**Intrusion Detection System For Web-based Attacks**

*Project report submitted*

*in partial fulfilment of the requirement for*

*the degree of*

**Bachelor of Technology**

*in*

**Computer Science & Engineering**

*By*

**Bhawan Kumar (19bcs031)**

**Tarun Anjaneya Varma (19bcs036)**

**Sai Ganesh (19bcs055)**

**Chandra Sekhar Sarma (19bcs059)**

*Under the guidance of*

**Dr. Malay Kumar**

Assistant Professor

Department of Computer Science & Engineering



**Indian Institute of Information Technology Dharwad**

**CERTIFICATE**

It is certified that the work contained in the project report titled “Intrusion Detection System For Web-based Attacks” by “Bhawan Kumar (19BCS031)”, “Tarun Varma (19BCS036)”, “Sai Ganesh (19BCS055)” and “Chandra Sekhar (19BCS059)” has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree.

**Signature of Supervisor**

**Dr. Malay Kumar**

Assistant Professor

Department of Computer Science and Engineering

IIIT Dharwad

Indian Institute of Information Technology, Dharwad 1

**Declaration**

We declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature with date)

Bhawan Kumar

(19BCS031)

(Signature with date)

Tarun Varma

(19BCS036)

(Signature with date)

Sai Ganesh

(19BCS055)

(Signature with date)

Chandra Sekhar

(19BCS059)

Indian Institute of Information Technology, Dharwad 2

**Approval Sheet**

This project report entitled “Intrusion Detection System For Web-based Attacks” by Bhawan Kumar, Tarun Varma, Sai Ganesh and Chandra Sekhar is approved for the degree of Bachelor of Technology in Computer Science and Engineering.

**Supervisor**

**Dr. Malay Kumar**

Assistant Professor

Department of Computer Science and Engineering

IIIT-Dharwad

**Head of Department**

**Dr. Pavan Kumar**

Assistant Professor

Department of Computer Science and Engineering

IIIT-Dharwad

**Examiners**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date :\_\_\_\_\_\_\_\_\_\_\_\_

Place:\_\_\_\_\_\_\_\_\_\_\_\_

Indian Institute of Information Technology, Dharwad 3

**ABSTRACT**

The increasing prevalence of the internet and the widespread use of web applications, cybersecurity threats have become more prevalent. A wide range of attack vectors, including Distributed Denial of Service (DDoS) attacks, port scanning, web-based attacks, and patator attacks on FTP and SSH services, pose a significant threat to users and organizations. Therefore, it is essential to develop effective intrusion detection systems (IDS) to reduce the possible damage caused by these attacks.

This project aims to build a machine learning (ML) and deep learning-based IDS for web-based attack detection using the CICIDS 2017 dataset. We created a balanced dataset from the original CICIDS 2017 dataset and used an interesting yet efficient feature reduction technique known as EFFST (Ensemble of filter feature selection technique) to detect web attacks. The selected features were used to train a multi-class classification model. The effectiveness of our approach was evaluated using real-world data, and the results showed a high level of accuracy in detecting various forms of web-based attacks.

The inclusion of AI techniques in our IDS provides a more proactive defence against evolving threats. Our approach to detecting web-based attacks can contribute to the development of an efficient and accurate system for detecting such attacks. This is essential for ensuring the security of online systems, which are vulnerable to attacks. The publicly available CICIDS 2017 dataset enables easy reproducibility and comparison of results, making it useful for researchers and cyber security professionals interested in improving the efficacy of IDS in detecting web-based attacks.

The techniques and approach presented in this project report can be used to strengthen cyber security measures and protect online systems from malicious attacks. Continued research in this area can lead to more effective IDS and cyber security measures that can better defend against a range of attack vectors. It is essential to develop a proactive approach to cyber security to safeguard online systems from the various threats that are present in the digital landscape.

Indian Institute of Information Technology, Dharwad 4

**Table of Contents**

**CHAPTER 1** **6-9**

* 1. Introduction 6
  2. Problem Statement 7
  3. Motivation 8
  4. Brief Workflow 9

**CHAPTER 2**  **10-13**

2.1. Literature Survey 10

* + 1. Machine Learning Approach 10
    2. EFFST Approach 11
    3. RNN and LSTM Approach 12

**CHAPTER 3**  **14-23**

3.1. Architecture 14

3.2. Dataset 16

* + 1. Binary Classification Dataset 17
    2. Multi Class Classification Dataset 17
    3. RNN and LSTM Approach 12
  1. Range of Attacks in our Dataset 19
  2. Algorithms 21
  3. Implementation 23

**CHAPTER 4 24-25**

4.1. Experimental Analysis / Results 24

**CHAPTER 5 26-30**

5.1. Conclusion 26

5.2. Contributions of this work 27

5.3. Future Scope 28

5.4. References 29

Indian Institute of Information Technology, Dharwad 5

**CHAPTER 1**

* 1. Introduction

With the growing dependence on digital technology, there has been a corresponding increase in cyber-attacks. Web-based attacks, in particular, are becoming increasingly prevalent and sophisticated, posing a significant threat to the security of online systems. Intrusion detection systems (IDS) are potential solutions for preventing cyber-attacks. IDS are designed to identify potential security breaches by monitoring network traffic and identifying patterns that indicate suspicious or anomalous behavior. It is expected and accepted by everyone that ML and DL can be really helpful in detecting intrusions in network traffic. In this paper, an ML and DL-based IDS is proposed for web-based attack detection using the CICIDS 2017 dataset.

Intrusion Detection System (IDS) has a rich history in the field of cybersecurity, serving as a critical component in the defence against cyber threats. IDS monitors and analyses network or system activities to detect and respond to suspicious or malicious behaviour. The origins of IDS can be traced back to the 1980s when computer networks were rapidly expanding, and the need for securing them became apparent. Early IDS systems primarily relied on signature-based detection, where known patterns of malicious activities, known as signatures, were pre-defined in the system. When network traffic or system events matched these signatures, the IDS would raise an alert to notify the security team.

As cyber threats evolved and became more sophisticated, traditional signature-based IDS became less effective in detecting new and unknown threats. In response, newer techniques were developed, such as anomaly-based detection, which involved monitoring for abnormal behaviours or activities that deviated from the expected norm. This approach helped to detect previously unknown threats or zero-day vulnerabilities that lacked known signatures.

IDS also evolved in terms of deployment options. Initially, IDS were deployed as standalone appliances or software running on dedicated servers. However, with the advancement of technology, IDS started being integrated into other security devices, such as firewalls, routers, and switches, as well as being offered as cloud-based services.

Another significant advancement in IDS was the integration of machine learning and artificial intelligence (AI) technologies. These techniques allowed IDS to learn and adapt to new threats and behaviours, making them more effective in identifying and responding to previously unknown attacks. Terms like supervised learning, unsupervised learning, deep learning, and neural networks are commonly associated with machine learning-based IDS. IDS has played a crucial role in detecting and mitigating various cyberattacks, including SQL Injection, Distributed Denial of Service (DDoS), Cross-Site Scripting (XSS), buffer overflow attacks, malware infections, brute-force attacks, insider threats, and more. By analysing network traffic, system logs, and other security events, IDS can identify patterns that deviate from normal behaviour and raise alerts to prompt investigation and response.

Modern IDS often utilize a combination of signature-based, anomaly-based, and machine learning-based detection techniques to detect known and unknown attacks. They also leverage

Indian Institute of Information Technology, Dharwad 6

threat intelligence feeds, behavioural analysis, and other advanced methods to enhance their detection capabilities and stay effective against evolving cyber threats. In recent years, IDS has been integrated with other cybersecurity technologies, such as Security Information and Event Management (SIEM) systems, and threat intelligence feeds, for more comprehensive threat detection and response.

* 1. Problem Statement

Web-based attacks have become a growing concern for individuals and organizations alike. According to recent statistics, there are over 100,000 web-based attacks occurring every day, with an average of 1 attack every 39 seconds. These attacks include a wide range of tactics such as SQL injection, cross-site scripting (XSS), and phishing attacks, which can result in significant financial losses and damage to reputation. In fact, it is estimated that the total cost of cybercrime will reach $6 trillion annually by 2021, making it clear that web-based attacks pose a significant threat to businesses of all sizes. With the increasing sophistication of these attacks, it has become essential for organizations to implement advanced security measures such as intrusion detection systems to protect their web applications and networks. By utilizing an effective IDS, businesses can proactively detect and prevent web-based attacks, minimizing the risk of financial and reputational damage.

To address the challenges associated with web-based attacks, we are developing an intrusion detection system (IDS) that can effectively detect and prevent such attacks. Our IDS is designed to analyse network traffic and identify suspicious activity that may indicate an ongoing attack. We aim to develop an IDS that can accurately differentiate between legitimate traffic and malicious traffic, minimizing false positives and false negatives. Additionally, our IDS will be designed to handle high volumes of traffic and complex attacks while maintaining high performance. With our IDS, we hope to provide businesses with a reliable and effective solution for protecting their web applications and networks from the growing threat of web-based attacks.

“In the hacking world, security is more of a response than a proactive measure. They wait for hackers to attack and then they patch, based on the attacks.”

IDS for web-based attacks operate by analysing network traffic and looking for patterns or signatures of known attacks. These systems use a variety of techniques to identify web-based attacks, including signature-based detection, anomaly-based detection, and heuristic-based detection. Signature-based detection relies on a database of known attack patterns, while anomaly-based detection compares network behaviour against established baseline activity. Heuristic-based detection uses machine learning algorithms to identify new attack patterns based on previously identified patterns. Despite these methods, web-based attacks are often challenging to detect, as attackers can use obfuscation techniques to evade detection by IDS.

