INTRUSION DETECTION USING MACHINE LEARNING ALGORITHMS

by

Deepthi Hassan Lakshminarayana

December 2019

Director of Thesis: Dr. Nasseh Tabrizi

Major Department: Computer Science

With the growing rate of cyber-attacks, there is a significant need for intrusion detection systems (IDS) in networked environments. As intrusion tactics become more sophisticated and more challenging to detect, this necessitates improved intrusion detection technology to retain user trust and preserve network security. Over the last decade, several detection methodologies have been designed to provide users with reliability, privacy, and information security. The first half of this thesis surveys the literature on intrusion detection techniques based on machine learning, deep learning, and blockchain technology from 2009 to 2018. The survey identifies applications, drawbacks, and challenges of these three intrusion detection methodologies that identify threats in computer network environments.

The second half of this thesis proposes a new machine learning model for intrusion detection that employs random forest, naive Bayes, and decision tree algorithms. We evaluate its performance on a standard dataset of simulated network attacks used in the literature, NSL-KDD. We discuss pre-processing of the dataset and feature selection for training our hybrid model and report its performance using standard metrics such as accuracy, precision, recall, and f-measure.

In the final part of the thesis, we evaluate our intrusion model against the performance of existing machine learning models for intrusion detection reported in the literature.

Our model predicts the Denial of Service (DOS) attack using a random forest classifier with 99.81% accuracy, Probe attack with 97.89% accuracy, and R2L attack with 97.92% accuracy achieving equivalent or superior performance in comparison with the existing models.

INTRUSION DETECTION USING MACHINE LEARNING ALGORITHMS

A Thesis

Presented to The Faculty of the Department of Computer Science

East Carolina University

In Partial Fulfillment of the Requirements for the Degree

Master of Science in Computer Science

by

Deepthi Hassan Lakshminarayana

December 2019

INTRUSION DETECTION USING MACHINE LEARNING ALGORITHMS

by

Deepthi Hassan Lakshminarayana

APPROVED BY:

DIRECTOR OF THESIS: _____

Dr. Nasseh Tabrizi

COMMITTEE MEMBER: _____

Dr. Venkat Gudivada

COMMITTEE MEMBER: _____

Dr. Rui Wu

CHAIR OF THE DEPARTMENT

OF COMPUTER SCIENCE: _____ Dr. Venkat Gudivada

DEAN OF THE

GRADUATE SCHOOL: _____ Paul J. Gemperline, PhD

Table of Contents

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

There is an immense explosion of data everywhere, from keeping our word documents, excel worksheets to the data owned and operated by industries, banking/financial sectors, and many other places. It is essential to secure this data from malicious activities. With the growing rate of cyber-attacks, there is a vast requirement for effective intrusion detection systems (IDS) in the network. As the invasion gets complicated and challenging to detect, better techniques are employed to retain the trust and security in the network. Over the last decade, many methodologies have been designed to provide users with reliability, privacy, and information security.

In order to detect these intrusions, various intrusion detection systems (IDSs) are implemented in many networks (e.g., in banking and educational organizations). These systems are classified into host-based IDS, network-based (NIDS), and hybrid IDS. HIDS monitors the system and looks for malicious activities, and NIDS examines the traffic payload in the network for suspicious events. Based on detection methods, IDS are characterized into two types, namely signature-based IDS and anomaly-based IDS [1]. With the advances in technology, the cyber-attacks have reached new heights.

If the attack is not detected in time, it causes much damage to the network and

its users. In order to overcome this disaster, collaborative intrusion detection systems (CIDS) have been designed [1]. CIDS is designed to increase the detection abilities of the single IDS. In this case, each IDS communicates with other IDS to collaborate on data sharing and provide trust management [2].

A number of techniques are employed to prevent attacks and provide a secure network to the users. This research reviews some of those techniques used for intrusion detection, namely blockchain, machine learning, and deep learning technologies [3]. Blockchain technology was first implemented in 2009, which is the hidden innovation behind cryptocurrencies like Bitcoin. In contrast to physical money, advanced money, and digital forms of money accompany an undeniable issue called Double-Spending. In a blockchain, all the transactions are stored in blocks [4].

This paper further provides deeper insight into blockchain technology. Machine Learning is one of the widely popular approaches in intrusion detection. Anomaly in the network can be detected by running various machine learning algorithms such as K-Nearest Neighbor (KNN), Support Vector Machine (SVM), etc., [5].

