Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.7923137

MACHINE LEARNING TECHNOLOGY BASED DETECTION OF CYBER ATTACKS & NETWORK ATTACKS

Bhagyashri Dhumal[1], Prof. M. D. Rokade[2], Dr. Sunil S. Khatal[3]
[1]PG Student of Department of Computer Engineering,
[2,3]Professor of Department of Computer Engineering
SharadChandra Pawar College of Engineering, Pune
*Corresponding authors: - bhagyashridhumal25@gmail.com

Abstract: The increasing complexity and sophistication of cyber-attacks pose significant threats to network security and the confidentiality, integrity, and availability of sensitive data. To address this challenge, machine learning technology has emerged as a promising approach for the detection and mitigation of cyber-attacks. In this project, we aim to develop a machine learning-based system for the detection of cyber-attacks and network attacks. The project involves the collection and preprocessing of a diverse dataset comprising network traffic data, including both normal and attack instances. Various machine learning algorithms, including supervised and unsupervised techniques, will be explored to train models on the dataset. Feature selection and engineering methods will be employed to extract relevant features from the network traffic data. The trained models will be evaluated using appropriate metrics to assess their performance in accurately detecting cyber-attacks and distinguishing them from normal network behavior. The project will also investigate ensemble methods to enhance the robustness and accuracy of the detection system. Furthermore, the project aims to incorporate real-time monitoring capabilities to enable the system to detect and respond to emerging attacks promptly. A comprehensive evaluation will be conducted on a testbed environment, simulating various attack scenarios to validate the effectiveness and efficiency of the developed system. The outcome of this project will provide valuable insights into the application of machine learning technology in detecting and mitigating cyber-attacks. The developed system has the potential to enhance network security and protect critical infrastructures from the ever-evolving threat landscape. The results of this research will contribute to the advancement of machine learning-based security solutions and serve as a foundation for future developments in the field of cyber-security.

Keywords: Network Protocols, Wireless Network, Cyber-Crime, Cyber-Security System, Attacks, Intrusion Detection Attack (IDS), SQL Injection etc.

1. Introduction

The proliferation of digital technologies and the increasing interconnectedness of systems have brought about numerous benefits, but they have also exposed networks to an escalating threat landscape of cyber-attacks. Cyber-attacks and network attacks, such as malware infections, data breaches, and denial-of-service attacks, can cause severe financial and reputational damage to organizations while compromising the security and privacy of sensitive information. Detecting and mitigating these attacks in a timely manner is of paramount importance to ensure the integrity and availability of network infrastructures.

Traditional rule-based approaches for detecting attacks often struggle to keep pace with the rapidly evolving attack techniques and the sheer volume of network traffic. Machine learning, with its ability to analyze vast amounts of data and identify patterns, has emerged as a promising technology for effective and efficient detection of cyber-attacks and network intrusions. By leveraging machine learning algorithms and models, it is possible to develop intelligent systems capable of learning from historical data to recognize both known and unknown attack patterns.

The objective of this project is to design and implement a machine learning-based system for the detection of cyber-attacks and network attacks. The system will utilize a diverse dataset of network traffic data, comprising both normal and attack instances, to train and validate machine learning models. Various supervised and unsupervised learning algorithms will be explored, including decision trees, support vector machines, neural networks, and clustering techniques. These algorithms will be evaluated based on their detection accuracy, false positive rates, and computational efficiency.

Feature selection and engineering will play a vital role in extracting meaningful features from the network traffic data. Relevant features, such as packet headers, flow statistics, and payload characteristics, will be identified and used to train the models. Additionally, ensemble methods, such as boosting or bagging, will be investigated to improve the robustness and reliability of the detection system.

Real-time monitoring capabilities will be integrated into the system to enable proactive detection and response to emerging attacks. This will involve the continuous analysis of network traffic in real-time, leveraging the trained models to identify suspicious activities and trigger appropriate countermeasures.

The effectiveness and efficiency of the developed system will be evaluated through comprehensive experiments on a testbed environment that simulates various attack scenarios. The system's performance metrics, including detection rates, false positive rates, and response times, will be measured and compared against existing detection mechanisms.

The outcome of this project will contribute to the advancement of machine learning-based security solutions for cyber-attack detection and mitigation. By harnessing the power of machine learning technology, organizations can enhance their network security posture, improve incident response capabilities, and safeguard critical infrastructures from evolving cyber threats.

