PROJECT REPORT
ON

Web Server Log Analysis

Carried Out at

CENTRE FOR DEVELOPMENT OF ADVANCED COMPUTING
ELECTRONIC CITY, BANGALORE

UNDER THE SUPERVISION OF
Mr. Mohit Ved

C-DAC Bangalore

Presented By

Prachi Ranjit Chavan	PRN:220950125067
Prashun Mishra	PRN:220950125068
Sandip Babanrao Gawande	PRN:220950125074
Sayali Yashvant Narale	PRN:220950125079
Satyam Kumar	PRN:220950125076

**PG DIPLOMA IN BIG DATA ANALYTICS
C-DAC, BANGALORE**

Candidate’s Declaration

We hereby certify that the work being presented in the report titled: “**Web Server Log Analysis”**, in partial fulfilment of the requirements for the award of PG Diploma Certificate and submitted to the department of PG-DBDA of the C-DAC ACTS Bangalore, is an authentic record of our work carried out during the period, 01/01/2023 to 10/03/2023 under the supervision of Mr. Mohit Ved, C-DAC Bangalore. The matter presented in the report has not been submitted by us for the award of any degree of this or any other Institute/University.

Name and Signature of Candidate:

Prachi Ranjit Chavan	PRN:220950125067
Prashun Mishra	PRN:220950125068
Sandip Babanrao Gawande	PRN:220950125074
Sayali Yashvant Narale	PRN:220950125079
Satyam Kumar	PRN:220950125076

Counter Signed by:

CERTIFICATE

TO WHOMSOEVER IT MAY CONCERN

This is to certify that

Prachi Ranjit Chavan

Prashun Mishra

Sandip Babanrao Gawande

Sayali Yashvant Narale

Satyam Kumar

Have successfully completed their project on

Web Server Log Analysis

Under the guidance of

Mr. Mohit Ved

Mr. Mohit Ved
(Project Guide)

Mrs. Uma Prasad
(Course Co-ordinator)

ACKNOWLEDGMENT

This project "**Web Server Log Analysis**" was a great learning experience for us and we are submitting this work to Advanced Computing Training School (CDAC ACTS).

We take this opportunity to express our gratitude to all those people who have directly and indirectly helped us during the competition of this project.

We pay thanks to Mr. Mohit Ved who has given guidance and a light to us during this project. His versatile knowledge about "**Web Server Log Analysis**" has eased us in the critical times during the span of this Final Project.

We acknowledge here support of our friends and seniors specially **Mr. Abhishek Chavan, Mr. Deepak Ror, Abhishek Bharti, Nalin Pushp, Sourav Shukla, Mr. Gaurav Khandpal, Shubham kore, Sagar Patki, Anuj Pathekar, and Swapnil Shukla** who contributed significantly to one or more steps.

We have tried our best to keep report simple yet technically correct. We hope we succeed in our attempt. We take full responsibility for any remaining sins of omission and commission.

From

The whole team.

ABSTRACT

Web has become the environment where people of all ages, languages and cultures conduct their daily digital lives. Working or entertaining, learning or socializing, home or on the road, individually or as a group, Web users are ubiquitously surrounded by an infrastructure of devices, networks and applications. With every click of website visitors log files are generated.

Web server log analysis is the process of examining and interpreting the log files generated by a web server. These log files contain a wealth of information about the traffic and activity on a website, including details about the users, their behavior, and the resources they accessed, warnings of failures.

This project involves machine learning techniques that can analyze web server log files from a web server for early detection of user behavior. The model takes live log data from a web server with help of spark streaming as input and predict the type of pattern followed by the user and based on that pattern model generates warning in case of anomaly.

We can use the python platform to perform operations that help to analyze the logs of a web server which will help the admin to restrict unwanted users.

Index

List of Figures

List of Tables

Chapter I

BASICS OF NETWORKING

OSI layers, a fundamental part of the open systems interconnection model (OSI), operate in conjunction to transfer the information received from a particular device to another in a network.

It enlists the standard protocols or rules necessary for information exchange between two systems over a particular network as a conceptual model. It uses seven abstract OSI layers to split the network communication, each handling a specific task independently to ensure connectivity between communication agents.

7 layers of OSI model:

1. **Application layer:**

   The topmost layer among all OSI model layers that connect the user applications to the network. However, applications don’t reside on this layer; the protocol does. HTTP, FTP, and SMTP are typical examples of application layer protocols. As layer 7, it also identifies the communication parties and checks resource availability in the network.

- **Hypertext Transfer Protocol (HTTP**) is an application-layer protocol for transmitting hypermedia documents, such as HTML.
- **Simple Mail Transfer Protocol (SMTP)** is a TCP/IP protocol used in sending and receiving email.
- **File Transfer Protocol (FTP)** is a network protocol for transmitting files between computers over Transmission Control Protocol/Internet Protocol (TCP/IP) connections.
- **POP3** is an older protocol that was originally designed to be used on only one computer.
- **Simple Network Management Protocol (SNMP**) is an application–layer protocol defined by the Internet Architecture Board (IAB) in RFC1157 for exchanging management information between network devices.

2. **Presentation layer:** This layer prepares data for layer 7 by checking its syntax and format against the standards. It also performs data translation, com-pression, and encryption to ensure communication parties understand the message accurately without encoding or formatting issues.

- **The Moving Picture Experts Group (MPEG)** is an alliance of working groups established jointly by ISO and IEC that sets standards for media coding, including compression coding of audio, video, graphics, and genomic data; and transmission and file formats for various applications.

- **The Secure Sockets Layer (SSL) protocol** was developed by Netscape Communications Corporation. SSL ensures the data that is transferred between a client and a server remains private.

- **Transport Layer Security (TLS)** encrypts data sent over the Internet to ensure that eavesdroppers and hackers are unable to see what you transmit which is particularly useful for private and sensitive information such as passwords, credit card numbers, and personal correspondence.

3. **Session layer:** The session layer establishes and maintains the connection between communication agents during data exchange and ultimately termi-nates it after successful trade to avoid resource exhaustion. As layer 5, it al-so leverages checkpoints to prevent data transfer from scratch after interruption.

- **Network Basic Input/Output System** (NetBIOS) is a network service that enables applications on different computers to communicate with each other across a local area network (LAN).

- **Session Announcement Protocol** (SAP) is used for multicast data session broadcasts and participant communication requirements.

4. **Transport layer:** This layer ensures reliable and accurate data exchange be-tween the sender and the receiver node. Layer 4 also provides connection-oriented or connectionless communication with flow control and error control. After receiving data from the session layer, it divides this information in-to smaller chunks, called segments, and then transfers them. The transport layer on the receiving side aggregates these data segments into a whole message for the session layer to interpret.