One of the other approaches used to enhance the capabilities of machine learning algorithms are Deep Learning techniques [6]. It is proved that deep learning-based approaches address the challenges of IDS efficiently [7]. Some of the other techniques for intrusion detection are statistical methods, data mining methods [8], genetic algorithms, etc. [9].

The information presented in the thesis report is also presented in the paper, "A Survey of Intrusion Detection Techniques" (accepted by IEEE International Conference on Machine Learning Applications 542) [10].

1.1 Research Contribution

In this research, we have conducted a systematic literature survey on the current intrusion detection techniques. This review is done from a period starting from 1998 to 2018. We have provided a basic introduction to the intrusion detection systems, machine learning, deep learning, and blockchain technology. Then, a detailed review of applications for the above mentioned three techniques is described. Furthermore, its limitations/challenges in the area of the intrusion detection system are listed. We have reviewed the present state of the block chain technology in cyber-security in detecting attacks. A classification of machine learning and deep learning technologies used for detecting malicious user attacks in the network is provided. The approaches used by various researchers to identify any malicious activities in the NSL KDD data using tools are highlighted. We have also classified the publications of the papers for the past 19 years by its year of release and its database source.

Although blockchain technology indicates a promising contribution to intrusion detection, it neither provides the ways to compare its performance with machine learning algorithms nor the dataset to build an effective algorithm for intrusion detection. For the above reasons, we proposed the following approach.

We conduct an experiment in the final stage of research to detect the attacks in the dataset called NSL-KDD. A machine learning intrusion detection model is built using three classifiers to detect the attack in the dataset. The model's performance is measured and evaluated.

1.2 Thesis Structure

The thesis is organized in the following structure. Chapter 2 provides background on Intrusion detection systems, various attacks, and intrusion detection techniques such

as machine learning, deep learning and blockchain technology.

Chapter 3 gives the literature survey of intrusion detection techniques for the past 19 years, and Chapter 4 provides the related work on machine learning and the NSL-KDD dataset.

Chapter 5 describes the proposed methodology, including dataset description, algorithm description, and sequence of steps taken to conduct the research experiment. Chapter 6 shows the results obtained from the experiment. Finally, Chapter 7 provides the conclusion, we conclude the current research and discuss future research directions.

Chapter 2

Background

2.1 Intrusion Detection System

These systems are classified into host-based IDS, network-based (NIDS), and hybrid IDS (HIDS). HIDS monitors the system and looks for malicious activities, and NIDS examines the traffic payload in-network for suspicious events. This section provides a background of IDS, CIDS, and their challenges. Intrusion Detection is a way of monitoring the events happening in a framework such as a network or computers to detect any abnormal or malicious behavior which breaches the security or standard policies [11]. Fig 2.1 and Fig 2.2 shows how intrusion detectors are installed in every network; They provide a security layer and watch for any malicious activity. Following is the classification of IDS into HIDS, NIDS, CIDS, and some of its challenges.

2.1.1 HIDS and NIDS

Based on detection methods, IDS are characterized into two types, namely signature-based IDS and anomaly-based IDS [1]. IDS are mainly categorized into HIDS, NIDS, and Hybrid IDS, which is the integration of HIDS and NIDS to provide additional defense [9].