Given the increasing sophistication of web-based attacks, IDS for web-based attacks must be able to analyse a broad range of web traffic, including encrypted traffic. This requires specialized IDS that can decrypt and analyse SSL/TLS traffic without affecting network performance. Additionally, IDS for web-based attacks must also be capable of detecting and preventing attacks that target the application layer. This is crucial as web-based attacks are often directed at the application layer, which makes them difficult to detect by traditional IDS that focus on the network

Indian Institute of Information Technology, Dharwad 7

layer. Overall, IDS for web-based attacks must be versatile and adaptable to emerging threats, given the ever-evolving nature of cyber threats in the modern landscape.

In conclusion, IDS for web-based attacks are essential for protecting web-based applications from cyber threats. As web-based attacks become more sophisticated and prevalent, IDS must evolve to keep up with the ever-changing threat landscape. This requires advanced IDS solutions that can detect and prevent a broad range of web-based attacks, including those targeting the application layer. Additionally, IDS for web-based attacks must be capable of decrypting SSL/TLS traffic without impacting network performance, making them more versatile and effective in detecting potential cyber threats. Ultimately, effective IDS for web-based attacks are essential for maintaining the security and integrity of web-based applications, and the data they contain.

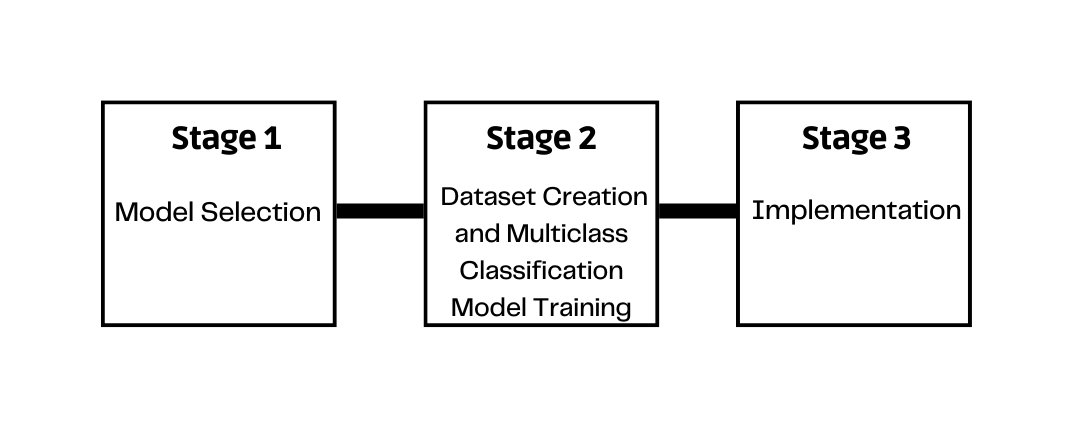
* 1. Motivation of Project

The rapid growth of web-based applications and services has made the internet an integral part of our daily lives. The threat of cyber-attacks has also increased. Web-based attacks such as SQL injection, cross-site scripting (XSS), and distributed denial of service (DDoS) are some of the most common types of attacks that can lead to devastating consequences for individuals and organizations. These attacks can cause data breaches, financial losses, and reputational damage, making it crucial for businesses to protect their web applications and networks from such threats. Traditional security measures such as firewalls, antivirus software, and intrusion prevention systems are no longer sufficient to protect against the ever-evolving nature of web-based attacks.

To mitigate the risk of these attacks, intrusion detection systems (IDS) have emerged as an effective solution for detecting and preventing web-based attacks. An IDS analyses network traffic and identifies patterns that may indicate an ongoing attack. However, despite the benefits of intrusion detection systems, there are still challenges that need to be addressed. It has been always difficult to detect and differentiate between safe and unsafe traffic which. One of the main challenges is the difficulty of identifying and differentiating between legitimate traffic and malicious traffic, which can result in false positives or false negatives. Moreover, the performance of intrusion detection systems can be impacted by factors such as network speed, volume of traffic, and the complexity of the attack. Therefore, there is a need for more advanced intrusion detection systems that can effectively detect and prevent web-based attacks while minimizing false alarms and maintaining high performance.

Indian Institute of Information Technology, Dharwad

* 1. Brief Workflow



Our project consists of three stages, each with its own set of goals and objectives.

Stage 1: Data Pre-processing and Model Selection

In the first stage, we selected a subset of the CICIDS 2017 dataset and extracted features using the EFFST framework. We then created binary classification models to determine the most appropriate machine learning or deep learning algorithm for our project. Our aim in this stage was to identify the best algorithm that could provide the highest accuracy in detecting web-based attacks.

Stage 2: Dataset Creation and Multiclass Classification Model Training

Once we identified the most appropriate algorithm, we progressed to stage 2. Here, we created a new dataset that was efficient and compatible with our machines, and extracted features from it using our framework. We then used the results from stage 1 to create multiclass classification models. Our goal in this stage was to improve the accuracy of the models and provide a more proactive defence against evolving web-based threats.

Stage 3: Real Traffic Implementation and Feature Extraction

In the final stage, we plan to implement real traffic using available tools and extract features from it. We aim to optimize the implementation process for efficiency so that we can give the extracted features to our machine to predict the output accurately. This stage will allow us to test the effectiveness of our models using real-world data and determine how well our IDS performs in detecting web-based attacks.

Indian Institute of Information Technology, Dharwad 9

**CHAPTER 2**

This section contains an overall overview of all the techniques, algorithms and machine learning models used in various aspects of the application. Here some algorithms are directly used in the applications while some of them have their parameters slightly changed and some models retrained to achieve the required accuracy.

* 1. Literature Survey

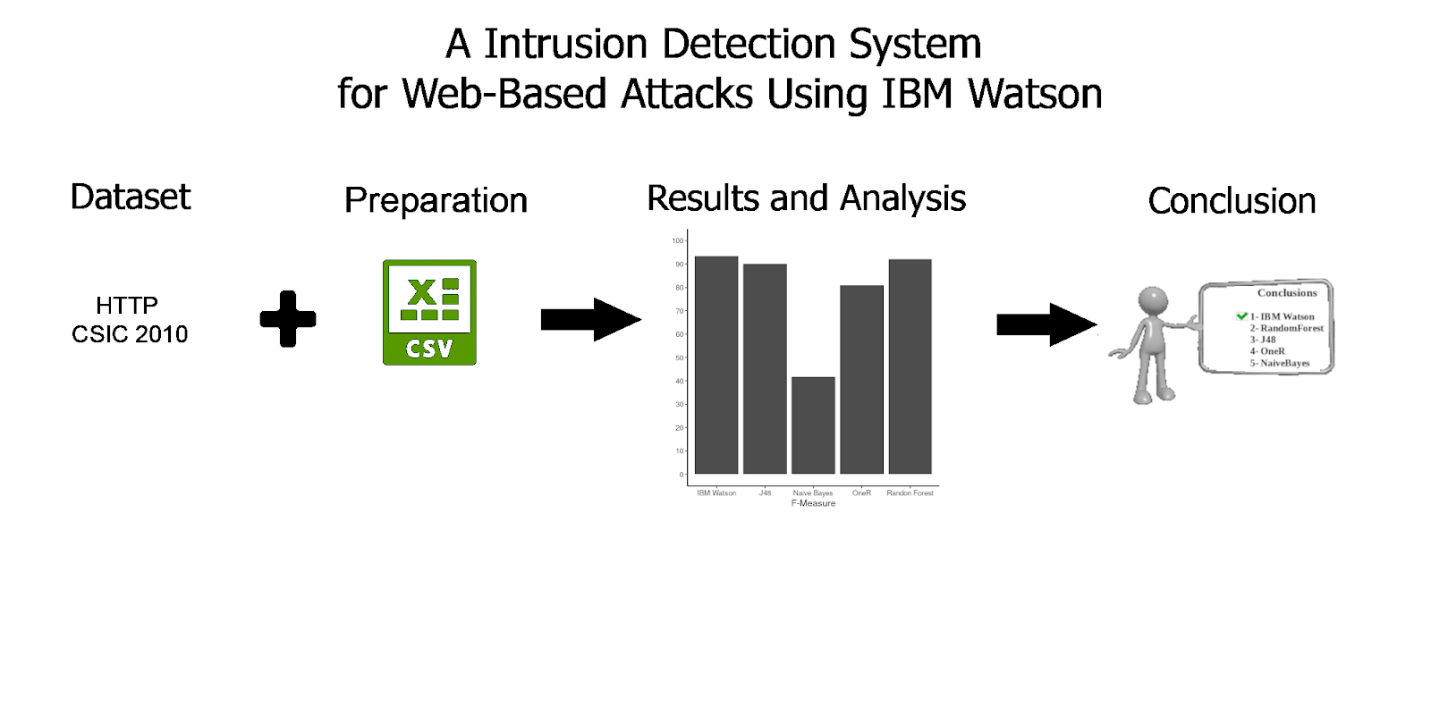
Extensive research was conducted before finalizing the approach for this project. Numerous papers were examined and analysed for their ideas and methodologies, from which we drew inspiration and borrowed various concepts. These included, but were not limited to, the use of Ensemble Filter Feature Selection Technique (EFFST), the identification of the best performing ML & DL algorithms, and the implementation of RNN's & LSTMs and many more.