- **Transmission Control Protocol (TCP)** is a standard that defines how to establish and maintain a network conversation by which applications can exchange data.-All development projects developed in the same SAP System and transported on the same transport routes are grouped together to form a transport layer.

5. **Network layer:** Layer 3 handles inter-network communication by allowing de-vices from diverse networks to exchange data. It partitions the data segments received from layer 4 into packets on the sender's end and reassembles them again as packets on the receiver's end. Other notable functions of this layer include logical addressing, routing, and congestion control. Devices use IP addresses to identify themselves.**IPV5:** IPv5 never actually existed. Although there is an IP version with 5 assigned as its number, only two versions of the protocol are actually recognized by their numbers: IPv4 and IPv6. In fact, what we refer to as IPv5 is officially known as the Internet Stream Protocol (ST).**IPv6**: IPv6 is the most recent version of Internet Protocol (IP). It's designed to supply IP addressing and additional security to support the predicted growth of connected devices in IoT, manufacturing, and emerging areas like autonomous driving.**ICMP:** The

Internet Control Message Protocol (ICMP) is a network layer protocol used by network devices to diagnose network communication issues.**IPsec (Internet Protocol Security):** IPsec is used for protecting sensitive data, such as financial transactions, medical records and corporate communications, as it's transmitted across the network.**ARP:** Address Resolution Protocol (ARP) is a procedure for mapping a dynamic IP address to a permanent physical machine address in a local area network (LAN).**MPLS:** Multiprotocol Label Switching, or MPLS, is a networking technology that routes traffic using the shortest path based on “labels,” rather than network addresses, to handle forwarding over private wide area networks.

6. **Data link layer:** This performs similar functions as layer 3 but within a net-work. In short, it manages intra-network communication. It divides the upper-layer packets into frames, making them suitable for transfer via physical wires in layer 1. Devices use MAC addresses to identify themselves.

- **PPP:** Point - to - Point Protocol (PPP) is a communication protocol of the data link layer that is used to transmit multiprotocol data between two directly connected (point-to-point) computers.

- **Frame relay:** Frame relay is a protocol that defines how frames are routed through a fast-packet network based on the address field in the frame. Frame relay takes advantage of the reliability of data communications networks to minimize the error checking done by the network nodes.

- **ATM:** The asynchronous transfer mode (ATM) protocol architecture is designed to support the transfer of data with a range of guarantees for quality of service. The user data is divided into small, fixed-length packets, called cells, and transported over virtual connections.

7. **Physical layer:** Layer 1 ensures a seamless physical connection between network nodes by enabling them to transfer and receive unstructured raw da-ta in the form of 0s and 1s. It also performs line configuration and bit synchronization as the lowermost layer of the OSI model.

- **RS-232:** It is a standard communication protocol for connecting computers and their peripheral devices to enable serial data exchange. In simple terms, RS232 represents the voltage for the path used for data exchange between the devices

- **100BASE-TX**: It is the predominant form of Fast Ethernet, and runs over two wire-pairs inside a category 5 or above cable. Each network segment can have a maximum cabling distance of 100 meters (328 ft). One pair is used for each direction, providing full-duplex operation with 100 Mbit/s of throughput in each direction.

- **ISDN or Integrated Services Digital Network**: It is a circuit-switched telephone network system that transmits both data and voice over a digital line. You can also think of it as a set of communication standards to transmit data, voice, and signaling. These digital lines could be copper lines.

How does data flow through the OSI model?

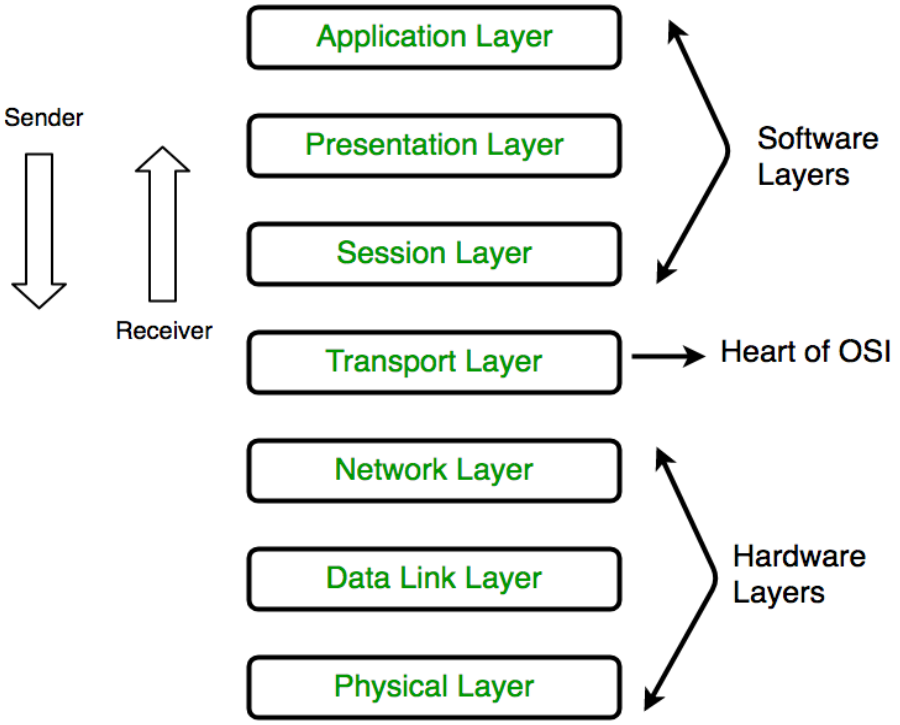


Figure 1: OSI MODEL

For information exchange over a network, data must travel from the application to the physical layer of OSI on the sender's device. While on the receiver's end, it should flow in the reverse direction in the OSI layer stack. In the sending process, the information first reaches the application layer where it’s assigned a relevant protocol, such as SMTP, then forwarded to the presentation layer. Upon receiving the message, the presentation layer encrypts and compresses it if required and forwards it to the session layer to open a communication session.

After this, the message reaches the transportation layer from which the process of data fragmentation begins. The data is broken into segments, then packets, and ultimately into frames at the transportation, network, and data link layer, respectively. Finally, the frames at the data link layer are forwarded to the physical layer for further transmission via physical cables in a bitstream of 1s and 0s.

Chapter II

Logs and Its Types

Log file is a textual data file that record events, processes, messages, and other data from applications, operating systems, network devices, web servers, and applications such as databases, email servers, and messaging systems etc. Log files can contain a wide range of information depending on the system or application being logged such as system events, error messages, security events, performance metrics, user activity, playing an important role in monitoring IT environments. Not only can you detect if things are working as they should be, but also if the system/network has been compromised. So, they can be used to track system activity, identify issues, and provide a record of what happened in case of problems or errors.