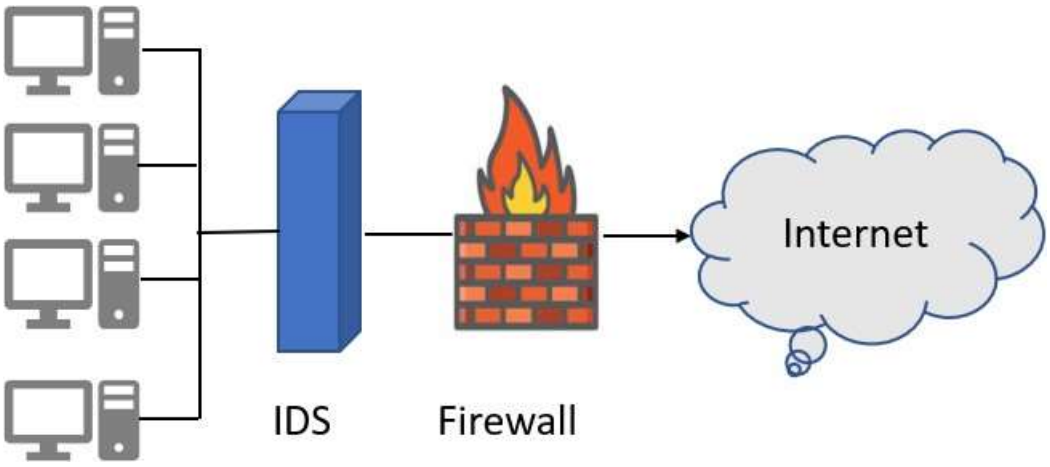

Figure 2.1: Intrusion detector in a network

Fig 2.2 shows the deployment of HIDS and NIDS in a network. In addition, an IDS is classified based on detection methods into signature-based, anomaly-based, and specification-based [1]. In the signature-based detection method, a stored signature is compared with the watched network system in order to detect the attack. In the anomaly-based detection method, any signs of malicious activity are determined, and an alarm is generated for such events. Specification-based detection method identifies any changes in a normal profile and watched events any changes in a normal profile and watched events and notifies the attack.

2.1.2 Collaborative Intrusion Detection System (CIDS)

CIDS increases the detection performance of single IDS, which can be easily surpassed in any complex attack like a denial-of-service (DOS) attack. If the attack is not

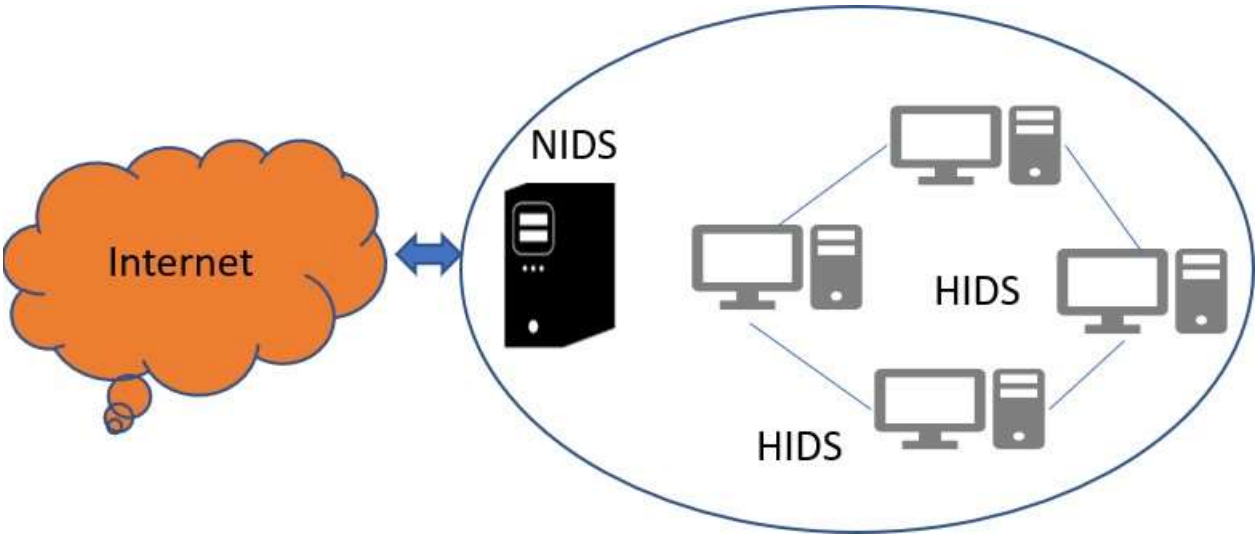

Figure 2.2: Deployment of HIDS and NIDS in a network

detected in time, it causes much damage to the network and its users. This disaster is overcome by collaborative intrusion detection systems (CIDS) [1]. CIDS is intended to increase the detection abilities of the single IDS. In this case, each IDS communicates with other IDS to collaborate on data sharing and provide trust management [2]. CIDS can be categorized into the following types;

- Hierarchical Collaboration System like Distributed intrusion detection system (DIDS) [12].
- Subscribe Collaboration System like (Distributed Overlay for Monitoring Internet Outbreaks) DOMINO [13].
- Peer-to-peer query-based collaboration system like an Internet-Scale Query Processor (PIER) [14].

