International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 09 Issue: 05 | May 2022 www.irjet.net p-ISSN: 2395-0072

SQL Injection and HTTP Flood DDOS Attack Detection and Classification Based on Log Data

Kapil Patel[1], Prof. Rajni Ranjan Singh Makwana[2]

[1]B. Tech. Student, Dept. of Computer Science and Engineering, Madhav Institute of Technology and Science, Madhya Pradesh, India
[2]Assistant Professor, Dept. of Computer Science and Engineering, Madhav Institute of Technology and Science, Madhya Pradesh, India

---

Abstract - *Due to continuous growth in Tech. Industry and also due to COVID - 19 there is a huge increase in web servers and web-applications, almost every office work and educational applications are online, due to this there is huge increase in cyber attacks. So it is important to detect them as early as possible to stop it before it causes much damage to a web-application and data loss. Since web application generates huge number of logs data it is boring and difficult task to do analysis of log data manually by person. But log data is important to monitor as it contains computer generated records, which contain data about every activity and operations, so analyzing these can help in early detection of some types of attacks like SQL injection, DDOS attack, brute force attack, and cross-site scripting (XSS), etc. To improve old method of manually inspection of log analysis in this paper, an anomaly detection and classification model has been proposed, which can also be used for early attack detection by analyzing log data. To build the model machine learning decision tree algorithm has been used to classify the data into three categories like normal logs, SQL injected logs and DDOS attack logs. After comparing logs data with trained model it successfully classifies the logs into different categories and also detects attacks.*

Key Words: **Log data, Machine Learning, Decision Tree, SQL Injection, HTTP Flood DDOS Attack**

1. INTRODUCTION

Now a days due to improvement in cyber world and easily available tools web applications faces many suspicious activities and attacks because of script kiddies, they generally performs scanning and attacks a website using an automated vulnerability scanner tools or trying to fuzz script (code) into a parameter for SQL injection, cross-site scripting (XSS) etc. and often performs DDOS attacks to down the server working etc. In many such cases, logs on the web-server have to be monitored and analyzed to figure out what is going on. If it is a serious case and suspicious matter then requires a cyber expert for forensic investigation. Since running server generates huge amount and different types of logs data it is very difficult to monitor manually, even though it is not efficient enough to filter different logs data into different categories and also demand many hours to inspect the data.

In this paper a SQL injection and HTTP flood DDOS attack log anomaly detection and classification model have been proposed based on machine learning to classify the logs data into categories based on anomalous data present in logs and which further can be used to detect attacks. For building classification model a simple rule-based decision tree classifier has been used which is enough to meet demands and successfully classify the logs data. Decision tree algorithm comes under supervised machine learning algorithm, in which labeled training data of both cases normal and unusual situations is used for training model. To build the model logs data files is selected in the first step and then the parsing of logs components is done using parsing techniques, then labeling and encoding of components according to presence of some patterns or data presence in components is done and after that the labeled components of log is passed to the decision tree classifier to predict the type of log based on anomalous data present in log into three categories: Normal logs data, SQL injected logs data, HTTP Flood DDOS Attack logs data.

Experiments with 45897 logs data from real web-application hosted on local XAMPP server, building and testing model shows overall accuracy of 98.88% of classification and detection.

2. BACKGROUND

In this part of the paper brief introduction of HTTP flood DDOS attack , followed by SQL injection, and then followed by the introduction to the web logs is presented.

2.1 HTTP Flood DDOS Attack

DDOS-simply stands for Distributed Denial Of Service. It could be of any kind like hijacking a server, port overloading, denying internet based services etc. HTTP flood DDOS attack is an application layer volumetric attack, mainly focus on crashing the web servers and online web applications. These attacks are comparatively sophisticated, here a huge number of legitimate looking HTTP GET, or POST requests are used to flood the server in

this type of attack. This in return causes a denial of services. Figure 1 depicts working view of attack.

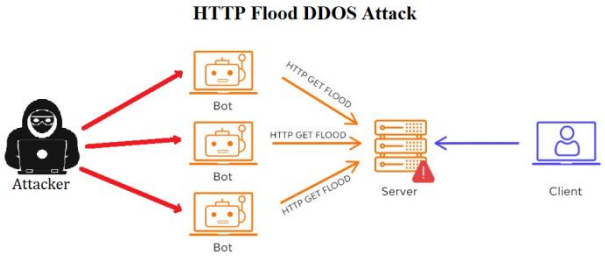

Fig -1: HTTP Flood DDOS Attack

2.1 SQL Injection Attack

SQL is a high-level scripting language which is used to store, access and maintain database systems. SQL injection is most common type of SQL attacks in which malicious SQL code is used to read, modify, and delete database information that are not allowed to access for normal users. Even it can also execute administrative operations. Because server contains huge amount of users' data which makes servers valuable target for attackers. By using SQL injection attack hackers also can bypass authentication system, compromised data and can do information disclosure. Even SQL injections can also be pivoted into remote command execution. Figure 2 depicts a view how SQL query is injected and attack is performed.