Table 1: Several types of Log files

1. **System Log Files**	These are logs created by the operating system and its components, such as the kernel and device drivers.
2. **Application Log Files**	These logs are created by applications and record events specific to the application.
3. **Security Log Files**	These logs record security-related events, including authentication attempts, access requests, and policy changes.
4. **Web Server Log Files**	These logs record events related to web server performance and user activity, such as requests for web pages, error messages, and usage statistics.
5. **Database Log Files**	These logs record database events, such as transactions and database modifications.
6. **Audit Log Files**	These logs record events related to compliance and auditing requirements.

WEB-SERVER LOG FILE

A web server log is a text document that contains a record of all activity related to a specific web server such as incoming requests (including information about the user, the request itself), and the server's response, over a defined period of time. The web server gathers data automatically and constantly to provide administrators with insight into how and when a server is used, as well as the users that correspond with that activity. The specific information logged depends on the server and its configuration. Some of the most common types of web server log files include:

Table 2: whats basically found in web server log files

1. **Access Log File**	It records every request made to the web server, providing detailed information about IP address of the requester, the date and time of the request, the requested URL, the HTTP status code of the response, and the size of the response. The access log file provides a rich source of data that can be used to monitor web server activity, analyze user behavior, and troubleshoot issues.
2. **Proxy Log File**	A proxy server acts as a gateway between user and the internet. It's an intermediary server separating end users from the websites they browse. The proxy log file typically contains information about the client, the request, and the server's response, similar to the information found in an access log file. However, because the request is forwarded through the proxy server (intermediary between clients and servers), the IP address of the client making the request is replaced by the IP address of the proxy server. Therefore, the proxy log file includes the IP address of the proxy server, rather than the IP address of the client.
	Error logs record details of any errors encountered by the web server, such as 404 errors, 500 errors, or any other HTTP errors.

File	Error logs is often used by developers, system administrators, and support teams to troubleshoot issues and provide valuable insights into server errors, bugs in applications, and other issues. Some common types of information found in these logs include error messages, stack traces, timestamps, user or session information, severity etc.
4. **Referrer Logs File**	A referral log is a type of log file that records information about the URLs or web pages that visitors have followed to reach a particular website. When a visitor clicks on a link to a website from another website, search engine, or social media platform, the URL of the referring page is recorded in the referral log. The referral log can then be used to track the traffic sources that are generating the most traffic, as well as to identify trends in visitor behavior and preferences.
5. **Agent Log File**	An agent log file is a type of log file that contains information about the activities of software agents, which are programs that automate tasks and interact with other programs, devices, or systems. Agent logs are commonly used in the fields of automation, robotics, and artificial intelligence to monitor and analyze the behavior of software agents. The fields included in an agent log file can vary depending on the specific application or system being used, but typically include agent id, timestamp, task or event, result or outcome etc.
6. **Security Log File**	A security log file is a type of log file that records information related to security events and activities on a computer system or network. The purpose of security log files is to provide an audit trail of security-related events, which can be used to investigate security incidents, monitor compliance with security policies and

	regulations, and identify potential security vulnerabilities. Some of the common security events logon and logoff events, failed logon attempts, account management events, system events.
7. **Application Log File**	An application log file is a file that contains a record of events that occur within an application, such as errors, warnings, and information messages. These log files are typically used for debugging and troubleshooting purposes. These details can help developers and system administrators diagnose issues and identify pattens that can lead to more efficient development and troubleshooting processes. Some of the typical details that may be included in an application log file are timestamp, severity level, message, source, Stack trace etc.

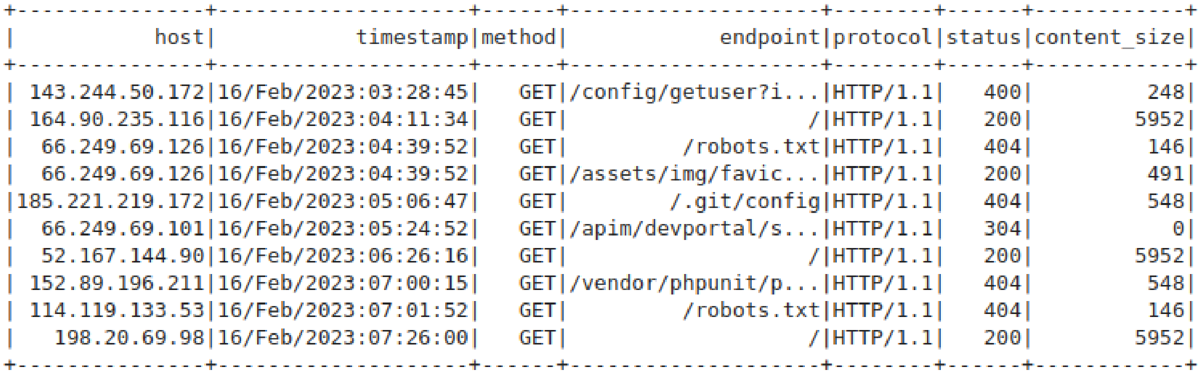
Chapter III

About Our Dataset

Nginx is a popular open-source web server that is widely used to serve web content, reverse proxy, load balance, and cache web applications. Nginx generates access logs, proxy logs and error logs. Nginx log files are typically stored in plain text format and can be analyzed using various log analysis tools and techniques. The access and error logs in Nginx will not only keep a tab on users activity but also save your time and effort in the process of debugging. Nginx log files are formatted using the "combined" log format by default, which includes the following fields: IP address, User, Time of request, Request line, HTTP status code, Size of response, Protocol.

Some Facts about our dataset:

- We Are fortunate to to get real log dataset from CDAC Bangalore’s Nginx server.
- The log consist of data from 12th February 2023 to 22nd February 2023.
- We have total three types of logs in dataset
    - Access log
    - Error log
    - Proxy log
- We did Extracted total 6218 rows and 7 columns from log files.



```
+---------------+--------------------+------+--------------------+--------+------+------------+
|           host|           timestamp|method|            endpoint|protocol|status|content_size|
+---------------+--------------------+------+--------------------+--------+------+------------+
| 143.244.50.172|16/Feb/2023:03:28:45|   GET|/config/getuser?i...|HTTP/1.1|   400|         248|
| 164.90.235.116|16/Feb/2023:04:11:34|   GET|                   /|HTTP/1.1|   200|        5952|
|  66.249.69.126|16/Feb/2023:04:39:52|   GET|         /robots.txt|HTTP/1.1|   404|         146|
|  66.249.69.126|16/Feb/2023:04:39:52|   GET|/assets/img/favic...|HTTP/1.1|   200|         491|
|185.221.219.172|16/Feb/2023:05:06:47|   GET|        /.git/config|HTTP/1.1|   404|         548|
|  66.249.69.101|16/Feb/2023:05:24:52|   GET|/apim/devportal/s...|HTTP/1.1|   304|           0|
|  52.167.144.90|16/Feb/2023:06:26:16|   GET|                   /|HTTP/1.1|   200|        5952|
| 152.89.196.211|16/Feb/2023:07:00:15|   GET|/vendor/phpunit/p...|HTTP/1.1|   404|         548|
| 114.119.133.53|16/Feb/2023:07:01:52|   GET|         /robots.txt|HTTP/1.1|   404|         146|
|   198.20.69.98|16/Feb/2023:07:26:00|   GET|                   /|HTTP/1.1|   200|        5952|
+---------------+--------------------+------+--------------------+--------+------+------------+
```

Figure 2: How our Dataset looks like

Analysis of Problem Statement

The Major Problem for this project is:

1. How to analyze the log files from the obtained data.
2. To use the available database to detect potential threat on live streaming data.
3. Attractive Visualization and Dashboard Creation.