­­

* + 1. Machine Learning Approach

The paper titled "An Intrusion Detection System for Web-Based Attacks Using IBM Watson" proposes an ML-based approach to detect web-based attacks. The authors used the popular CSIC 2010 dataset. The goal of the study is to develop an IDS that can detect various web-based attacks such as SQL injection, cross-site scripting (XSS), command injection, and others. The authors decided upon using the random forest algorithm in order to reduce the computation burden by only using certain features in training the model for this task. After selecting the features, SVM was used in performing the classification.

To further enhance the accuracy of the IDS, the authors incorporated IBM Watson's Natural Language Processing (NLP) capabilities to extract meaningful information from network logs. The extracted information was then used to enhance the feature selection process. The results of the study showed that their architecture fetched accuracies of 98.6% for SQL injection attacks, 99.9% for XSS attacks, and 100% for command injection attacks, with an overall accuracy of 99.8%



Indian Institute of Information Technology, Dharwad 10

The study highlights the effectiveness of using machine learning algorithms for developing IDSs that can detect web-based attacks. The authors also demonstrated the usefulness of NLP techniques to extract information from network logs and enhance the feature selection process. Overall, the proposed IDS can provide a more robust and efficient solution for detecting web-based attacks, and it can be implemented in various organizations to protect against cyber threats.

One of the strengths of this study is that the authors used a real-world dataset, which makes the proposed IDS more applicable to real-world scenarios. Additionally, the use of IBM Watson's NLP capabilities is an innovative approach to enhance the accuracy of the IDS. However, one of the drawbacks of this research was that there was no comparative study of this IDS with the state-of-art IDS. Therefore, it is unclear how the proposed IDS compares to other IDSs in terms of performance and accuracy. Nonetheless, the study provides a valuable contribution to the development of IDSs that can effectively detect web-based attacks.

This has inspired us in using ML concepts for attack classification and this gave very good performance metrics as well. We have also gone through several other pacers of the same kings where they have used various ML techniques. So, we have considered to train our model based on all these best performing algorithms like SVM, Naive bayes, Random Forest, KNN, decision tree etc.

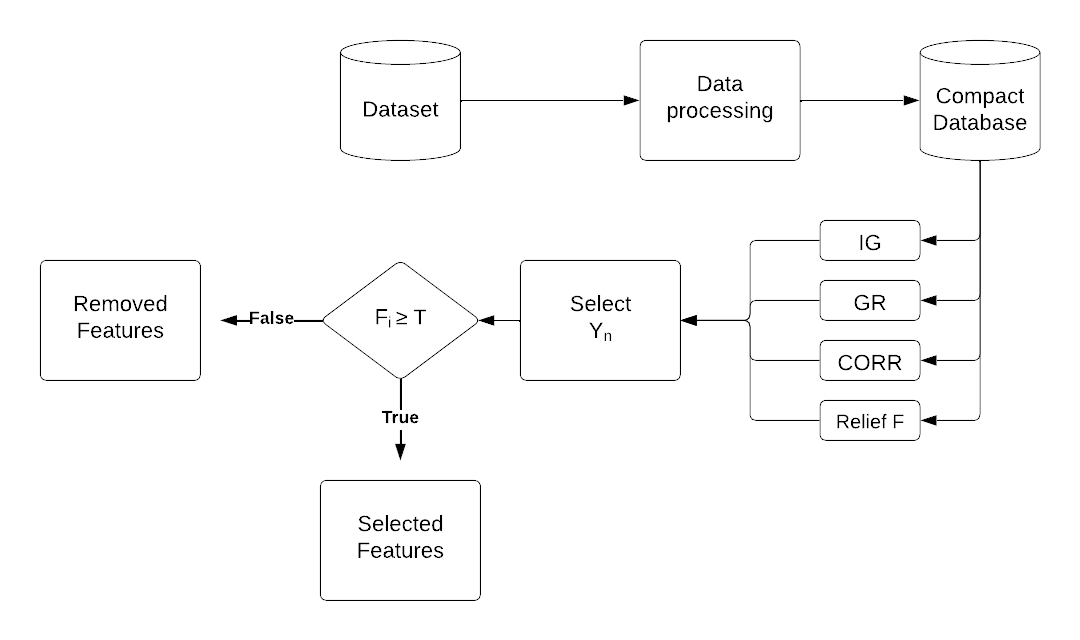
* + 1. Ensemble of Filter Feature Selection Techniques & Approach

The paper "Towards an intrusion detection system for detecting web attacks based on an ensemble of filter feature selection techniques" proposes an ensemble of filter feature selection technique (EFFST) for IDS based web attacks detection. The proposed system uses a dataset consisting of HTTP requests and responses captured from a web server. The EFFST technique is used to select a subset of features that are most relevant for detecting attacks. The selected features are then used to train various classifiers, including KNN, DT, RF and also the age old SVM.

The KDD Cup 99 dataset was used for this purpose and a custom dataset consisting of real HTTP traffic captured from a web server. The performance of the system was evaluated based on various metrics, including accuracy, precision, recall, F1-score, and Area Under the ROC Curve (AUC). The results show that the proposed system achieved an accuracy of 98.2% and an AUC score of 0.998 on the KDD Cup 99 dataset and an accuracy of 97.3% and an AUC score of 0.991 on the custom dataset.

The results of the study suggest that the proposed system can be used as an effective IDS for detecting web-based attacks.

Indian Institute of Information Technology, Dharwad 11



In conclusion, the paper proposes a special feature extraction technique which can be integrated in an IDS. The proposed system achieved high accuracy and AUC score on both the KDD Cup 99 dataset and a custom dataset consisting of real HTTP traffic. The results of the study suggest that the proposed system can be used as an effective IDS for detecting web-based attacks and outperforms other existing IDSs.

The concept of ensemble filter feature selection techniques (EFFST) proposed by Kshirsagar and Kumar in their paper "Towards an intrusion detection system for detecting web attacks based on an ensemble of filter feature selection techniques" has inspired us in our project of building an IDS for web-based attacks. We believe that using EFFST can help improve the performance of our system by selecting the most relevant features and reducing the dimensionality of the dataset. By doing so, we can achieve better accuracy in detecting attacks and reduce the false positive rate. We plan to incorporate EFFST as part of our methodology and evaluate its effectiveness in improving the performance of our IDS.

* + 1. RNN and LSTM Approach

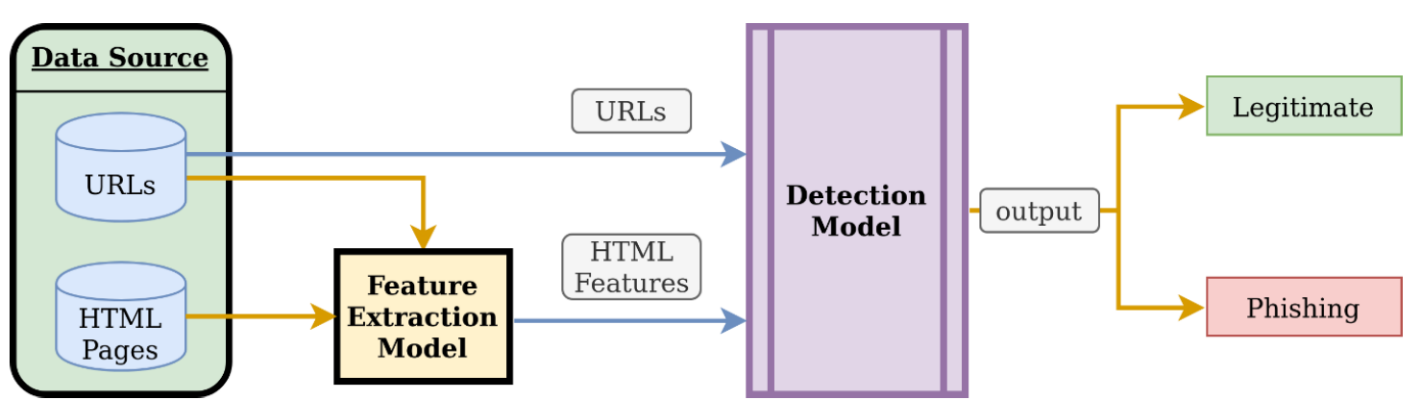
Hong and Lee (2020) presented a Deep Learning (DL) based Intrusion Detection System (IDS) for web applications that uses a LSTM model but along with attention mechanisms. The proposed IDS was designed to detect web-based attacks such as SQL injection, cross-site scripting, and command injection. The researchers trained the LSTM model on a dataset of normal and attack traffic and evaluated its performance on a separate test set. The system achieved an accuracy of 99.2% in detecting attacks, outperforming traditional Machine Learning (ML) based IDS.

The proposed system used an attention mechanism to focus on the most relevant parts of the input data, which helped to improve the accuracy and efficiency of the system. The most important features in the input data were thus handled using the attention concept. This helped to reduce the amount of noise in the data and improve the overall performance of the system.

Indian Institute of Information Technology, Dharwad 12

The research also states that the proposed DL-based IDS does the classification with very fewer false positive rates, which is important for practical applications. False positives can lead to unnecessary disruptions to legitimate traffic, and a low false positive rate is therefore desirable in an IDS.