Each of these techniques aims to detect any abnormal activity in the network.

2.1.3 Challenges

IDS solutions deal with many challenges like security and trust some of these are listed below;

- Wireless ad-hoc networks are dynamic, and hence, it is difficult to depend on a centralized server for analysis and correlation jobs [15].
- It is challenging to secure a mobile host physically because there are chances it could get captured and later join the network to get the information.

2.2 Blockchain Technology

Blockchain is a chain of blocks where each block holds the record of transactions. Blockhead contains the metadata and block body does the recording of transactions [16]. In contrast to physical money, advanced money, and digital forms of payment accompany an undeniable issue called double-spending, one of the significant issues addressed by the blockchain.

The fundamental usefulness given by a blockchain is a sequentially secure fashion for getting a block and data records. Blockchains are ordinarily shared and synchronized over a distributed system, and accordingly, are regularly utilized as a public ledger of transactions. Furthermore, each member in the blockchain system can see the record information dismiss or check it, dependent on the protocol. Once acknowledged, records are annexed to the blockchain in a subsequent request of their check [1].

2.2.1 Blockchain Architecture

A blockchain is an arranged chain of blocks. Blocks are holders of some related data and the information of transactions. A blockchain structure consists of three main

parts;

- a hash value: It provides the information of the previous block.
- The hash function and a timestamp: hash function, on the other hand, stores the blockchain data, and the timestamp holds the time when the blocks are created [17].
- A Merkel root: used to condense all transactions in the block and to check the presence and respectability of the transaction information rapidly. Body of the block stores every single reviewed transaction during the generation of the block. These blocks are linked with hash values to create a blockchain [16].

2.3 Machine Learning

Machine learning algorithms are extensively used in cyber-security to detect anomaly in the network, and it is proven to provide high detection rates [18]. These algorithms are applied to the training data to build the prediction model [19]. This prediction model is used to test against the given data for any malicious activities [20]. This type of learning is called supervised learning [21].

Some of these are support vector machine (SVM), naive Bayes classifier, decision table, and decision tree [22]. Supervised learning algorithms require the data to be class labeled, whereas the unsupervised learning algorithm does not require class labeled data [23]. Unsupervised algorithms include clustering, k-means, deep neural network, etc. [24]. Semi-supervised algorithms lie between the above mentioned two algorithms. They do not require all data to be class labeled [25]. Graph-based, self-training, and generative models are some of these examples of semi-supervised machine learning algorithms [26]. Table 2.3 shows the classification of machine learning techniques [27].

2.4 Deep Learning

Deep Learning originated from the Neural Network algorithm [27]. Different techniques are employed to tackle the drawback of one hidden layer in NN [28].
Deep learning has these immense number of techniques and learning strategies. The authors [29] [30] separates deep learning into three sub-gatherings, generative, discriminative, and hybrid [27] [31].

2.5 Blockchain Application and Limitation

In this section we discuss the applications and drawbacks of blockchain such as Cybersecurity [1] [4], Supply chain [32] [33], Smart Contract and Sidechain [34] [35], Cryptocurrency [36] [37] [38] etc. They are listed and explained as follows;

- Cybersecurity: Blockchain is combined with domain name service so that it gives domain owners protection from attacks by making it decentralized [39].
- Supply Chain: Blockchain bridges the transparency gap in the supply-chain between customers and buyers by showcasing its features such as public availability to track the goods from factories to consumers. The decentralized feature enables the participation of all parties in the supply chain [40].
- Smart Contract & Sidechain Technology: Smart contracts are the electronic contracts that enforce the agreements & transactions in blockchain ex. Ethereum supports the execution of code on the blockchain, and bitcoin has been supporting smart contracts for a long time now. Sidechain technology is software.

Table 2.3: Classification of the machine and deep learning methods

Machine Learning Methods			
	Supervised Learning	Unsupervised Learning	Semi-supervised Learning
Definition	Requires labelled dataset with pre-defined classes	Requires labelled dataset with out pre-defined classes	Lie in between they do not require all data to be class labelled
Method	Classification	Clustering	Graph Models
Example	SVM, Naive bayes classifier, descision table & trees	k-means,deep nueral network	Graph based,self training.