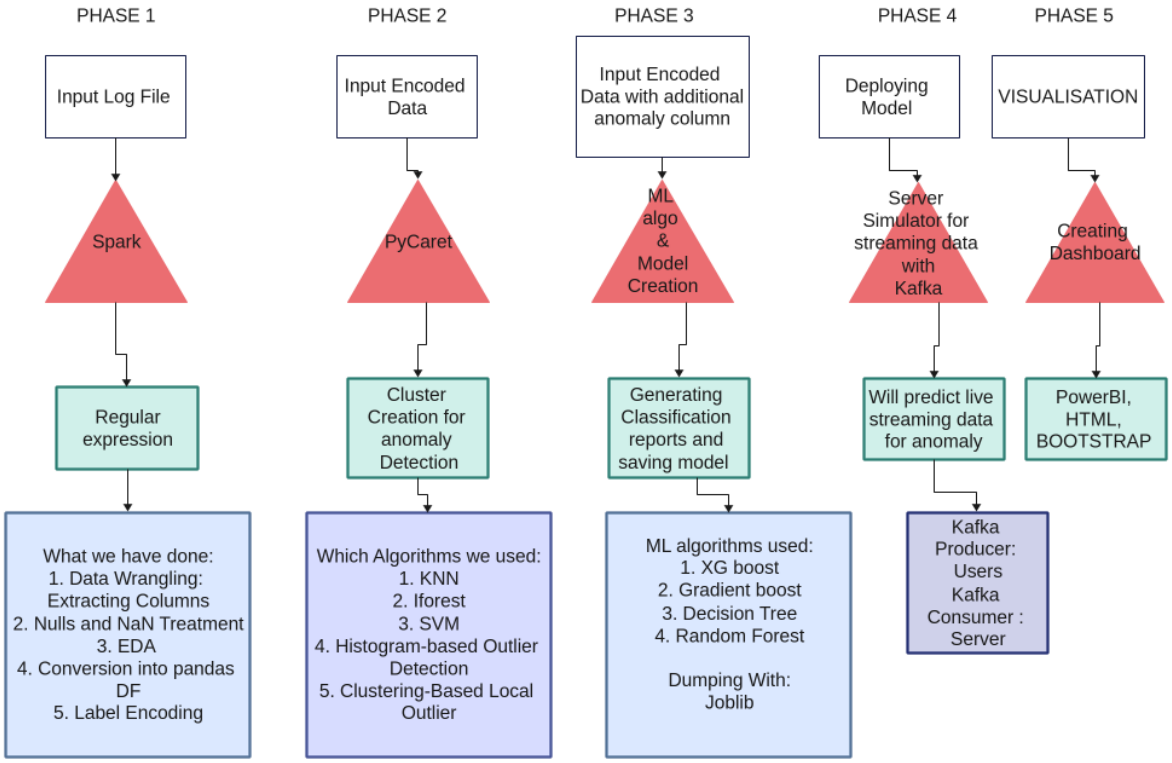

PHASE 1
PHASE 2
PHASE 3
PHASE 4
PHASE 5
Input Log File
Input Encoded Data
Input Encoded Data with additional anomaly column
Deploying Model
VISUALISATION
Spark
PyCaret
ML algo & Model Creation
Server Simulator for streaming data with Kafka
Creating Dashboard
Regular expression
Cluster Creation for anomaly Detection
Generating Classification reports and saving model
Will predict live streaming data for anomaly
PowerBI, HTML, BOOTSTRAP
What we have done:
1. Data Wrangling: Extracting Columns
2. Nulls and NaN Treatment
3. EDA
4. Conversion into pandas DF
5. Label Encoding
Which Algorithms we used:
1. KNN
2. Iforest
3. SVM
4. Histogram-based Outlier Detection
5. Clustering-Based Local Outlier
ML algorithms used:
1. XG boost
2. Gradient boost
3. Decision Tree
4. Random Forest
Dumping With: Joblib
Kafka Producer: Users
Kafka Consumer : Server

Figure 3: How we tried to solve this problem

PHASE 1 : Data Loading, Cleaning, and EDA

Steps followed

The two main libraries used are PySpark and Pandas

1. Create a new Virtual Environment.
2. Download and import the needed packages, libraries and Jupyter Lab.
3. Configuring spark variable.
4. Loading all dependencies like regular expression, pandas
5. Extract the log Dataset with glob.
6. Data Wrangling
    - Extracting Host Names
    - Extracting Time stamps
    - Extracting HTTP Request Method, URL’s and Protocols
    - Extracting HTTP Status Codes
    - Extracting HTTP Response Content Size
7. Putting All data together
8. Checking every Columns for nulls and unique values with pandas
9. Finding Missing values (spark)
10. Finding Null Counts (482 in status & content_size)
11. Passing missing information through our log data parsing pipeline.