Overall, this study demonstrates the potential of DL-based IDS for web applications and highlights the benefits of using advanced AI techniques such as LSTM models with attention mechanisms. The proposed system achieved high accuracy in detecting attacks while minimizing false positives, and provides a promising foundation for further research in this area



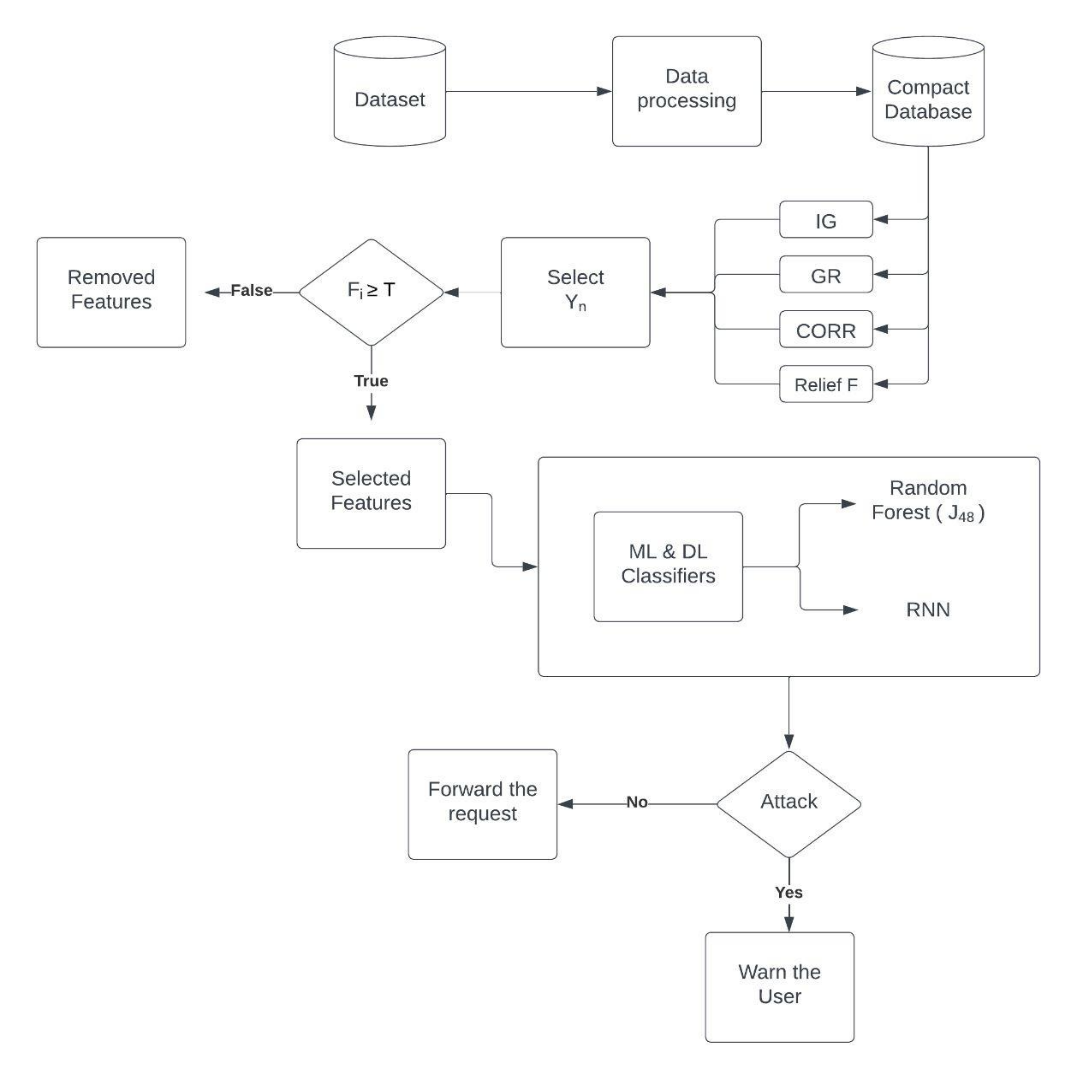
Our goal is to leverage the best-performing methods and techniques from these papers to develop an efficient and effective IDS for web-based attacks. By combining the strengths of these approaches, we aim to create a more robust and accurate IDS that can detect and prevent web-based attacks with higher accuracy and reliability than existing systems. Our IDS will incorporate deep learning-based models with attention mechanisms, ensemble filter feature selection techniques, and other advanced algorithms to provide comprehensive protection against various web-based attacks.

|  |  |
| --- | --- |
| Indian Institute of Information Technology, Dharwad | 13 |

**CHAPTER 3**

This section contains the information of all the experiments and their respective environment. It also includes information about dataset and the architecture of the project.

* 1. Architecture



The proposed architecture for the study incorporates the use of the feature reduction technique, EFFST, to refine the dataset and obtain the most relevant features. EFFST is an ensemble method that combines various filter-based feature selection techniques to obtain a more dependable set of features.

The EFFST framework begins with calculating the Information gain (IG) for each feature. IG measures the amount of information that can be gained about the class label by knowing the feature value. This technique is used to select features that have high mutual information with the class label. The next step involves calculating the Gain Ratio (GR) for each feature, which is a normalized version of the IG and takes into account the intrinsic information of the feature. GR is used to select features that have high mutual information with the class label but are not redundant.

Indian Institute of Information Technology, Dharwad 14

The third step involves calculating the correlation between each feature and its target label. Correlation is a measurement of the linear relationship of two variables and is used to select features that are highly correlated with the class label. Finally, the fourth step involves calculating the ReliefF value for each feature, which estimates the quality of a feature by looking at the difference in the feature values of the nearest neighbours with different class labels. This technique is used to select features that can effectively discriminate between different classes.

After the feature selection process using the EFFST framework, the resulting feature subset will be used as the input to develop and evaluate the performance of various machine learning and deep learning algorithms. These algorithms will be used to predict attacks on the network based on the selected features. By integrating the EFFST framework into the proposed architecture, we aim to achieve better results by focusing on the most impacting features for attack detection.

EFFST Algorithm

1. Input: Fine Tuned dataset after pre-processing
2. Declare measures enum with IG, GR, Correlation, and ReliefF.
3. Initialize an empty dictionary feature\_set.
4. For each measure in the enum:
   1. Initialize empty list top\_features
   2. Calculate this measure for every feature.
   3. Arrange this list in descending order.
   4. Then take the top 25 percentile of the list and store them in top\_features.
   5. Store this measure as a key and top\_features as a value in the feature\_set.
5. Initialize an empty dictionary feature\_frequency.
6. Take the count of every feature present in feature\_set and store them in feature\_frequency.
7. Initialize a threshold T(T = 2 in this case) and an empty list final\_feature\_set.
8. If any feature\_count in feature\_frequency >= T:
   1. Add that feature to our final\_feature\_set.
9. Output: More contributing features.

To calculate the information from the dataset that we described earlier we used the variation of these formulas such that we will get the importance of each attribute when compared with others.

**Information Gain (IG)**: It determines the relevance of a feature (i.e., an attribute or input variable) in a dataset for predicting a specific outcome or target variable. It quantifies the reduction in entropy (i.e., the degree of randomness or uncertainty) in the target variable that results from splitting the dataset based on a particular feature.

|  |
| --- |
| IG (M, N) = H (M) – H (M|N) |

H (M): Entropy (M)

H (M|N): Probability (M|N)

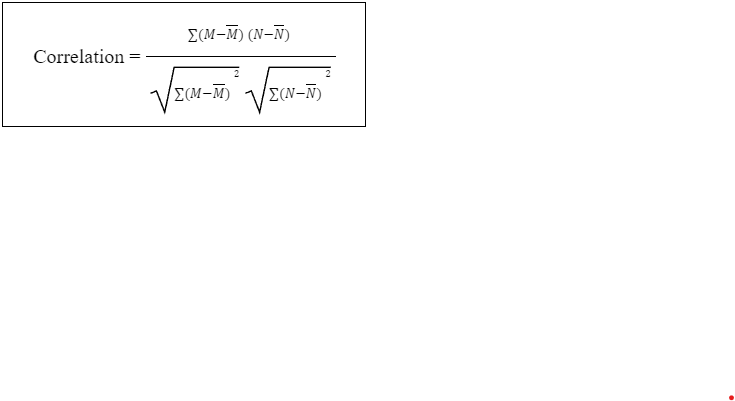
Indian Institute of Information Technology, Dharwad 15

**Gain Ratio (GR)**: GR is an addition to IG which ignores the bias of the features by applying the below formula.

|  |
| --- |
| GR (M, N) = [Gain (M, N)] / Intrinsic Information (N) |

Where, Intrinsic Information = -∑ [ |Si|/S] \* [Sⅈ|/S]

**Correlation (CR)**: It is a mathematical quantity that defines the relation between two or more variables.



**ReliefF:** It calculates the quality of a feature based on how well it helps to differentiate between examples that are similar to each other but belong to different classes.

|  |
| --- |
| Wi= Wi- xi- nearHiti2+xi- nearMissi2 |

Once we apply the above methods we will get our desired dataset on which we can train our ML/DL Model. Once we are ready with the data we will be applying various Ml and DL techniques. RNNs would be our main focus since it has performed very well in previous situations as suggested in the literature. We will also try various ML algorithms and finally list the best-performing algorithms and their results from both domains.