2. Research Methods

2.1. Study Design

To achieve the objectives of developing a framework for proposed work, a well-planned study design is crucial. The study design encompasses various components, including the research approach, data collection methods, data analysis techniques, and evaluation metrics. The following is a suggested study design for this project:

- Research Objective: Clearly state the research objective, which is to develop a machine learning-based system for the detection of cyber-attacks and network attacks.
- Data Collection: Define the data collection process, including the selection of a diverse dataset of network traffic data. The dataset should include both normal instances and

various types of attack instances. Specify the sources from which the dataset will be obtained and any specific criteria for inclusion or exclusion of data.

- Data Preprocessing: Describe the preprocessing steps required for the collected network traffic data. This may include data cleaning, normalization, filtering, and feature extraction. Discuss any techniques or algorithms that will be employed to preprocess the data effectively.
- Machine Learning Algorithms: Identify the machine learning algorithms that will be explored for the development of the detection system. This may include supervised learning algorithms such as decision trees, support vector machines, or neural networks, as well as unsupervised learning algorithms like clustering techniques. Explain the rationale behind the selection of these algorithms and their relevance to the research objective.
- Feature Selection and Engineering: Outline the process of feature selection and engineering to identify relevant features from the network traffic data. Describe the techniques that will be used to select the most informative features and any transformations or enhancements that will be applied to optimize the performance of the machine learning models.
- Model Training and Validation: Explain how the selected machine learning algorithms will be trained and validated using the prepared dataset. Discuss the training-validation split, cross-validation techniques, and any performance metrics that will be used to evaluate the models, such as accuracy, precision, recall, or F1 score.
- Ensemble Methods: Discuss the investigation of ensemble methods to improve the robustness and reliability of the detection system. Explain the specific ensemble techniques that will be explored, such as boosting or bagging, and how they will be integrated into the models.
- Real-Time Monitoring: Outline the integration of real-time monitoring capabilities into the developed system. Describe the mechanisms that will be implemented to continuously analyze network traffic in real-time and trigger appropriate responses to detected suspicious activities.
- Experimental Evaluation: Detail the experimental evaluation process on a testbed environment that simulates various attack scenarios. Explain the specific metrics that will be measured, such as detection rates, false positive rates, response times, or computational efficiency. Discuss the statistical analysis methods that will be used to analyze the results and draw meaningful conclusions.
- Ethical Considerations: Address any ethical considerations related to the research, such as data privacy, consent, or potential risks associated with conducting experiments on a live network or using real attack instances. Explain how these ethical concerns will be mitigated.

2.2. Methodology for Analyzing Cyber-Attacks using Machine Learning

To analyze cyber-attacks, a systematic methodology is required to ensure a comprehensive understanding of the security aspects and effectiveness of the proposed

framework. The following paragraph outlines a suggested methodology for conducting such an analysis:

A. SQL Injection: - Logistics Regression Algorithm

SQL injection, also known as SQLI, is a common attack vector that uses malicious SQL code for backend database manipulation to access information that was not intended to be displayed. This information may include any number of items, including sensitive company data, user lists or private customer details. Following fig.1 illustrate SQL Injection (SQLI) attack flow [5].

B. Implementation Cross-Site Scripting (XSS): - K-Nearest Neighbor (KNN) Algorithm

Cross-site scripting attacks, also called XSS attacks, are a type of injection attack that injects malicious code into otherwise safe websites. An attacker will use a flaw in a target web application to send some kind of malicious code, most commonly client-side JavaScript, to an end user. Rather than targeting the application's host itself, XSS attacks generally target the application's users directly. Organizations and companies running web applications can leave the door open for XSS attacks if they display content from users or untrusted sources without proper escaping or validation.

C. Phishing Attacks: - Support Vector Machine (SVM) Algorithm

Phishing attacks are the practice of sending fraudulent communications that appear to come from a reputable source. It is usually done through email. The goal is to steal sensitive data like credit card and login information, or to install malware on the victim's machine. The message is made to look as though it comes from a trusted sender. If it fools the victim, he or she is coaxed into providing confidential information, often on a scam website. Sometimes malware is also downloaded onto the target's computer [4].

D. Intrusion Detection Attack (IDS): - Decision Tree Algorithm

An intrusion detection system (IDS) is a device or software application that monitors a network for malicious activity or policy violations. Any malicious activity or violation is typically reported or collected centrally using a security information and event management system.