12. Handling Time Stamp

13. HTTP Code Analysis and handling skewness

14. Analyzing Frequent Hosts

15. Analyzing Frequent Endpoints and Error Endpoints

16. Total Unique hosts, unique daily hosts

17. Avg no of daily request per hosts

18. Analyzing 404 codes, daily hourly

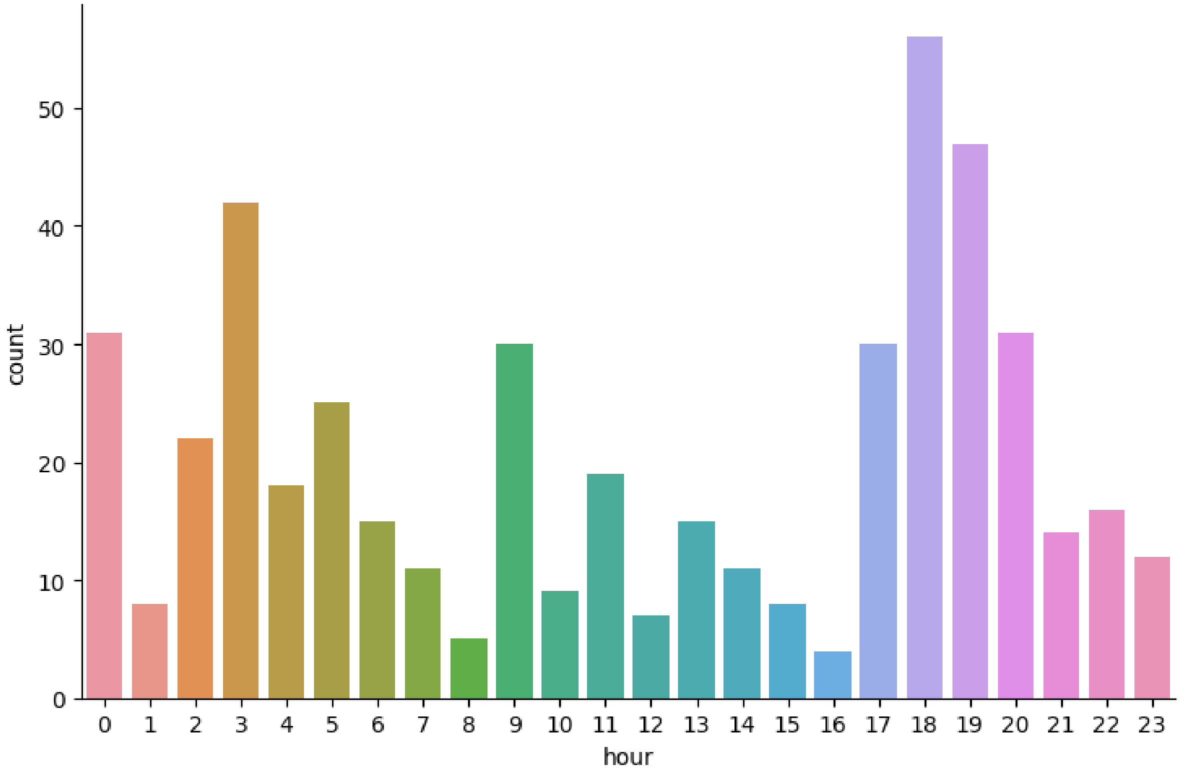


Figure 4: Visualizing Hourly 404 Errors

Chapter VI

PHASE 2 : Cluster Creation for Anomalies in Data

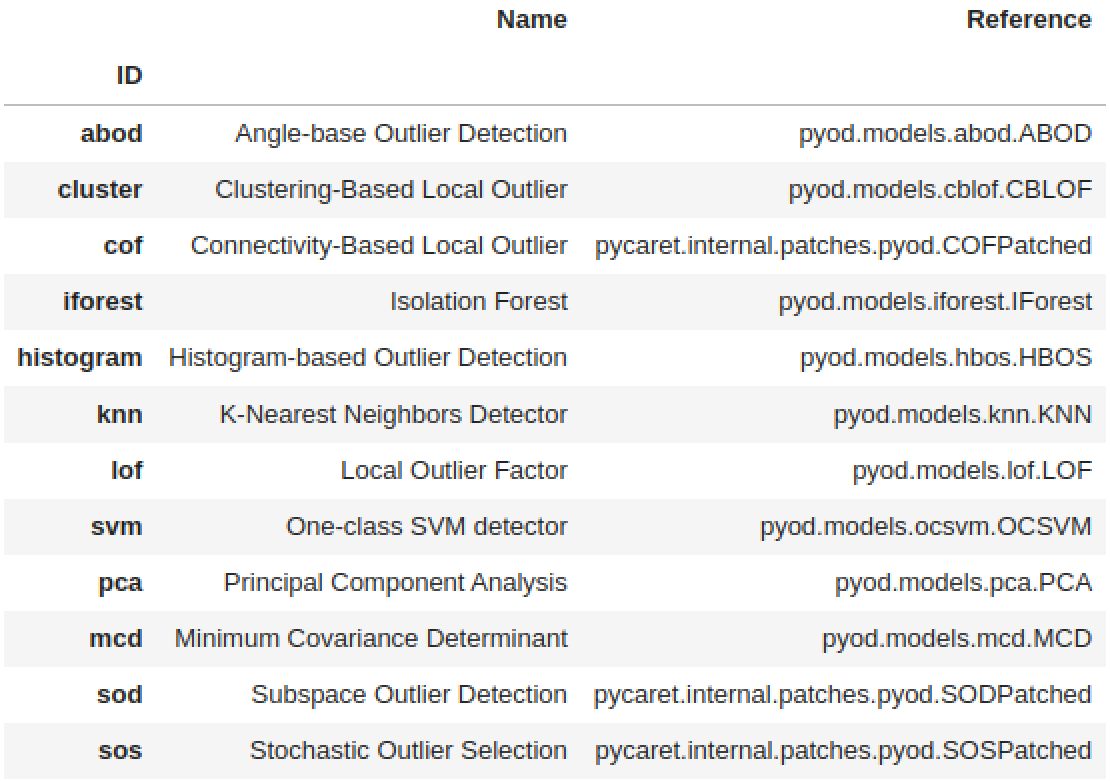
Anomaly detection

- It is a process of finding those rare items, data points, events, or observations that make suspicions by being different from the rest data points or observations. Anomaly detection is also known as outlier detection.

- To detect anomaly in dataset we have created PyCaret library.

What is Pycaret

- PyCaret is an open-source, low-code machine learning library in Python that automates machine learning workflows.