* 1. Dataset

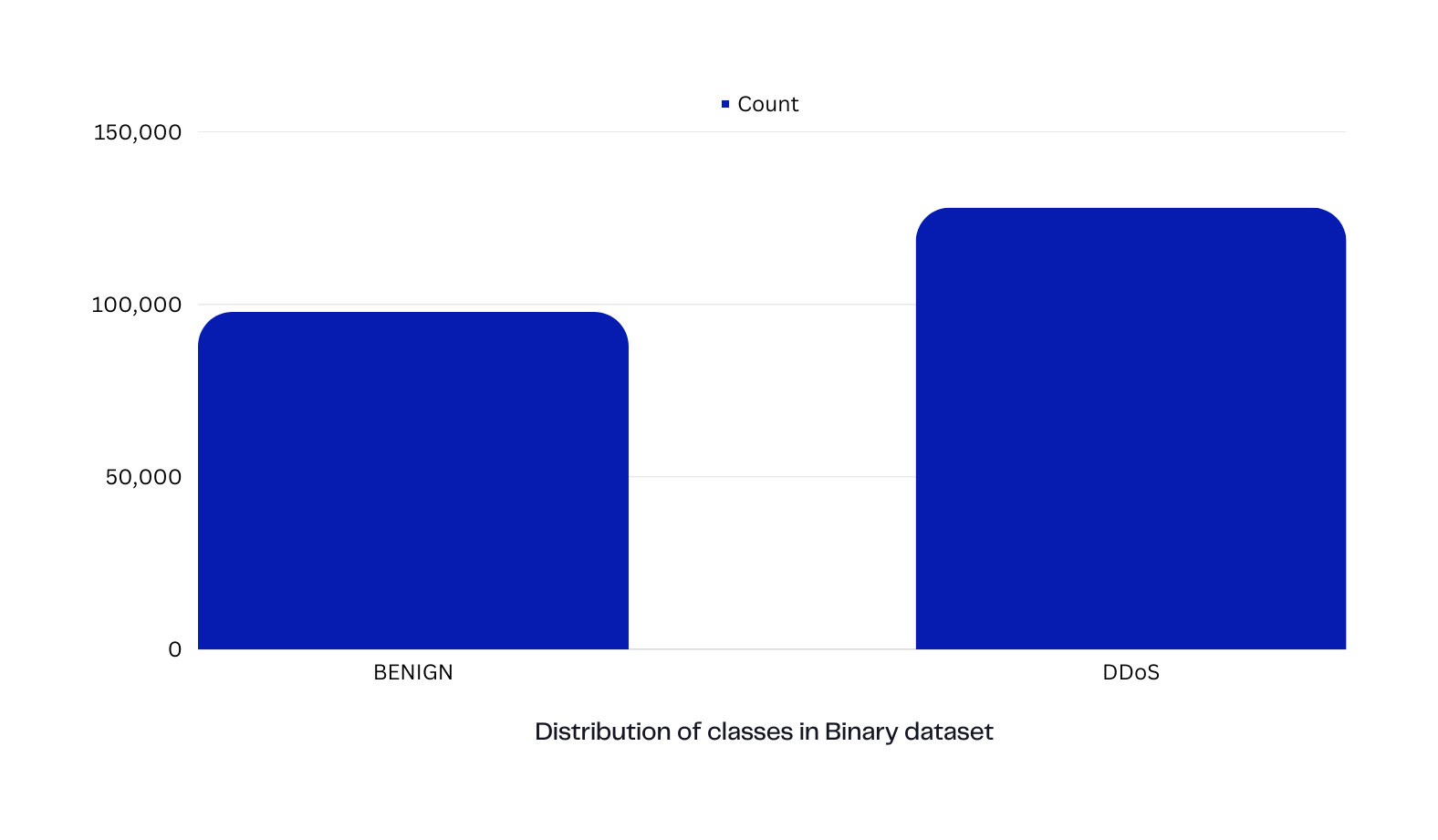
The CIC-IDS 2017 dataset is a publicly available dataset for cybersecurity. It was released in 2017, with the aim of providing a benchmark dataset for researchers to develop and test intrusion detection systems (IDS) in a controlled lab environment. The dataset contains network traffic data generated using the Stratosphere IPS network security monitoring platform and frameworks such as Metasploit, Nmap, and hping3. The traffic consists of both normal and malicious network traffic generated by various attacks, including botnets, DoS/DDoS, port scans, brute-force attacks, network scans, and Denial-of-Service (DoS) attacks.

The dataset consists of over 2 million network traffic records captured over seven days. It is split into two subsets: a training set that contains around 80% of the data and a test set that contains the remaining 20%. The dataset includes various network protocols such as TCP, UDP, ICMP, and others.

Indian Institute of Information Technology, Dharwad 16

* + 1. Binary Classification Dataset

We have considered a subset of the CICIDS dataset that contains only two classes: benign and DDoS. This subset has a total length of around 225,745, where 128,027 instances belong to the DDoS class and 97,718 instances belong to the benign class. Initially the dataset was considered to have 78 features, which were reduced and selected 19 useful features by using EFFST framework. By using this newly obtained dataset we have trained binary classification ML and DL models.

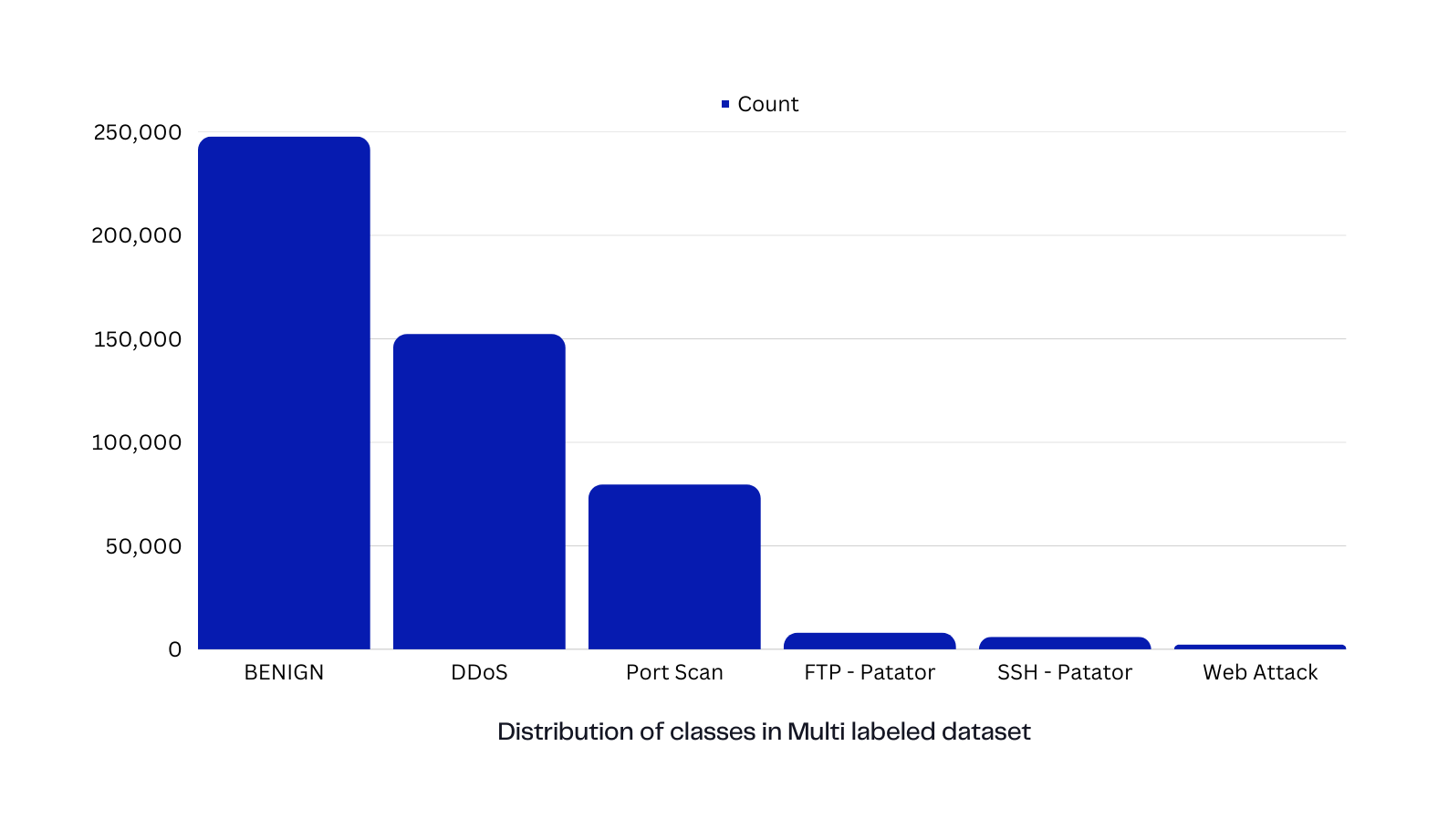


* + 1. Multi Class Classification Dataset

The original CICIDS dataset contains over 2 million records, which can be computationally challenging to analyse. Therefore, we created a smaller subset of 500,000 records by selecting an equal number of benign records and records containing different types of attacks from the original dataset. This subset allowed for a manageable dataset size while still providing a variety of attack types for analysis.

After obtaining the normalized dataset, the EFFST framework was applied to select the most informative features for intrusion detection. The study then moved on to a multi-class classification problem to evaluate the performance of different feature selection methods in identifying informative features for detecting multiple types of attacks.