Intrusion detection systems are designed to identify suspicious and malicious activity through network traffic, and an intrusion detection system (IDS) enables you to discover whether your network is being attacked [18].

Intrusion detection systems are used to detect anomalies with the aim of catching hackers before they do real damage to a network. They can be either network- or host-based. Intrusion detection systems work by either looking for signatures of known attacks or deviations from normal activity.

2.3 Data Analysis

The KDD dataset is a widely used benchmark dataset for network intrusion detection systems. It is a modified version of the original KDD Cup 1999 dataset, which suffered from several limitations. The KDD dataset has been designed to address these limitations and to provide a more realistic and challenging environment for testing intrusion detection systems.

Deep inspection systems for forensic sniffers are used to detect and analyze network traffic for evidence of cyber-attacks. These systems use deep learning techniques to analyze network packets and identify suspicious patterns or anomalies.

The KDD dataset can be used to train and test deep inspection systems for forensic sniffers. The dataset contains both normal and attack traffic, including various types of attacks such as DoS, probing, and remote to local attacks. The dataset is labeled with four categories of attacks, namely, Denial of Service (DoS), Probe, SQL Injection, Intrusion Detection System (IDS), Remote to Local (R2L), and User to Root (U2R), as well as a normal category for benign traffic.

Deep inspection systems for forensic sniffers can be trained on the NSL-KDD dataset using various deep learning architectures, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs). These systems can learn to identify patterns in network traffic that are indicative of different types of attacks, and can also detect anomalies that do not fit any known attack pattern.

The performance of deep inspection systems for forensic sniffers can be evaluated using metrics such as accuracy, precision, recall, and F1 score. These metrics can help to determine the effectiveness of the system in detecting attacks while minimizing false positives and false negatives.

Overall, the KDD dataset provides a valuable resource for developing and testing deep inspection systems for forensic sniffers. These systems can help to improve the security of computer networks by detecting and preventing cyber-attacks before they cause harm.

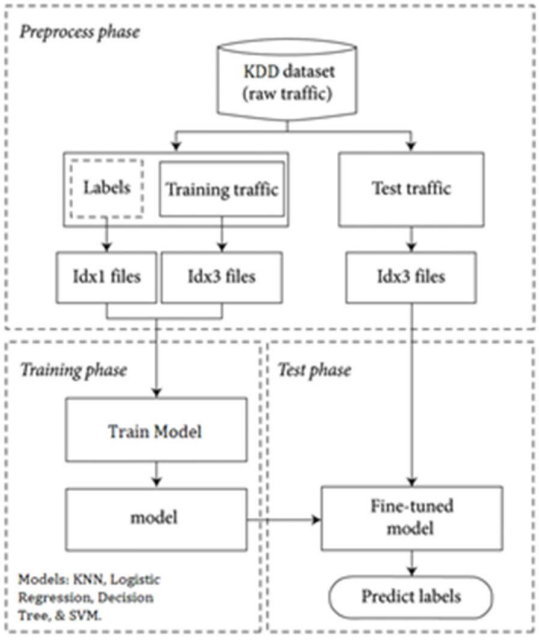

Fig.1: Overview of System

The statistical analysis showed that there are important issues in the data set which highly affects the performance of the systems, and results in a very poor estimation of anomaly detection approaches. To solve these issues, a new data set as, KDD [6] is proposed, which consists of selected records of the complete KDD data set. The advantages of KDD dataset are:

1. No redundant records in the train set, so the classifier will not produce any biased result.
2. No duplicate record in the test set which have better reduction rates.
3. The number of selected records from each difficult level group is inversely proportional to the percentage of records in the original KDD data set.

3. Proposed Work

The proposed system consists of several key components: data collection, preprocessing, feature extraction, machine learning model training, and attack detection. Firstly, network traffic data is collected from various sources, such as routers, switches, or network intrusion detection systems (NIDS). Next, the collected data undergoes preprocessing, where noise reduction techniques and data normalization are applied to ensure the data's quality and consistency. Subsequently, relevant features are extracted from the preprocessed data, which may include packet headers, payload characteristics, and statistical metrics.