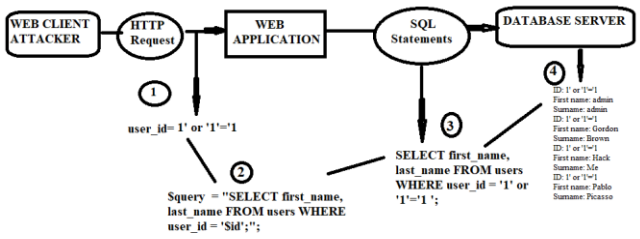

Fig -2: SQL Query Injected Attack

2.3 Web server logs

Logs are generally computer automatic generated records, which record every progress and problems during running of applications and systems. A web server log files usually contains the records of user access, requests and errors record of each time whenever user requests from web server. Mainly there are four types of log files in XAMPP Apache Server file that are: access log file, error log file, php error log file, and ssl_request log file. In this paper only access log file have taken to predict the logs type based on their content present. Apache access log file records all events that are requested, and response sent by server. Generally Apache server follows Common Log Format specification by default, so each HTTP request is written in separate line and composed of several tokens which are separated by spaces, blank values of tokens are represented by a hyphen (-). For demonstration a single log record and its specifications are given in figure 3.

192.168.169.17 - - [26/Mar/2022:21:53:29 +0530] "GET /dvwa/vulnerabilities/fi/?page=include.php HTTP/1.1" 200 4183 "http://192.168.169.107/dvwa/instructions.php" "Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0"		
Host	**192.168.169.17**	The IP address of the client.
Identity	-	The identity information reported by the client.
User	-	The user name of a successful HTTP authentication.
Date	**[26/Mar/2022:21:53:29 +0530]**	The date and time of the request.
Request	**"GET /dvwa/vulnerabilities/fi/?page=include.php HTTP/1.1"**	The request line from the client is given in double quotes.
Status	**200**	The three-digit HTTP status code generated in response to the clients request.
Bytes	**4183**	The number of bytes in the object returned to the client.
Request Header Referer	**"http://192.168.169.107/dvwa/instructions.php"**	The HTTP request header referer contains an absolute or partial address of the page that makes the request.
Request Header User Agent	**"Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0"**	The user agent identifies the application, operating system, vendor and/or version of the requesting user agent.

Fig -3: Example of a Log record

3. MODEL FRAMEWORK OVERVIEW

Figure 4 demonstrates the complete model framework overview. To detect anomaly based on web access logs data the paper mainly involves four phases: log collection, log parsing, features extraction and labeling and then training decision tree classifier and prediction of attacks.

Log collection: During run time of web-application it generates logs on every request. These valuable records could be utilized for various purposes but here used for anomaly detection, and so logs data are collected in first step for further usage. For building model data is collected by self hosting web application on local XAMPP server.

***Log parsing*:** logs are continuously written records and generally unstructured data which have to be parsed to extract useful information out of them and to make them structured. For splitting logs and extracting features out of them regular expressions are used so that information can be easily stored and manipulated.

Feature extraction and labeling: After parsing logs into separate parts, we need to further search some patterns in logs parts and based on that patterns, label them and then encode them into numerical feature arrays, whereby machine learning models can be applied.

Training decision tree classifier and prediction: Now, the feature arrays of data can be used to feed to machine

learning decision tree classifier models for training and to generate a model that can be used for prediction and classification. The trained model now can be further used to classify a new log data into a normal log, SQL injected log, DDOS attack log based on anomaly present in data.