Indian Institute of Information Technology, Dharwad 17



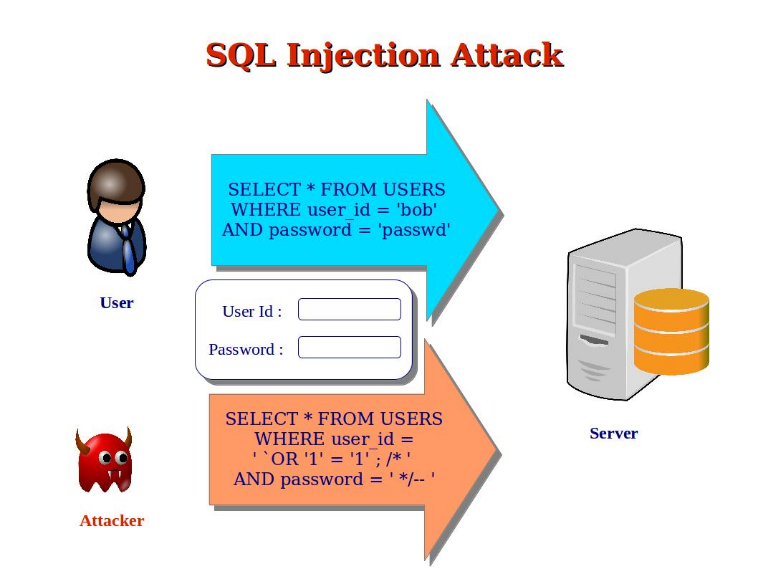
The following features were included in our dataset:

* Total Backward Packets
* Total Length of Backward Packets,
* Backward Packet Length Max
* Flow Packets
* Backward Packets,
* Packet Length Variance
* PSH Flag Count
* Down/Up Ratio
* Sub flow Backward Packets
* Sub flow Backward Bytes
* Init Win bytes forward
* Act data packet forward
* Minimum seg size forward
* Flow Duration
* Flow IAT Mean
* Flow IAT Max
* Forward IAT Mean
* Forward IAT Max
* Idle Std

These can be defined as the features of interest which were fed to the model while training.

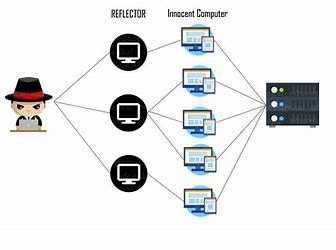
|  |  |
| --- | --- |
| Indian Institute of Information Technology, Dharwad | 18 |

* 1. Range of Attacks in our Data set
     1. SQL Injection



SQL injection is a type of web-based attack that targets databases. In an SQL injection attack, an attacker uses malicious code to exploit vulnerabilities in a web application's input validation mechanisms. The attacker inputs SQL statements that are executed by the database, allowing them to access, manipulate or steal sensitive data stored in the database they can even modify or delete data, or even take control of the entire database. Protecting against SQL injection attacks requires robust input validation mechanisms and properly securing the database.

* + 1. Distributed Denial of Service (DDoS)

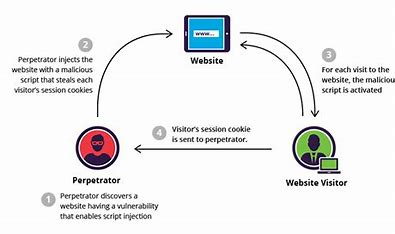


A Distributed Denial of Service (DDoS) attack is a type of cyberattack where a large number of compromised computers, known as bots or zombies, flood a targeted website or server with traffic, making it inaccessible to legitimate users. DDoS attacks are a popular weapon for cybercriminals, hacktivists, and other malicious actors because they can be easily launched and have the potential to cause significant disruption to a website or service. DDoS attacks can take many different forms, including amplification attacks, which exploit vulnerable servers to increase the volume of traffic

Indian Institute of Information Technology, Dharwad 19

sent to the target, and application-layer attacks, which target specific vulnerabilities in web applications. Protecting against DDoS attacks typically involves a combination of network monitoring, traffic filtering, and the use of specialized DDoS mitigation tools and services. It's important for organizations to have a response plan in place in case of a DDoS attack to minimize damage and reduce downtime.

* + 1. Cross Site Scripting (XSS)



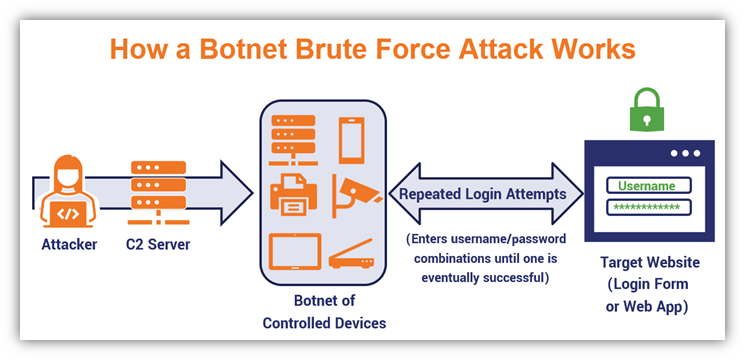
In an XSS attack, is one where a harmful code is sent as a script into a less secured application by the hacker. When other users view the affected web page, the script is executed by their browser, allowing the attacker to steal their sensitive data or perform other malicious actions. XSS attacks can take many different forms, including reflected XSS, where the malicious code is reflected back to the user from a web server, and stored XSS, where the code is stored on the server and executed every time a user accesses the affected page. Protecting against XSS attacks requires careful input validation and output encoding of user-generated content, as well as regular security audits to identify and patch vulnerabilities.

* + 1. Buffer Overflow

Buffer Overflow is a type of attack that exploits vulnerabilities in software applications that fail to properly manage the memory allocation for system resources. In this, an attacker sends data that is larger than the memory buffer allocated for it, causing the excess data to overflow into adjacent memory locations. This can cause the system to crash or allow the attacker to execute malicious code that can compromise the system or steal sensitive information. Buffer overflow attacks are particularly dangerous because they can be difficult to detect and exploit vulnerabilities in widely used software, such as web browsers or operating systems. Protecting against buffer overflow attacks requires robust input validation mechanisms and the use of secure programming techniques, such as bounds checking and memory protection. It's also important to keep software applications up to date with the latest security patches and updates to prevent known vulnerabilities from being exploited.

Indian Institute of Information Technology, Dharwad 20

* + 1. Brute force



A brute force attack is a type of cyberattack that involves trying multiple combinations of usernames and passwords until the correct combination is found. In a brute force attack, an attacker uses automated software or scripts to generate a large number of password guesses in rapid succession, often using a list of common passwords or a dictionary of words. This method allows attackers to bypass authentication mechanisms and gain access to systems or accounts without requiring any prior knowledge of the target. Brute force attacks are particularly effective against weak passwords and poorly secured systems. Protecting against brute force attacks requires implementing strong password policies, such as using long, complex, and unique passwords, limiting the number of failed login attempts, and using two-factor authentication. It's also important to regularly monitor systems for unusual login activity and to implement network security measures, such as firewalls and intrusion detection systems, to detect and block brute force attacks.

* 1. Algorithms
     1. Random Forest

Random Forest is a type of machine learning algorithm that is often used to solve classification and regression problems. Each decision tree in the random forest is built using a random subset of the available data and features. By doing this, the decision trees can be less likely to overfit the data and better generalize to new, unseen data. When making a prediction with a random forest, each decision tree in the ensemble generates its own prediction. The final output of the random forest is then determined by combining the predictions from all the individual decision trees. One of the key benefits of Random Forest is its capacity to tackle large amounts of data points, while also being resistant to overfitting. The algorithm also provides useful feature importance metrics, which can help identify the most important features in the data.

Indian Institute of Information Technology, Dharwad 21

* + 1. XGBoost

XGBoost (Extreme Gradient Boosting) is a popular machine learning algorithm that is used for both regression and classification tasks. It is a type of ensemble learning method that combines multiple weak models, typically decision trees, into a single strong model. XGBoost is known for its efficiency, accuracy, and flexibility.

XGBoost also gives the benefit of holding large data chunks while being trained. It does this by using a technique called gradient boosting, which involves iteratively adding weak models to the ensemble and adapting the weights according to those of the data points to enhance the overall performance. XGBoost also includes several regularization techniques to prevent overfitting and improve generalization.

* + 1. SVM

The basic idea behind SVMs is to find the best separating hyperplane between two classes of data points. This is done by maximizing the margin between the hyperplane and the closest data points, which are known as support vectors.

One of the strengths of SVMs is their ability to handle high-dimensional data with relatively few training examples. SVMs are also versatile, as they can transform the data into higher dimensions using kernels, owing them for a non-linear separation of the classes. Additionally, SVMs have been shown to perform well in many different applications, including image recognition, text classification, and gene expression analysis.

* + 1. Decision Trees

A decision tree is a type of algorithm used in machine learning that helps to predict outcomes by dividing a dataset into smaller, more manageable subsets. The algorithm does this by looking at the values of different features in the data and determining which ones are most important for predicting the outcome or target variable.

The algorithm then recursively splits the data into subsets based on these features until the resulting subsets are as homogeneous as possible with respect to the target variable. This means that each subset has similar values for the target variable.

The final result of a decision tree is a set of if-then rules that can be used to predict the outcome of new data. These rules essentially provide a roadmap for making decisions based on the features of the data. The algorithm uses these rules to determine the outcome of new data points by following the path through the decision tree that corresponds to the values of the features in the new data.

One of the advantages of decision trees is their ability to handle both numerical and categorical data and to perform well even in the presence of noisy or irrelevant features. Decision trees are also easy to interpret and visualize, which can be useful for understanding how the algorithm is making its predictions. However, decision trees are prone to overfitting and can be sensitive to small changes in the data or the decision-making criteria.