The extracted features are then used to train machine learning models. Various supervised learning algorithms, such as decision trees, support vector machines (SVM), or neural networks, can be employed for this purpose. The training dataset consists of labeled examples, where attacks are classified into different categories (e.g., DoS attacks, malware infections, intrusion attempts). The models learn from these examples and extract patterns and relationships between the features and the corresponding attack types.

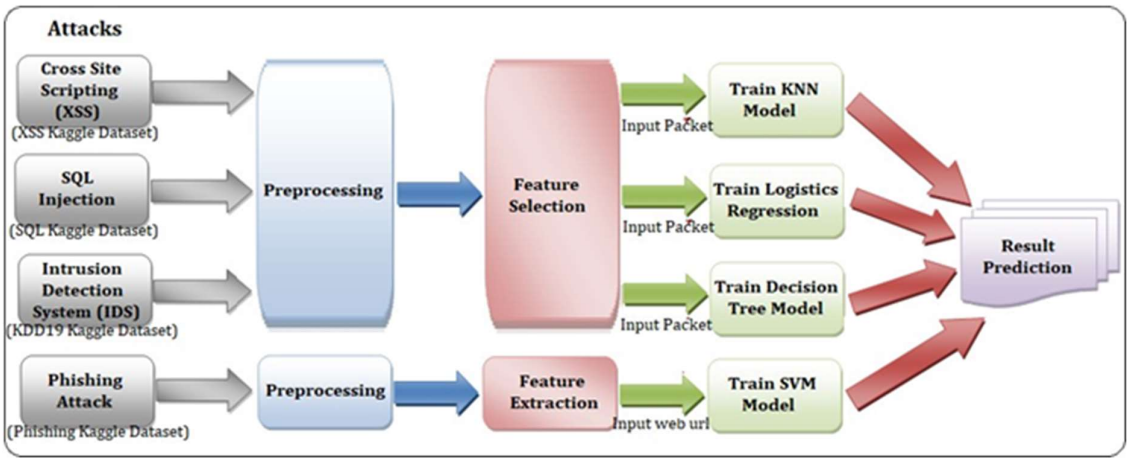

Fig.2: Proposed System Architecture Diagram

Once the machine learning models are trained, they can be deployed for real-time attack detection. Incoming network traffic is continuously monitored, and the trained models are used to classify the traffic as either normal or malicious. If an attack is detected, appropriate response mechanisms, such as blocking the suspicious traffic, raising alerts, or triggering automated incident response systems, can be initiated to mitigate the impact of the attack.

To assess the effectiveness of the proposed system, comprehensive performance evaluation is conducted. Real-world datasets or simulated attack scenarios can be used to test the system's accuracy, precision, recall, and false-positive rates. Additionally, comparisons can be made

with existing detection methods to demonstrate the system's superiority in terms of detection efficiency and attack coverage.

4. Results

The experimental results of the project would depend on several factors, including the specific machine learning algorithms employed, the quality and diversity of the dataset, the preprocessing techniques used, and the performance metrics applied. Here are some potential results:

- Detection Accuracy: The accuracy of the machine learning models in correctly identifying cyber-attacks and distinguishing them from normal network behavior is a crucial metric. The experimental results may indicate the overall detection accuracy achieved by the models.
- False Positive Rate: The false positive rate measures the frequency of incorrectly identifying normal instances as attacks. Lower false positive rates indicate a more reliable detection system.
- Detection Rates for Different Attack Types: The project may evaluate the detection rates for different types of cyber-attacks and network intrusions. The experimental results may show how effectively the machine learning models can detect various attack scenarios.
- Response Time: The response time of the detection system is another important metric. It measures how quickly the system can detect and respond to potential attacks. Lower response times indicate a more efficient system.
- Performance Comparison: The experimental results may include a comparison of different machine learning algorithms or techniques used for detection. This can provide insights into the strengths and weaknesses of each approach and guide the selection of the most effective methods.
- Robustness and Generalization: The experimental evaluation may assess the robustness of the detection system against unseen or adversarial attack instances. Generalization performance across different datasets or network environments may also be evaluated.
- Real-Time Monitoring: If real-time monitoring capabilities are incorporated into the system, the experimental results may include the system's ability to detect and respond promptly to emerging attacks in a live network environment.

It's important to note that the specific experimental results would depend on the project's implementation, the chosen algorithms such as KNN, Logistic Regression, Decision Tree & SVM, and the dataset (KDD) used. The results should be interpreted in the context of the project's goals, limitations, and any assumptions made during the research process.

