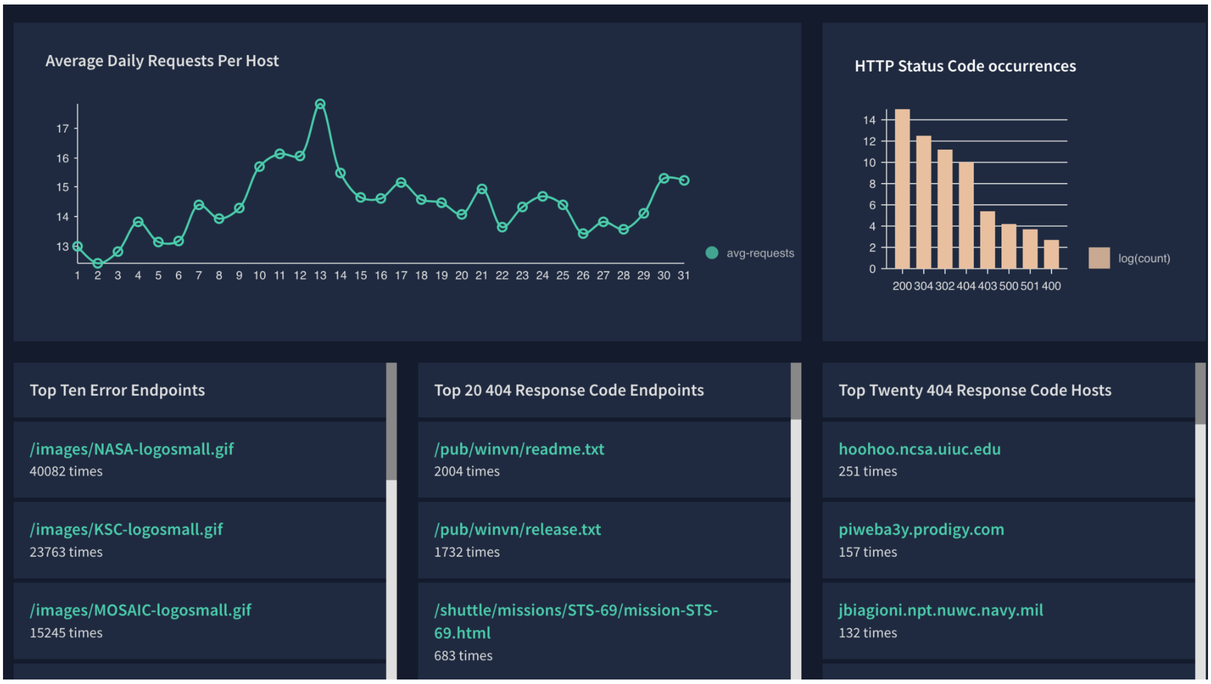
Top 20 Frequent Endpoints Accessed

/images/NASA-logosmall.gif
208714 times

/images/KSC-logosmall.gif
164970 times

/images/MOSAIC-logosmall.gif
127908 times

Figure 1



Average Daily Requests Per Host

HTTP Status Code occurrences

Top Ten Error Endpoints

/images/NASA-logosmall.gif
40082 times

/images/KSC-logosmall.gif
23763 times

/images/MOSAIC-logosmall.gif
15245 times

Top 20 404 Response Code Endpoints

/pub/winvn/readme.txt
2004 times

/pub/winvn/release.txt
1732 times

/shuttle/missions/STS-69/mission-STS-69.html
683 times

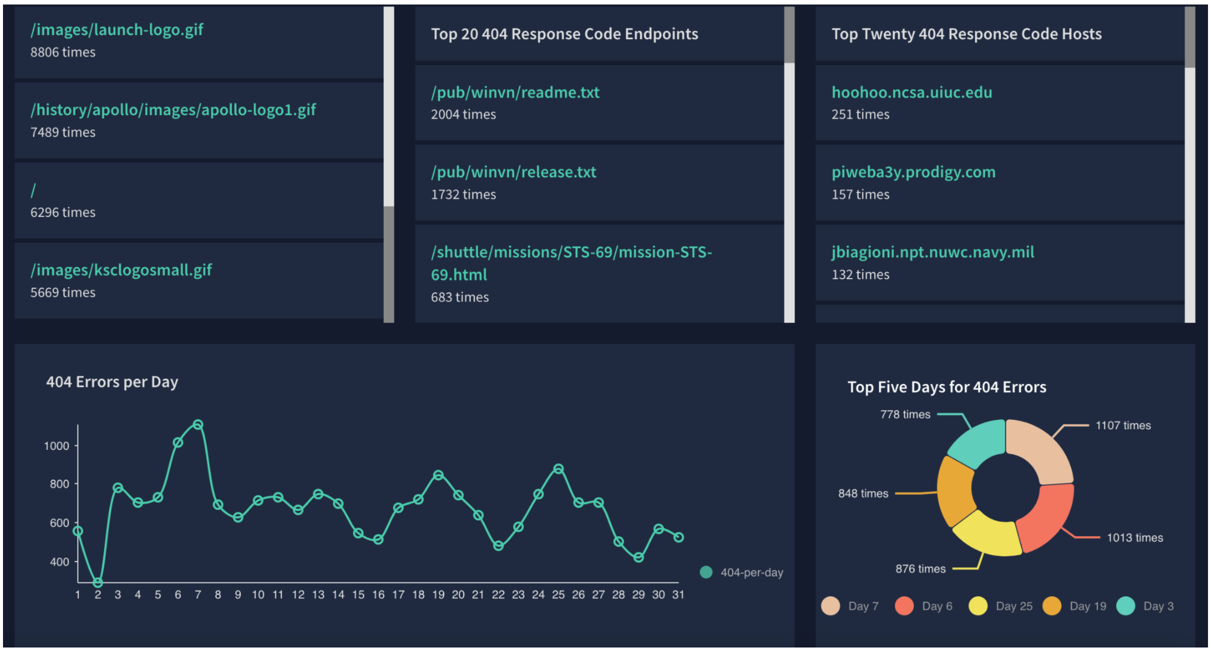
Top Twenty 404 Response Code Hosts

hoohoo.ncsa.uiuc.edu
251 times

piweba3y.prodigy.com
157 times

jbiagioni.npt.nuwc.navy.mil
132 times

Figure 2



/images/launch-logo.gif
8806 times

/history/apollo/images/apollo-logo1.gif
7489 times

/
6296 times

/images/ksclogosmall.gif
5669 times

Top 20 404 Response Code Endpoints

/pub/winvn/readme.txt
2004 times

/pub/winvn/release.txt
1732 times

/shuttle/missions/STS-69/mission-STS-69.html
683 times

Top Twenty 404 Response Code Hosts

hoohoo.ncsa.uiuc.edu
251 times

piweba3y.prodigy.com
157 times

jbiagioni.npt.nuwc.navy.mil
132 times

404 Errors per Day

Top Five Days for 404 Errors

Figure 3

5. HOW TO RUN

- The first step is to install docker and Nodejs on your machine.
- After this in the root directory of the project folder, run the command **docker-compose up,** this will run spark master, spark worker and Jupyter notebook as docker containers. The Jupyter notebook can be accessed at the address localhost:8888.
- To see the graphs and plots of our results change the Directory into the frontend folder and run the command **npm start,** and this will run the front-end on localhost:3000 where you can see our results of the data analysis.

REFERENCES

[1] https://data.world/shad/nasa-website-data