Indian Institute of Information Technology, Dharwad 22

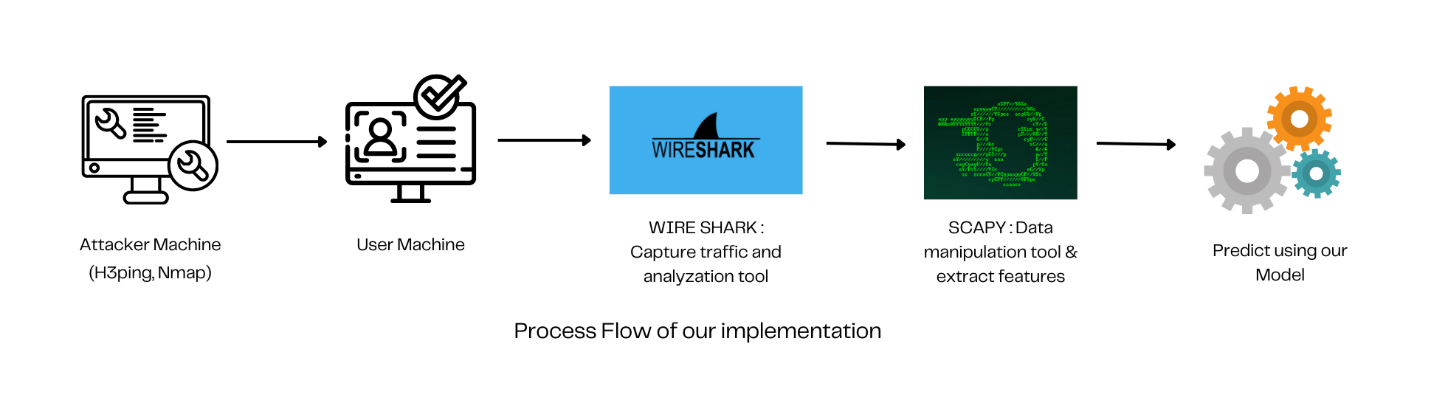
* + 1. KNN

The algorithm works by finding the K closest data points in the training set to a given test point, based on a distance metric such as Euclidean distance, and using the majority vote of the K neighbours to classify the test point or to predict its value.

KNN is a non-parametric algorithm, meaning that it does not make any assumptions about the underlying distribution of the data. It is also a lazy algorithm, meaning that it does not actually learn a model from the training data, but instead stores the entire training set in memory and uses it to make predictions at test

time. One of the advantages of KNN is its simplicity, which makes it use and apply. However, KNN can be costly in terms of calculations, particularly for large datasets or high-dimensional data, and it can be sensitive to the choice of distance metric and the value of K.

* 1. Implementation



We set up a virtual environment using virtual machines to simulate network traffic. One machine was designated as the attacker, while the other was the victim. We used various tools such as Nmap, hping3, and Metasploit to generate network traffic and send it to the victim machine. The traffic was then captured using Wireshark, a network analyser tool, on the victim machine. Once the data file was captured, we used Scapy, a packet manipulation tool, to extract the necessary features according to our final dataset requirements. These extracted features were then used in our machine learning models to predict the exact type of attack. This process allowed us to create a controlled and realistic dataset for our study.

Indian Institute of Information Technology, Dharwad 23

**CHAPTER 4**

This section discuss in detail about the cause of the experiments the results that are generated.

* 1. Experimental Analysis / Results
     1. Binary Classification

|  |  |
| --- | --- |
| Method | Accuracy score (%) |
| SVM | 97.54 |
| RNN | 89.1 |
| Random Forest | 99.947 |
| XGBoost | 99.95 |

Table 2(a): Comparative analysis of different classifiers for Binary Classification.

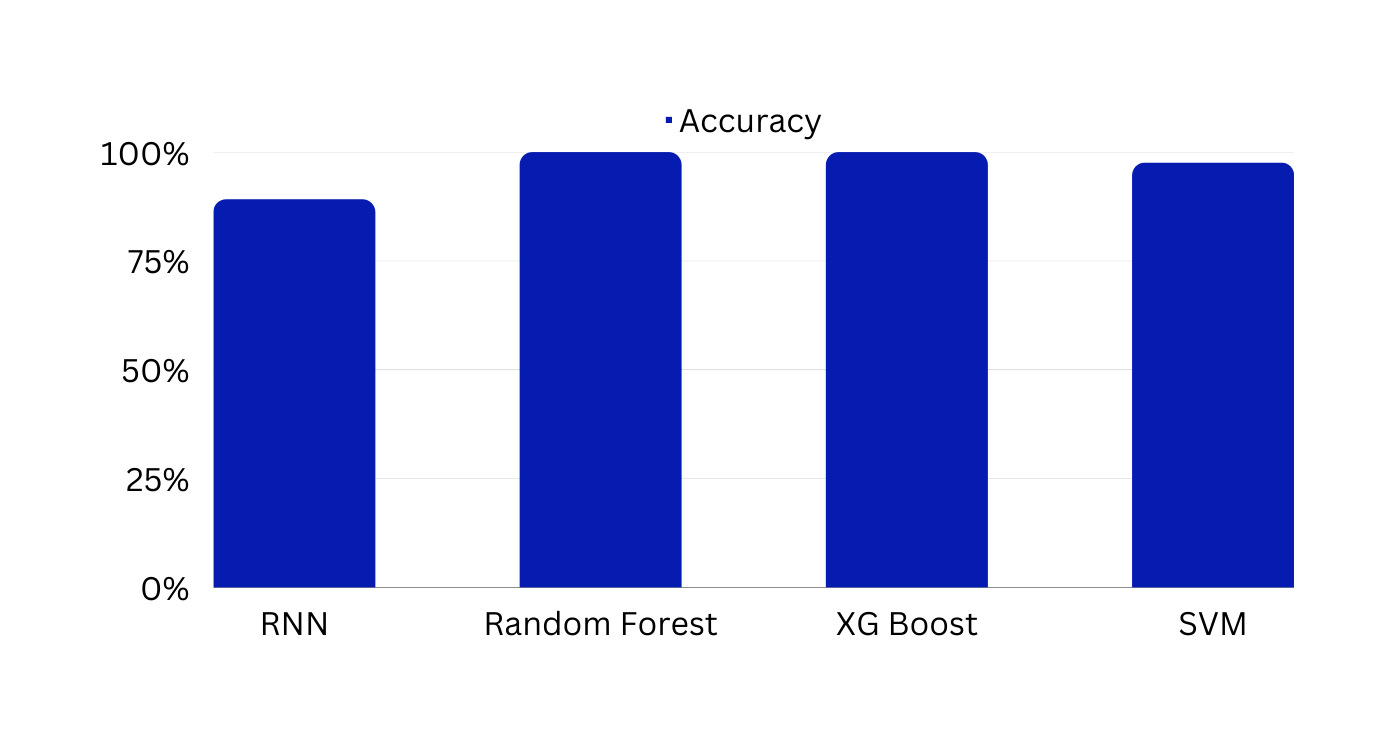


Fig: Performance Comparison of Multiple Classifiers for Binary Classification

Our study has successfully identified high accuracy rates in detecting web-based attacks using binary classification methods. The Random Forest and XGBoost models achieved the highest accuracies of 99.947% and 99.95%, respectively, indicating their robustness in identifying web-based attacks. The SVM model also performed well, achieving an accuracy rate of 97.54%. However, the RNN model has obtained a comparatively lower accuracy of 89.10%, indicating that it may not be the best option for detecting web-based attacks in comparison to the other ML models. This could be attributed to the fact that RNNs are more suitable for analysing sequential data and may not perform as well in detecting web-based attacks.

Indian Institute of Information Technology, Dharwad 24

* + 1. Multi-Class Classification

The Randomforest and XGBoost models have achieved the highest accuracies of 99.88% and 99.6%, respectively, indicating that these models are more robust in detecting web-based attacks. The KNN model has also achieved a high accuracy of 93.4%, making it a viable option for detecting web-based attacks. The random forest and XGBoost were also fine tuned on the hyperparameter set to get these results

On the other hand, the SVM model and Decision Tree model obtained comparatively lower accuracy scores of 93.4% and 80.5%, respectively. These results suggest that the Random Forest, XGBoost, and KNN models are more effective for this task in comparison to the SVM and Decision Tree models.

In conclusion, our results highlight the potential of Random Forest and XGBoost models in detecting web-based attacks in a multi-class classification setting, while the KNN model can also be used as a viable alternative. These findings may be useful in the development of more effective intrusion detection systems for web-based attacks. Additionally, our results suggest that DL methods, such as RNNs, may not be the optimal choice for detecting web-based attacks and that other ML models should be considered instead.

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Accuracy score (%) | F1 Score | Precision |
| KNN | 97.35 | 97.34 | 97.37 |
| Random Forest | 99.74 | 99.75 | 99.78 |
| SVM | 93.42 | 93.34 | 93.45 |
| XGBoost | 99.6 | 99.6 | 99.6 |
| Decision Tree | 80.56 | 81 | 80.1 |

Table 2(b): Comparative analysis of different classifiers for Multi-Class Classification.