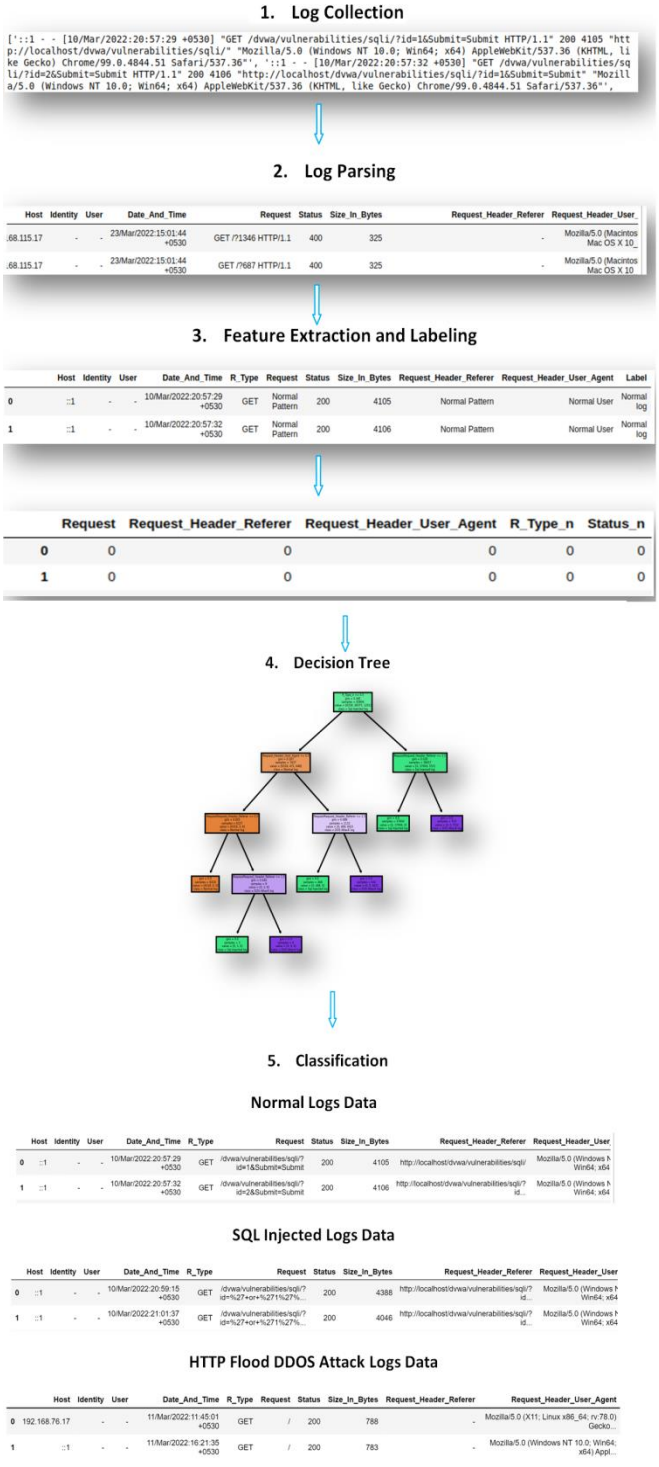

Fig -4: Anomalous Log Detection and Classification System Framework Overview

4. MODEL DESIGN AND IMPLEMENTATION

In this part of the paper detailed explanation of model framework is presented. In this section, it tells how to extract useful features and find patterns in them and label data based on that pattern. After labeling them the decision tree classifier is used to predict type of log data and to classify the log data present in file into different categories.

4.1 Data preprocessing

Features extraction: After parsing the logs file and making them structured it is important to select only important features out of them which are not much variable can provide valuable information about the log data. The Features that are taken are: HTTP status code, request type, request, request header referrer and request header user-agent which give enough information to detect presence of anomaly. HTTP request type are generally divide into two categories GET and POST. GET is used to get something from server without changing it and carries request without hiding parameter details in URL, whereas POST is used to make changes in data based on request and is more secure. HTTP status code tells whether a request has completed or not, status code has special meaning, it is generated based on responsive status of a web server. Some of the common found HTTP status code and information are listed in figure 5.

HTTP Status code	Information
200	OK
206	Created
301	Moved Permanently
302	Found
304	Not Modified
400	Bad Request
403	Forbidden
404	Not Found
414	URI Too Large

Fig -5: Some common HTTP status code

It is often found that in SQL injection attack logs data, malicious SQL query is injected into some parameters to perform the attack, so to filter out such query this model uses request, request header referer, request header user-agent of log data to filter out such injected query. To distinguish features between normal data and SQL injected data few examples of such logs are taken in figure 6 and some features are also underlined to filter out.

Fig -6: SQL Query Injected logs.

Similarly in HTTP flood DDOS attack log data also has some patterns that are common in request data, status code, request header referer, request header user-agent of log data. Few examples of such logs are taken in figure 7 and some features are also underlined to filter out.