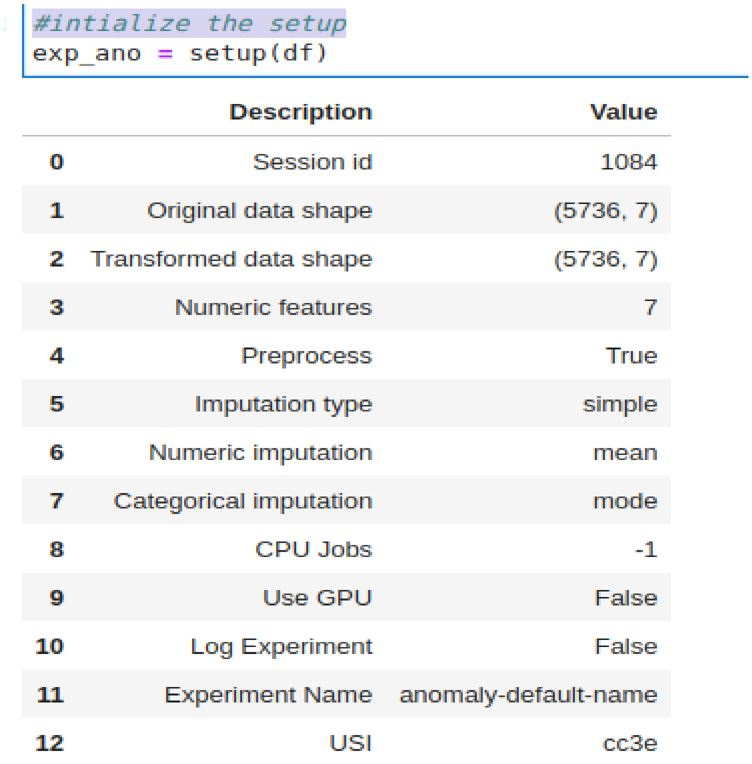


	Name	Reference
ID		
abod	Angle-base Outlier Detection	pyod.models.abod.ABOD
cluster	Clustering-Based Local Outlier	pyod.models.cblof.CBLOF
cof	Connectivity-Based Local Outlier	pycaret.internal.patches.pyod.COFPatched
iforest	Isolation Forest	pyod.models.iforest.IForest
histogram	Histogram-based Outlier Detection	pyod.models.hbos.HBOS
knn	K-Nearest Neighbors Detector	pyod.models.knn.KNN
lof	Local Outlier Factor	pyod.models.lof.LOF
svm	One-class SVM detector	pyod.models.ocsvm.OCSVM
pca	Principal Component Analysis	pyod.models.pca.PCA
mcd	Minimum Covariance Determinant	pyod.models.mcd.MCD
sod	Subspace Outlier Detection	pycaret.internal.patches.pyod.SODPatched
sos	Stochastic Outlier Selection	pycaret.internal.patches.pyod.SOSPatched

Figure 5: ML models in pycaret

Steps followed

1. Importing required Dependency
2. Loading Labeled dataset
3. Initialize pycaret setup



```
#intialize the setup
exp_ano = setup(df)
```

	Description	Value
0	Session id	1084
1	Original data shape	(5736, 7)
2	Transformed data shape	(5736, 7)
3	Numeric features	7
4	Preprocess	True
5	Imputation type	simple
6	Numeric imputation	mean
7	Categorical imputation	mode
8	CPU Jobs	-1
9	Use GPU	False
10	Log Experiment	False
11	Experiment Name	anomaly-default-name
12	USI	cc3e

Figure 6: Initialize pycaret setup

4. Creating Models; we will further continuing with KNN

Table 3: Output of cluster creation with different Algorithms

Algorithm Name	No of 0’s	No of 1’s
K-Nearest Neighbors Detector	5511	225
Isolation Forest	5449	287
One-class SVM detector	5449	287

4. Visualizing cluster

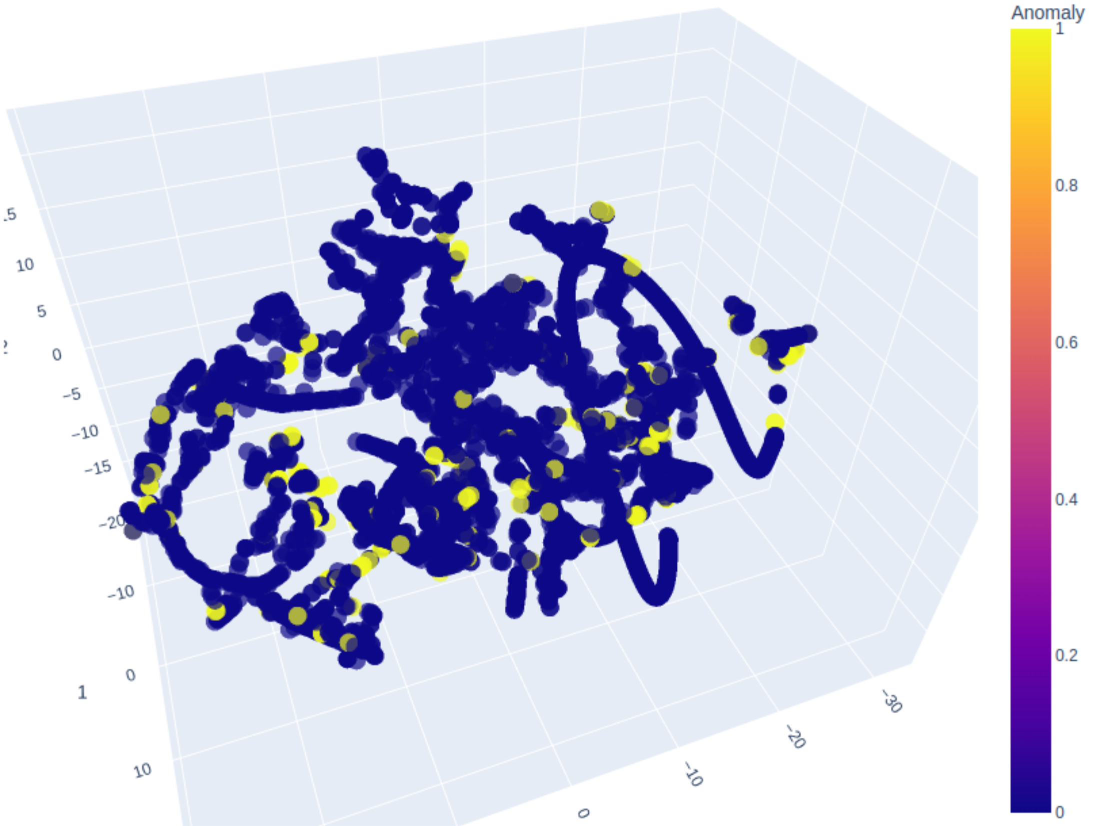
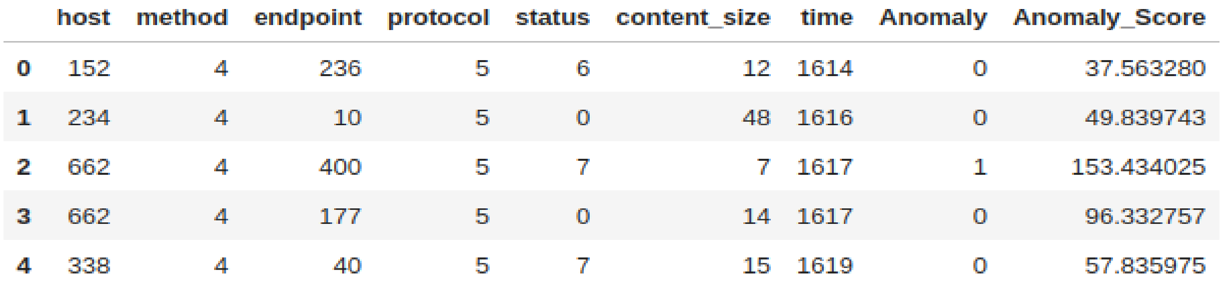