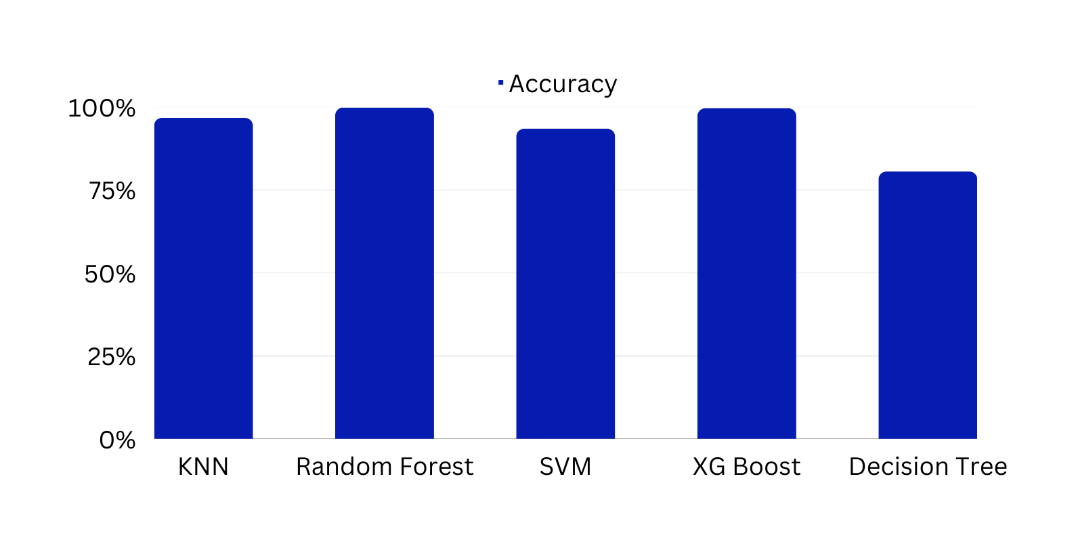
****

Fig: Performance Comparison of Multiple Classifiers for Multi-Class Classification

Indian Institute of Information Technology, Dharwad 25

**CHAPTER 5**

* 1. Conclusion

The study discussed in the statement proposes an ensemble-based approach to develop a more effective Intrusion Detection System (IDS) for detecting web-based attacks using AI methods. The research papers that we have been through helped us in understanding the various types of cyberattacks and a need for efficient threat detection system, particularly for web-based attacks.

The proposed framework is designed to address the shortcomings of traditional IDS that rely on rule-based or signature-based methods, which can be easily evaded by sophisticated attacks. The proposed ensemble-based approach uses multiple models trained on different subsets of the data to enhance the performance.

ML and DL are ideal in achieving continuous learning of known and unknown attacks. However, in order to create a more effective IDS model, more study has to be done on the huge dataset. This would involve collecting and labelling more data, as well as optimizing the algorithms and hyperparameters used in the models.

Overall, the proposed framework represents an important step towards developing more effective and reliable IDS for detecting web-based attacks using ML and DL techniques. The study highlights the potential of these approaches in improving the security of web-based systems and the need for continued research in this area.

Indian Institute of Information Technology, Dharwad 26

* 1. Contributions of this work

As previously mentioned, our approach to feature selection involved utilizing an Ensemble filter feature selection technique. This technique has been found to be effective in reducing the dataset size while maintaining the classification accuracy. We have integrated this feature selection technique in our simulator, and we have obtained a smaller dataset with a limited number of features. This smaller dataset is easier to handle and analyze, which has been helpful in our experiments.

Moreover, the creation of multi-class attack detection models is an essential contribution of our work. Traditional IDS models often focus on binary classification, but multi-class classification is more useful in real-world scenarios, where an attack can have multiple categories. We have used various machine learning and deep learning algorithms, to train our multi-class detection models. Our experiments have shown that our models can classify different types of attacks with high accuracy, which demonstrates the efficacy of our approach.

In addition, we have also contributed to the development of a balanced dataset from the CICIDS 2017 dataset. Imbalanced datasets are common in intrusion detection systems, where the number of benign traffic samples significantly exceeds the number of attack samples. To address this, we have created a balanced dataset with a 1:1 ratio of benign traffic and attacks, which has been helpful in training and testing our models.

Our project has also made a significant contribution in developing a real-world IDS by simulating and integrating various packages to capture and analyze network traffic, and predict the type of attack.

We have used Wireshark to capture the network traffic, which is a widely used tool for network traffic analysis. We have then analyzed the traffic using Scrapy, a web crawling framework in Python, to extract important features of interest. This has helped us to identify the key features for training our multi-class detection models.

Furthermore, we have integrated our Python code for the Ensemble filter feature selection technique and multi-class prediction model with the network analysis and feature extraction pipeline. This has enabled us to predict the type of attack in real-time, based on the features extracted from the network traffic. By integrating these various packages, we have developed a real-world IDS system that can be used for detecting web-based threats in real-time.

Overall, our project's development of a real-world IDS system using various packages and integration of our prediction models is a significant contribution to the field of intrusion detection systems. Our approach can be useful in developing more efficient and accurate IDS systems that can detect web-based threats in real-time, which is essential in the current cyber era.

Indian Institute of Information Technology, Dharwad 27

* 1. Future Scope

Intrusion Detection Systems (IDS) using Machine Learning (ML) and Deep Learning (DL) have already shown promising results in detecting various types of network attacks and anomalies. There are several areas where ML and DL can be further applied to improve IDS, such as:

* + Adversarial attacks: Adversarial attacks are a major challenge for IDS, as attackers can use them to evade detection by the system. ML and DL can be used to develop more robust IDS that can detect and counter adversarial attacks.
  + Real-time detection: IDS using ML and DL can be further improved to provide real-time detection of network attacks and anomalies. This can be achieved by optimizing the algorithms and the hardware used for training and inference.
  + Online learning: IDS using ML and DL can be designed to continuously learn from new data and update their models accordingly. This can improve the accuracy and effectiveness of the IDS over time, as the models are updated with new attack patterns and anomalies.
  + Multi-modal data: IDS using ML and DL can be extended to incorporate multiple types of data, such as network traffic, system logs, and user behavior. This can improve the accuracy of the system and enable it to detect more sophisticated attacks that span multiple modalities.
  + Explainability: ML and DL-based IDS can be further improved to provide more transparency and explainability to the decisions made by the system. This can help security analysts to better understand the reasoning behind the detections and take appropriate actions.

Overall, the future of IDS using ML and DL looks promising, with several avenues for research and development that can improve the effectiveness, efficiency, and usability of these systems in detecting and mitigating network attacks and anomalie.

Indian Institute of Information Technology, Dharwad 28

References

1. Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J. and Ahmad, F., 2021. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Transactions on Emerging Telecommunications Technologies, 32(1), p.e4150
2. Liu, H. and Lang, B., 2019. Machine learning and deep learning methods for intrusion detection systems: A survey. applied sciences, 9(20  ), p.4396.
3. Salloum, S.A., Alshurideh, M., Elnagar, A. and Shaalan, K., 2020. Machine learning and deep learning techniques for cybersecurity: a review. In Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020) (pp. 50-57). Springer International Publishing.
4. Thapa, N., Liu, Z., Kc, D.B., Gokaraju, B. and Roy, K., 2020. Comparison of machine learning and deep learning models for network intrusion detection systems. Future Internet, 12(10), p.167.
5. Maithem, Mohammed, and Ghadaa A. Al-sultany. "Network intrusion detection system using deep neural networks." Journal of Physics: Conference Series. Vol. 1804. No. 1. IOP Publishing, 2021.
6. Hamid, Yasir, M. Sugumaran, and V. R. Balasaraswathi. "Ids using machine learning-current state of art and future directions." British Journal of Applied Science & Technology 15.3 (2016).
7. Gautam, Rohit Kumar Singh, and Er Amit Doegar. "An ensemble approach for intrusion detection system using machine learning algorithms." 2018 8th International conference on cloud computing, data science & engineering (confluence). IEEE, 2018.

[8] Biswas, Saroj Kr. "Intrusion detection using machine learning: A comparison study." International Journal of pure and applied mathematics 118.19 (2018): 101-114.

1. Kumar, I., Mohd, N., Bhatt, C. and Sharma, S.K., 2020. Development of IDS using supervised machine learning. In Soft Computing: Theories and Applications: Proceedings of SoCTA 2019 (pp. 565-577). Springer Singapore.
2. Repalle, S.A. and Kolluru, V.R., 2017. Intrusion detection system using ai and machine learning algorithm. International Research Journal of Engineering and Technology (IRJET), 4(12), pp.1709-1715.
3. Ahanger, A.S., Khan, S.M. and Masoodi, F., 2021, April. An effective intrusion detection system using supervised machine learning techniques. In 2021 5th International Conference on Computing Methodologies and Communication (ICCMC) (pp. 1639-1644). IEEE.

Indian Institute of Information Technology, Dharwad 29

1. Pelletier, Z. and Abualkibash, M., 2020. Evaluating the CIC IDS-2017 dataset using machine learning methods and creating multiple predictive models in the statistical computing language R. Science, 5(2), pp.187-191.
2. Musa, Usman Shuaibu, et al. "A review on intrusion detection system using machine learning techniques." 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS). IEEE, 2021.
3. Ugochukwu, Chibuzor John, E. O. Bennett, and P. Harcourt. An intrusion detection system using machine learning algorithm. LAP LAMBERT Academic Publishing, 2019.
4. Alqahtani, H., Sarker, I.H., Kalim, A., Minhaz Hossain, S.M., Ikhlaq, S. and Hossain, S., 2020. Cyber intrusion detection using machine learning classification techniques. In Computing Science, Communication and Security: First International Conference, COMS2 2020, Gujarat, India, March 26–27, 2020, Revised Selected Papers 1 (pp. 121-131). Springer Singapore.
5. Zala, J., Panchal, A., Thakkar, A., Prajapati, B. and Puvar, P., 2020. Intrusion Detection System using Machine Learning. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 42(3), pp.61-71.
6. Chiba, Z., Abghour, N., Moussaid, K. and Rida, M., 2019. Intelligent approach to build a Deep Neural Network based IDS for cloud environment using combination of machine learning algorithms. computers & security, 86, pp.291-317

Indian Institute of Information Technology, Dharwad 30