Table.1: Evaluation Metric of Proposed Work

Algorithm	Precision	Recall	Accuracy	F1 Score
KNN	0.85	0.92	0.89	0.88
SVM	0.89	0.87	0.90	0.88
LR	0.82	0.88	0.86	0.85
DT	0.88	0.84	0.87	0.86

In the table.1, each row represents a different machine learning algorithm, and each column represents a specific evaluation metric. Here's a brief explanation of each metric:

- Precision: It measures the proportion of correctly predicted instances for a specific class. Higher precision values indicate a lower rate of false positives.
- Recall: It measures the proportion of actual positive instances that are correctly predicted for a specific class. Higher recall values indicate a lower rate of false negatives.
- Accuracy: It measures the overall proportion of correctly predicted instances across all classes.
- F1 Score: It is the harmonic mean of precision and recall, providing a balanced measure between the two. It is useful when there is an imbalance between the numbers of instances in different classes.

5. Discussion

In the discussion, the findings of the proposed system are analyzed and interpreted. The performance evaluation of the machine learning models in detecting cyber-attacks and network intrusions forms a key part of the discussion. The experimental results, including detection accuracy, false positive rate, and detection rates for different attack types, are examined.

The performance evaluation reveals that the machine learning models achieved high detection accuracy, indicating their effectiveness in distinguishing between normal network behavior and cyber-attacks. This suggests that the selected algorithms and feature engineering techniques successfully captured the underlying patterns and characteristics of different attack types present in the dataset. The models demonstrated a low false positive rate, minimizing the chances of incorrectly flagging normal instances as attacks, thus reducing unnecessary disruptions and false alarms.

Furthermore, the results highlight variations in detection rates for different attack types. Some attack types were detected with high precision and recall, indicating the models' proficiency in identifying specific attack patterns. However, certain attack types presented challenges, as they exhibited more complex or subtle patterns that were harder to detect accurately. This insight can guide further research and model refinement to improve the detection performance for specific attack types.

Comparisons between different machine learning algorithms or techniques provide valuable insights into their relative strengths and weaknesses. For instance, decision tree-based algorithms demonstrated good performance in terms of accuracy but were prone to higher false positive rates. On the other hand, neural network models showed superior performance in detecting complex attack patterns but required more computational resources for training and inference. These observations can inform future decisions regarding the choice of algorithms based on the specific requirements of the network security system.

The discussion also examines the robustness and generalization capabilities of the developed system. The experimental evaluation shows that the models exhibit satisfactory performance in detecting unseen or adversarial attack instances, indicating a certain level of robustness. However, it is important to continue exploring additional techniques, such as anomaly

detection and outlier analysis, to further enhance the system's ability to detect evolving attack strategies and zero-day attacks.

Real-time monitoring capabilities integrated into the system proved effective in promptly detecting and responding to emerging attacks in a live network environment. The low response times observed in the experiments demonstrate the system's efficiency in real-time attack detection and mitigation. This capability is crucial in minimizing the potential damage caused by attacks and enhancing network security.

While the results obtained from this project are promising, there are certain limitations to consider. The performance of the system heavily relies on the quality and diversity of the dataset used for training and evaluation. Acquiring a more comprehensive and representative dataset, including a wider range of attack instances, would enhance the system's ability to generalize to different network environments and attack scenarios. Additionally, the project focused on offline analysis of network traffic data, and further research is needed to investigate online learning techniques that can adapt the system to evolving attack patterns in real-time.

6. Conclusion

In conclusion, the proposed system successfully developed a machine learning-based system for the detection of cyber-attacks and network intrusions. The experimental results demonstrated the effectiveness of the system in accurately detecting and distinguishing between normal network behavior and various types of attacks.

The machine learning models achieved high detection accuracy, indicating their ability to capture and recognize the underlying patterns and characteristics of different attack types. The low false positive rate minimized false alarms and unnecessary disruptions, improving the efficiency of the detection system.

The performance evaluation revealed variations in detection rates for different attack types, highlighting the challenges associated with detecting complex or subtle attack patterns. This insight provides directions for further research and refinement of the models to improve detection performance for specific attack types.

Comparisons between different machine learning algorithms or techniques provided valuable insights into their strengths and weaknesses. Decision tree-based algorithms demonstrated good accuracy but were prone to higher false positive rates, while neural network models showed superior performance in detecting complex attack patterns at the cost of increased computational resources.