Deep Learning Methods			
	Generative (Unsupervised learning)	Discriminative (Supervised learning)	Hybrid
Definition	Also named generative architectures uses unlabeled data.	Helps to distinguish the data parts for pattern classification	Combines both generative & discriminative architectures
Aim	Pattern recognition in supervised learning	Mostly used for image recognition	To distinguish data as a discriminative approach
Example	Auto Encoder, SPN, RNN, Boltzmann Machine (BM)	Convolutional Neural Network (CNN)	Deep Neural Network

program in blockchain ledger, like a smart contract that allows the important information from one block to the other.

- Cryptocurrency is a Digital economy: Utilizes s cryptography to control the financial issuance and verifies the transactions. The principal digital currency, Bitcoin, made in 2009, is as yet the most broadly utilized cryptography money. Bitcoin enables engineers to include 40 bytes of subjective information to an exchange, which can be for all time recorded on the blockchain. Along these lines, the blockchain of Bitcoin has been utilized to enlist resource and proprietorship other than money-related transactions.
- Logistics: Blockchain-based applications in logistics are out there with companies like SmartLog using it successfully to obtain transparency across their supply chain network.
- IoT & Blockchain integration: Autopay feature in the car enables users to pay for fuel using smart contracts on the blockchain. Many other IoT applications include Iotcoin, Community currency, Enigma, International travel, etc.
- Voting: E-Voting is an issue with numerous challenges. Public verifiability and security are addressed by blockchain.

There are a wide variety of areas where blockchain is being applied, and some of the applications are listed in Fig.2.4.

Figure 2.4: Other applications of blockchain technology

- Others: Various other applications of blockchain include securing data for personal use to Business, Enterprises, and Shared Economy. Blockchain is also being extensively used in Predictive analysis, Digital identity, Copyright protection, and many others.

2.5.1 Challenges and Limitations of Blockchain technology in the intrusion detection system

In this section, we discuss some of the challenges in current blockchain technology in network intrusion detection [1] [4].

- Scalability & Speed: With the increase in the number of transactions, the blockchain

gets cumbersome. It is necessary that every node stores the transactions and check their validity, and the speed depends on the protocol utilized. Speed acts as a constraint to the scalability of blockchain. As the blockchains can process only seven transactions per second, it cannot satisfy the necessity of handling millions of transactions due to the small size of the block [4]. The scalability issue of blockchain is addressed in various ways such as;

a) Capacity Optimization of Blockchain: It is difficult for a node to work on the duplicate ledger; the author came up with an idea to remove the old transactions from a network, and all the addresses will be held by a database named account tree [41]. Another way to address this issue was proposed by Versum [42]. Versum enabled lightweight customers to redistribute costly calculations over huge inputs. It guarantees the calculation result is right through looking at results from various servers.

b) Redesigning Blockchain: New generation bitcoin called Bitcoin-NG was proposed in [43], whose primary thought was to decouple the traditional block into two sections called key block and micro block, which was used for leader election and storing the transactions respectively. When the key block is created, the node turns into the pioneer who oversees producing micro blocks. Bitcoin-NG likewise broadened the heaviest (longest) chain technique in which micro blocks convey no weight. Along these lines, blockchain is overhauled, and the reciprocation between block size and system security has been provided.

- Privacy Leakage: Blockchain can protect a specific measure of security through

the open key and private key. Clients execute with their private key and open key without presenting their identity. However, it appears that blockchain can not ensure the security in transactions since all the transactions are open for the public to view. Furthermore, the ongoing examination has demonstrated that a client's Bitcoin exchanges can be connected to uncover the client's data. Additionally, a strategy has been exhibited to interface the client's fictitious names to IP addresses, notwithstanding when clients are behind firewalls [4] [44]. Every customer can be exceptionally recognized by the number of the nodes it interfaces with. In any case, this set can be learned and used to discover the source of a transaction. Different techniques have been proposed to improve the obscurity of blockchain, which could be generally classified into two sorts, namely mixing and anonymous.

- Selfish Mining: Blockchain is prone to attacks by selfish miners. Selfish miners do not broadcast the mined blocks and get more revenue. In order to fix this issue, Heilman [45] displayed a novel methodology for legitimate mine workers to pick which branch to pursue. With arbitrary signals and timestamps, genuine excavators would choose all the newer squares. Be that as it may, [45] is helpless against forgeable timestamps. ZeroBlock expands on the basic plan: Each block should be produced and acknowledged by the system inside a greatest time interim. Inside ZeroBlock, selfish miners can't accomplish more than its normal reward.