Figure 7: Visualisation of Cluster

5. Saving Pandas DataFrame with one additional columns Anomaly, and Anomaly_Score.



	host	method	endpoint	protocol	status	content_size	time	Anomaly	Anomaly_Score
0	152	4	236	5	6	12	1614	0	37.563280
1	234	4	10	5	0	48	1616	0	49.839743
2	662	4	400	5	7	7	1617	1	153.434025
3	662	4	177	5	0	14	1617	0	96.332757
4	338	4	40	5	7	15	1619	0	57.835975

Figure 8: DataFrame after Clustering

Chapter VII

PHASE 3 : Using ML algorithm and Model Creation

Steps followed

1. Importing required Dependency
2. Splitting target columns
3. Splitting train-test data (80:20)
4. Training different ML models and evaluating themselves

Table 4: Classification reports of ML models

Algorithms		precision	recall	F1 score	Support
Gradient Boost	0	0.96	1.00	0.98	1091
	1	0.92	0.21	0.34	57
	Accuracy	0.96			
XG Boost	0	0.98	0.99	0.99	1091
	1	0.83	0.53	0.65	57
	Accuracy	0.97			
Decision Tree	0	0.98	0.99	0.98	1091
	1	0.72	0.58	0.64	57
	Accuracy	0.97			
Random Forest	0	0.97	1	0.98	1.90
	1	0.89	0.42	0.57	57
	Accuracy	0.97			

5. Balancing Dataset with imblearn

Table 5: Classification report of ML model after tuning dataset

Algorithms		precision	recall	F1 score	Support
Gradient Boost	0	0.96	0.96	0.96	83
	1	0.96	0.96	0.97	92
	Accuracy	0.97			
XG Boost	0	0.98	0.93	0.96	58
	1	0.93	0.98	0.96	57
	Accuracy	0.96			
Decision Tree	0	0.95	0.93	0.94	58
	1	0.93	0.95	0.94	57
	Accuracy	0.94			
Random Forest	0	0.98	0.93	0.96	58
	1	0.93	0.98	0.96	57
	Accuracy	0.96			

6. Dumping model (gradient boost) with joblib because it have less ratio of FP and FN comparing to other models.

Chapter VIII

PHASE 4 : Deployment of Model over Kafka & MongoDB Connection

Steps Followed(for consumer→ users):

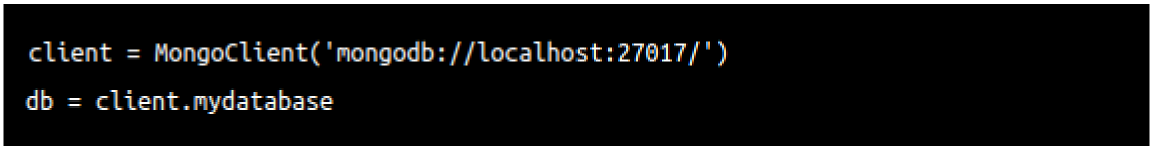
1. Importing required Dependency
2. Creation of Kafka topic
3. Creation of Admin Object
4. Creating Topic on Kafka Server
5. Reading labeled log dataset(using pandas)
6. sending contents as dictionary.

Steps Followed(for producer→ server):

1. Importing required Dependency
2. Loading model with joblib.
3. Assigning topic to consumer
4. Implementing ML model to predict incoming data that transferred from users.

Steps followed (for MongoDB Connection):

1. Importing required Dependency (pymongo→ MongoClient)
2. Connectiviy to MongoDB server

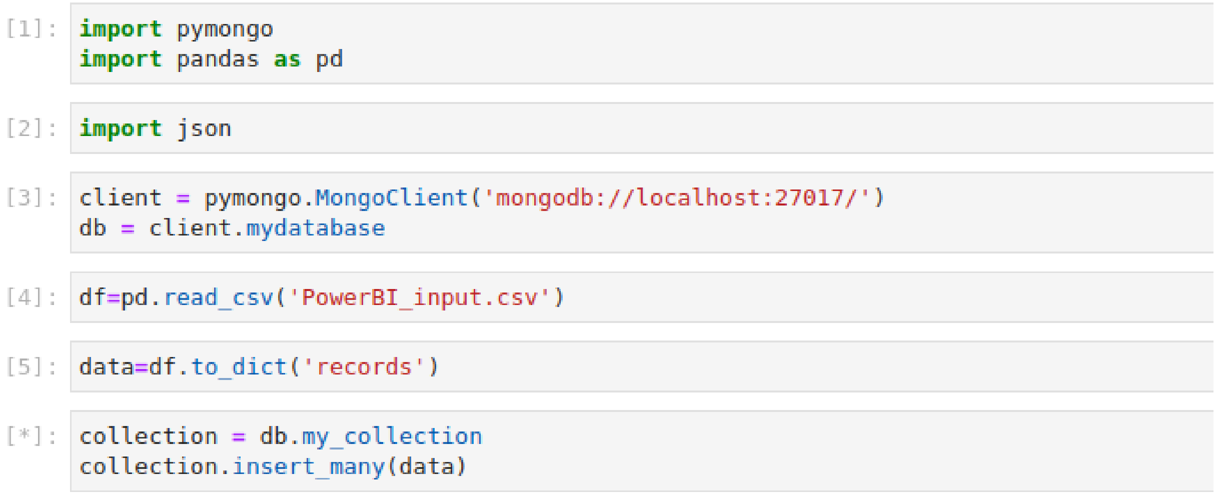


```
client = MongoClient('mongodb://localhost:27017/')
db = client.mydatabase
```

Figure 9: Connectivity to MongoDB server

3. Inserting datafram to MongoDB collections

4. Checking for successful transactions



```
[1]: import pymongo
     import pandas as pd

[2]: import json

[3]: client = pymongo.MongoClient('mongodb://localhost:27017/')
     db = client.mydatabase

[4]: df=pd.read_csv('PowerBI_input.csv')

[5]: data=df.to_dict('records')

[*]: collection = db.my_collection
     collection.insert_many(data)
```