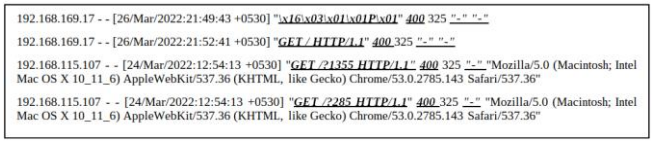

Fig -7: HTTP Flood DOS Attack logs.

Data labeling: Based on patterns found in logs data different type of labels are assigned to log data tokens. Labels taken for Labeling for each unit of data are listed in table 1.

Table -1: Labels used for each unit of data

Log Tokens	Label based on Patterns		
Request Type	GET	POST	
Request	Normal Pattern	SQL Injected Pattern	DOS Attack Pattern
Request Header Referer	Normal Pattern	SQL Injected Pattern	Empty
Request Header User Agent	Normal User	Special User	No User
Log Classification Label	Normal log	SQL Injected log	DOS Attack log

Based on collection of logs data during attack and normal situation and by analyzing each unit of data, labels have been assigned manually into three categories for classification are: normal log, SQL injected log, DOS attack log. After labeling based on these patterns, these labels have to be encoded into numerical features to train the model. To do so, I have encoded normal data as '0', SQL injected data as '1'and DOS attack data as '2'. Now after encoding these data are passed to train machine learning models.

4.2 Decision Tree Classifier

After data parsing, processing and preparing training dataset to build a model, now a suitable machine learning algorithm has to be selected which best fits for the model, to do so a simple rule-based decision tree classifier is enough to predict and classify the log data.

Decision Tree algorithm is a classification algorithm, it adds the data point to a particular labeled group on the basis of some conditions. Decision Tree graphically represents flowchart-like structure which demonstrates all the possible solutions that can be used to take a decision. Decisions are generally taken based on some conditions and which can be easily explained. Splitting of decision tree in this model is done by based on Gini impurity, which is used to split nodes when categorical data has to be predicted.

4.3 Prediction and Classification

After training decision tree classifier of model, now model is tested with new dataset to make predictions, which successfully classifies the logs dataset into the normal data set, SQL injected dataset and DOS attack dataset. After getting these datasets, inspection of each piece of logs data is done manually to make sure that there is not any false prediction, classification and labeling. After checking all datasets, this model is further used for live attack detection of SQL injection attack and HTTP flood DOS attack and found that the model is predicting perfectly.

5. EXPERIMENT

5.1 Dataset

The data has been collected by self hosting web applications on local XAMPP server and by performing HTTP flood DOS attack using pentmenu tool and slowloris script and SQL Injection by sqlmap tool and burp suite, etc. This dataset consists of three types of logs dataset during normal browse, SQL injection attack and HTTP DOS attack which is visualized in figure 8, 9, 10.

For training dataset 45897 logs data are taken from web access log file, which has data during normal running of web server and also during attack, which is enough for creating training and testing dataset for the model.

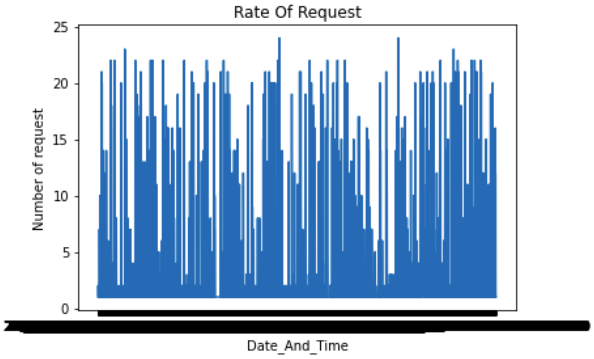

Fig -8: Logs during normal browse.

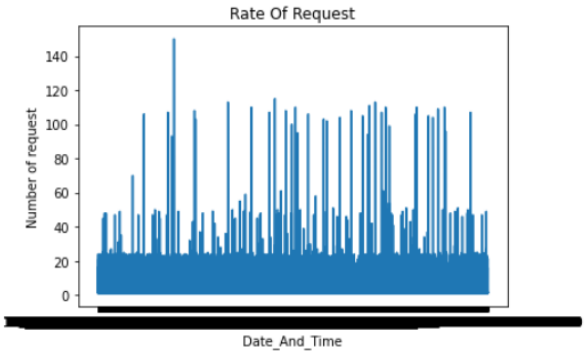

Fig -9: Logs during SQL injection attack.