- Cost and Energy: It requires a huge amount of energy to perform computations and to verify the transactions in any bitcoin mining for a single miner. And the

cost and energy increase as the network evolve [1].

- Privacy & Security: Attacks like DOS are very common on the blockchain platform. There is a major demand for privacy & security as the applications related to the blockchain involves smart contractions and transactions on the shared ledger.

- Complexity & delay: Blockchains are distributed; it takes time for transactions to complete, and users to update their ledgers. This delay can invite attackers.

- Adoption & Awareness: People lack the basic understanding of how the blockchain technology and how it can be adopted. This is causing major hindrance in its establishment.

- Size & Organization: Many firms and organizations prefer to develop their own blockchain system, given the significant size of distributed ledgers this deteriorates the performance of existing blockchain and make it less proficient.

- Management & Guidelines: Rules and regulations are frequently a long way behind the cutting-edge innovation. Because of the absence of normal principles for finishing transactions on a blockchain, Bitcoin blockchain has avoided existing guidelines for better proficiency. Be that as it may, blockchain applications are required to work inside guidelines.

Chapter 3

Literature Review

3.1 Research Methodology

The purpose of this study is to review the different techniques employed in intrusion detection. These papers conduct a thorough review of intrusion detection using various blockchain, machine learning, and deep learning techniques.

We started by researching on" Intrusion Detection Systems" and what are the ways the attacks can be predicted. There were 200+ papers published in the databases we have selected. We have reviewed over 56 papers, few books, and journal articles as our search was specific to the blockchain, machine, and deep learning technologies. The research was limited to the past 19 years (1999-2018), including books and journal publications.

The databases we chose are ACM Digital Library, Applied Science and Technology Full Text, computing (ProQuest), Gartner, IEEE Xplore, ProQuest Science, and SpringerLink using the search text" Intrusion detection blockchain, machine and deep learning."

3.1.1 Classification of papers by publication date

There have been numerous publications on intrusion detection. We have selected over 56 papers from 200+ papers for our review. There has been a huge increase in the publication in the year 2016 and later. In this survey, we present the total number of papers published from 1999-2018, as shown in Fig 3.1.