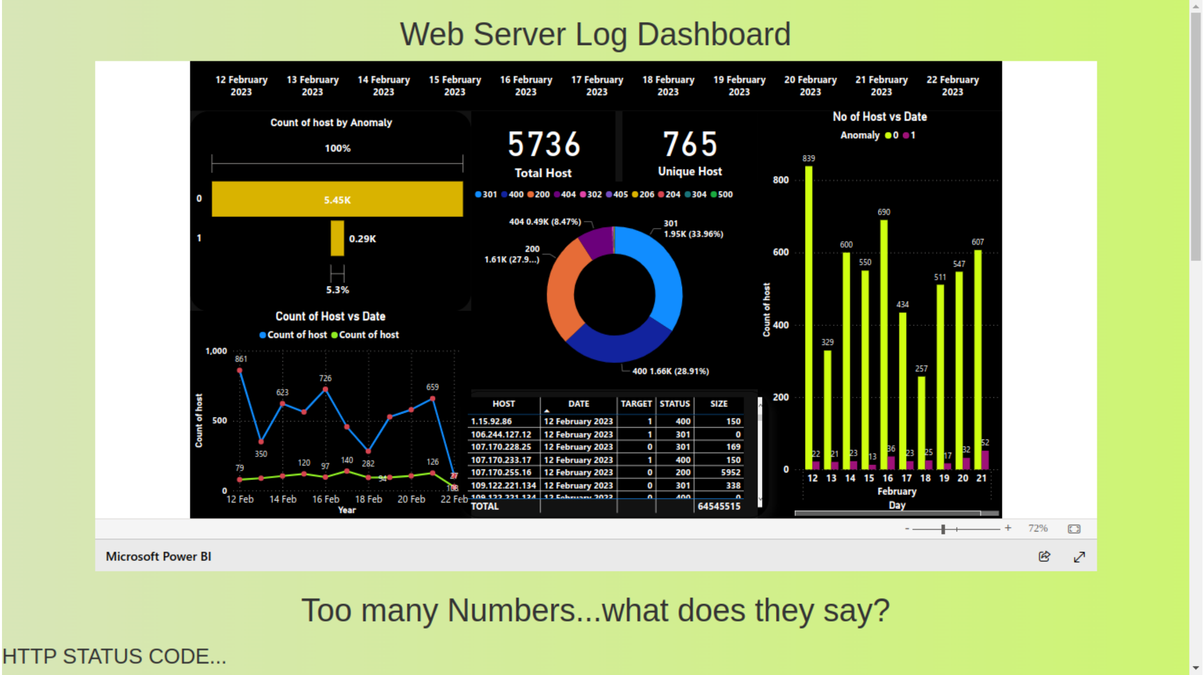
Figure 10: Mongo Client connectivity

Chapter IX

PHASE 5 : Visualization

Steps Followed:

1. Loading Dataset in power BI(Labeled +unlabeled Data)
2. Generating Report in Power BI using
    - Bar charts
    - Donut Charts
    - Line
    - Tables
    - Slicer
    - and cards
3. Creating workspace and uploading files in it
4. Publishing Report to web
5. Embedding code to HTML
6. Creating html page with iframe tag
7. Creating paragraphs , adding additional information
8. using bootstrap



Web Server Log Dashboard

HOST	DATE	TARGET	STATUS	SIZE
1.15.92.86	12 February 2023	1	400	150
106.244.127.12	12 February 2023	1	301	0
107.170.228.25	12 February 2023	0	301	169
107.170.233.17	12 February 2023	1	400	150
107.170.255.16	12 February 2023	0	200	5952
109.122.221.134	12 February 2023	0	301	338
109.122.221.134	12 February 2023	0	400	0
TOTAL				64545515

Too many Numbers...what does they say?

HTTP STATUS CODE...

Figure 11: Visualisation with HTML and Power BI

Chapter X

Conclusion

➔ Some of the key findings that can be drawn from web server log analysis include:

✓ The most popular pages on the website
✓ Most errors found on which page
✓ Unique users
✓ Traffic Hours
✓ Average request per user
✓ Average error per user
✓ The most common errors encountered by users
✓ The most frequent sources of referral traffic
✓ The patterns and trends in website traffic over time

➔ By analyzing web server logs, it is also possible to identify potential security threats, such as hacking attempts or unauthorized access to the website. This information can be used to strengthen website security and protect sensitive data.

➔ Overall, a web server log analysis project can provide valuable insights into website performance, user behavior, and security threats, which can be used to improve the website and enhance the user experience.

Chapter XI

Application

- Web Server Log Analysis has a wide range of applications in the trending fields of Data Security, Data Analytics, Digital Marketing as well as many other field.

- It can be use for Page advertisement, SEO, web user experience enhancement, Data Base admin to monitor for potential threats.

- Model perform well for the dataset that was used in the project and to enhance experience large reliable dataset is required Spark and Kafka both can be used as used in the project to detect potential threat in transactions of data from web server.

- Web-server log analysis can be used to get a sense of the overall user experience. This type of processing is advantageous to any company that relies largely on its website for revenue generation or client communication.

- In future this project can be scaled up and be used in enterprises which rely on user interaction for revenue generation.

REFERENCES

- https://towardsdatascience.com/
- https://www.youtube.com/@BizEcommerce
- https://www.w3schools.com/
- https://www.geeksforgeeks.org/
- https://stackoverflow.com/
- https://www.youtube.com/@DarshilParmar
- Teddy Mantoro, Normaziah binti Abdul Aziz, Faculty Science and Technology, Jakarta, Indonesia. "Log Visualization of Intrusion and Prevention Reverse Proxy Server Against Web Attacks", 2013 International Conference on Informatics and Creative Multimedia.
- Valeur, F., Vigna, G., Kruegel, C. & Kirda, E. (2006), "An anomaly driven reverse proxy for web applications". In Proc. of the 2006 ACM symposium on Applied computing, pages 361-368.
- K. R. Suneetha, Dr. R. Krishnamoorthi, Bangalore Institute of Tech, Vishveshwaraya Technology University, Anna University, "Identifying User Behavior by Analyzing Web Server Access Log File", IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.4, April 2009
- Jaideep Srivastava, Robert Cooley, Mukund Deshpande, Pang Ning Tan "Web usage mining: Discovery and Applications of usage patterns from web data" SIGKDD Explorations- vol-1, issue-2 Jan 2000 pages 12-33.
- Haibin Liu, Vlado Keselj , Faculty of Computer Science, Dalhousie University, 6050 University Ave, Halifax, NS, Canada, "Combined mining of Web server logs and web contents for classifying user navigation patterns and predicting users' future requests", Data & Knowledge Engineering 61 (2007) 304–330, www.elsevier.com/locate/datak.

- Justin Myers, Michael R. Grimaila , Michael R. Grimaila , Center for Cyberspace Research Air Force Institute of Technology Wright-Patterson, “ Towards Insider Threat Detection using Web Server Logs”, CSIIRW '09, April 13-15, Oak Ridge, Tennessee, USA.

- Resul Das *, Ibrahim Turkoglu b aDepartment of Informatics, Firat University, 23119 Elazig, Turkey, “ Creating meaningful data from web logs for improving the impressiveness of a website by using path analysis method” , Expert Systems with Applications 36 (2009) 6635–664, journal homepage: www.elsevier.com/locate/ewsa.

- Hideki Koike, Kazuhiro Ohno, Graduate School of Information Systems University of Electro-Communications, Japan, “SnortView: Visualization System of Snort Logs”,

- https://www.kaggle.com/code/eliasdabbas/webserver-log-file-analysis

- https://www.kaggle.com/code/nebaroland/neba-roland-ngwa-web-usage-mining