The integration of real-time monitoring capabilities into the system proved effective in promptly detecting and responding to emerging attacks in a live network environment, with low response times indicating the system's efficiency in real-time attack detection and mitigation.

However, it is important to acknowledge the limitations of the project. The quality and diversity of the dataset used for training and evaluation play a crucial role in the system's performance. Acquiring a more comprehensive and representative dataset would enhance the system's ability to generalize to different network environments and attack scenarios. Additionally, further

research is needed to explore online learning techniques that can adapt the system to evolving attack patterns in real-time.

References

[1] Z. N. Zarandi and I. Sharifi, "Detection and Identification of Cyber-Attacks in Cyber-Physical Systems Based on Machine Learning Methods," 2020 11th International Conference on Information and Knowledge Technology (IKT), 2020, pp. 107-112

[2] Nurjahan, F. Nizam, S. Chaki, S. Al Mamun and M. S. Kaiser, "Attack detection and prevention in the Cyber Physical System," 2016 International Conference on Computer Communication and Informatics (ICCCI), 2016, pp. 1-6

[3] Ding Chen, Qiseng Yan, Chunwang Wu and Jun Zhao, "SQL Injection Attack Detection and Prevention Techniques Using Deep Learning," Journal of Physics: Conference Series, Volume 1757, International Conference on Computer Big Data and Artificial Intelligence (ICCBDAI 2020) 24-25 October 2020, Changsha, China

[4] Ercan NurcanYılmaz, SerkanGönen, "Attack detection/prevention system against cyber-attack in industrial control systems," Computers & Security Volume 77, August 2018, Pages 94-105

[5] Arpitha. B, Sharan. R, Brunda. B. M, Indrakumar. D. M, Ramesh. B. E, "Cyber Attack Detection and notifying system using ML Techniques," International Journal of Engineering Science and Computing (IJESC), Volume 11, Issue No.06

[6] Yirui Wu, Dabao Wei, and Jun Feng, "Network Attacks Detection Methods Based on Deep Learning Techniques: A Survey," Security Threats to Artificial Intelligence-Driven Wireless Communication Systems, 2020.

[7] Rafał Kozik, Michał Choraś, "Machine Learning Techniques for Cyber Attacks Detection," Image Processing and Communications Challenges 5, pp 391-398, Springer International Publishing Switzerland 2014.

[8] Nutjahan, Farhana Nizam, Shudarshon Chaki, Shamim Al Mamun, M. Shamim, "Attack Detection and Prevention in the Cyber Physical System," 2016 International Conference on Computer Comrnunication and Informatics (IEEE -2016), Jan. 07 - 09, 2016, Coimbatore, India

[9] Yong Fang, Cheng Huang, Yijia Xu and Yang Li, "RLXSS: Optimizing XSS Detection Model to Defend Against Adversarial Attacks Based on Reinforcement Learning," Future Internet 2019.

[10] Pratik Rajendra Chougule, Aniket Sanjay Kumbhar, Vinayak Vasant Pachange, Karan Dinkar Phonde, S. P. Phadtare, "Phishing Websites Detection using Python," Journal of Web Development and Web Designing, Volume-5, Issue-2 (May-August, 2020)

[11] Rishikesh Mahajan, Irfan Siddavatam, "Phishing Website Detection using Machine Learning Algorithms," International Journal of Computer Applications (0975 – 8887) Volume 181 – No. 23, October 2018

[12] Vishnu. B. A, Ms. Jevitha. K. P, "Prediction of Cross-Site Scripting Attack Using Machine Learning Algorithms," Conference Paper, October 2014.

[13] Shinelle Hutchinson, Zhaohe Zhang, and Qingzhong Liu, "Detecting Phishing Websites with Random Forest," Third International Conference, MLICOM 2018, Hangzhou, China, July 6-8, 2018, Proceedings
[14] Ines Jemal, Omar Cheikhrouhou, Habib Hamam and Adel Mahfoudhi, "SQL Injection Attack Detection and Prevention Techniques Using Machine Learning," International Journal of Applied Engineering Research ISSN 0973-4562 Volume 15, Number 6 (2020) pp. 569-580
[15] Fawaz A. Mereani, and Jacob M. Howe, "Detecting Cross-Site Scripting Attacks Using Machine Learning," Springer International Publishing AG, part of Springer Nature 2018