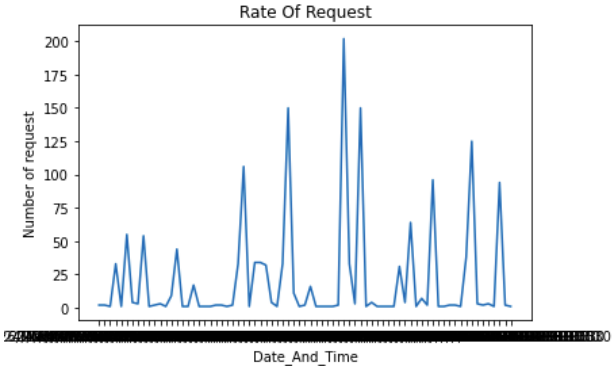

Fig -10: Logs during HTTP flood DOS attack.

5.2 Confusion Metrics:

Confusion metrics provides a holistic view of the classification model. Confusion matrix is one of the classification matrices, used to evaluate performance of classification algorithms. To examine the anomaly detection and classification system, this paper first examines the model by using confusion matrix, which describe the performance of detection and classification model in detail. Actual values: column in confusion matrix that are real values. Predicted values: rows in confusion matrix are the predictions. Based on actual and predicted logs confusion metrics for model is shown in table 2.

5.3 Results

The result calculated based on confusion matrix is presented in table 3. Accuracy tells the percentage of total number of predictions and classifications of log data that were correctly predicted and classified. Precision tells the proportion of really positive prediction and classification of logs data. Recall tells the measure of identifying true positives of predictions and classifications of logs data. F1-Score gives harmonic mean of the precision of the classifier model and recall of the classifier model. Based on these analysis it is found that the model successfully predicts and classifies the log data with (TP=45383) out of (45897) with overall accuracy of 98.88 %.

Table -2: Confusion Matrix

		Actual Label		
Predicted Label		Normal logs	SQL Injected logs	DOS Attack logs
	Normal logs	6301(TP)	18	0
	SQL Injected logs	470	36162(TP)	0
	DOS Attack logs	20	6	2920(TP)

Table -3: Result of Prediction and Classification

	Accuracy	Precision	Recall	F1-score
Normal logs	98.89%	1.0	0.93	0.96
SQL Injected logs	98.92%	0.99	1.0	0.99
DOS Attack logs	99.94%	0.99	1.0	1.0

6. RELATED WORK

Anomaly detection refers to find unusual patterns in data that do not follow regular patterns or give unexpected behavior. In past, there had been a lot of anomaly detection models were proposed. In 2004, where a Decision Tree model was applied to find error detection for web request log system in [1].In 2010 based on web log data K. R. Suneetha, R. Krishnamoorth build a classification model to identify interested users using decision trees in [2]. Similarly in 2017 based on web access log data a machine learning model is applied by Qimin Cao and Yinrong Qiao to detect anomalies in web log file in [3]. Influenced by these papers, this paper

adopted a more enhanced filtering and classifying supervised machine learning model with simple text analysis and pattern matching approach that can be used to detect and classify both normal logs data and anomaly logs data present into categories.

7. CONCLUSIONS

This paper describes a detection and classification model based on supervised machine learning for web server access log file. The model is able to analyze, detect and classify the log data based on anomaly present in access log data. The data for building model is collected data from real web-applications which has been hosted on local XAMPP server and performed different SQL injection attacks and DOS attacks on that web server. In this paper first extraction of valuable features have been done, based on some patterns and presence of some features and then labeling and encoding of that features have been presented, based on prepared dataset a decision tree classifier has been built and trained. Based on results it is found that this model successfully classifies web access log file data into three categories normal log file, SQL injected log file, and DOS log file. After testing this model, it was found that it achieved overall 98.88 % accuracy of detection and classification.

REFERENCES

[1] M. Y. Chen, A. X. Zheng, J. Lloyd, M. I. Jordan, and E. A. Brewer, "Failure diagnosis using decision trees," in 1st International Conference on Autonomic Computing), 2004, pp.36–43.

[2] K. R. Suneetha, R. Krishnamoorthi, "Classification of Web Log Data to Identify Interested Users Using Decision Trees", 2010.

[3] Qimin Cao and Yinrong Qiao, "Machine Learning to Detect Anomalies in Web Log Analysis," in 2017 3rd IEEE International Conference on Computer and Communications.