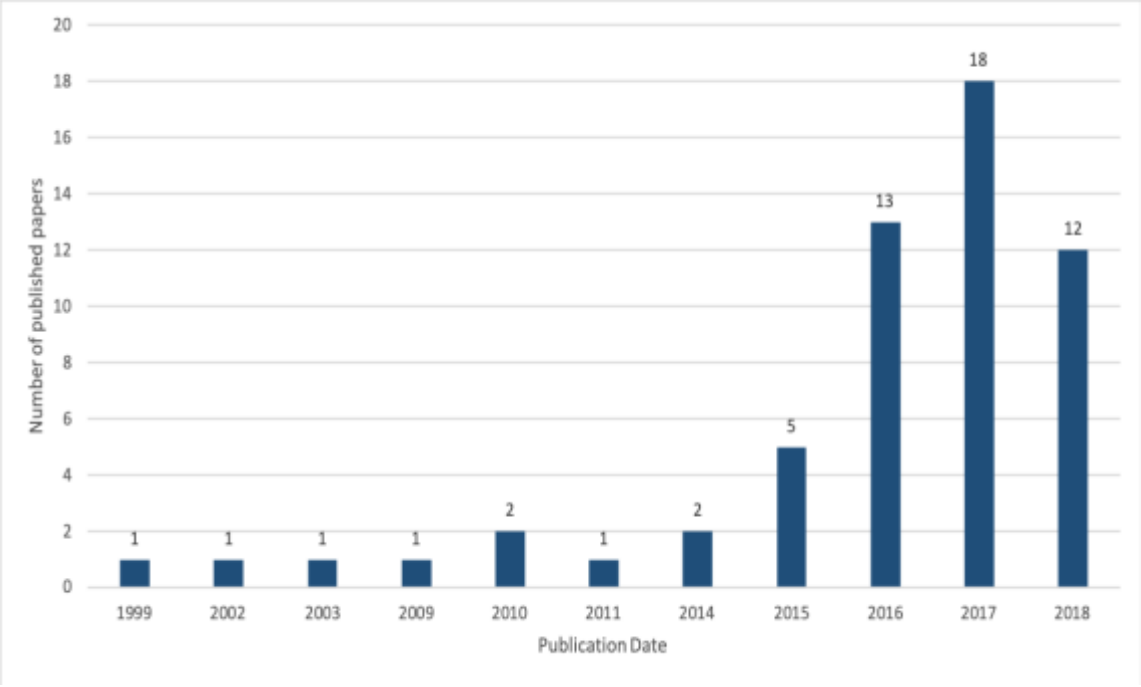

Figure 3.1: Papers published in blockchain, machine learning and deep learning for intrusion detection

3.1.2 Classification of papers by its Sources

The papers considered for this literature survey are from various databases such as ACM, SpringerLink, and others, as shown in Fig 3.2 It shows that most of the papers are published from IEEE Xplore.

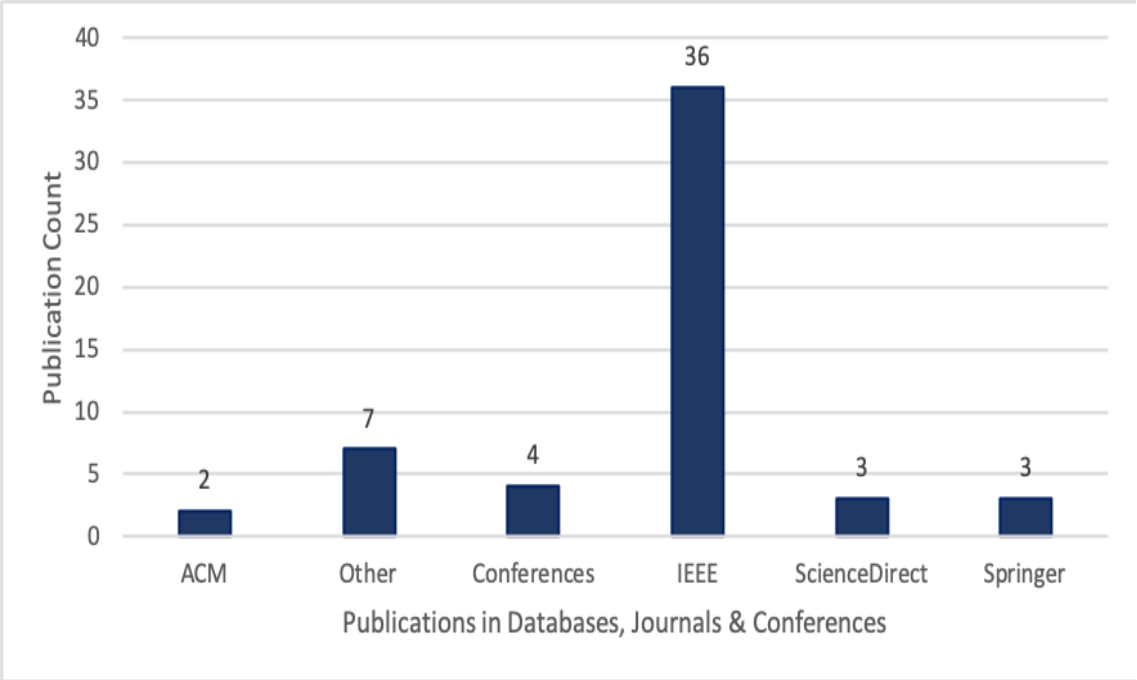

Figure 3.2: Papers published in various databases.

3.2 Systematic Survey

The goals of this survey are to review 56 related studies focusing on these techniques between the period 1998-2018 and provide the below key points-

- Review different hardware installations and software methodologies involved in blockchain technology for intrusion detection
- Review machine learning and deep learning approaches to intrusion detection.

This survey tries to find solutions to-

- What are the different machine learning and deep learning techniques employed to detect the threat in the network?

In this section, different challenges of the intrusion detection system are listed, and also various techniques employed by the authors to overcome these challenges are

discussed. Firstly blockchain-based solutions will be described. Followed by the contribution of different machine learning [46] and deep learning algorithms in protecting the network from malicious attacks will be provided.

3.2.1 Blockchain-Based Intrusion Detection

Blockchain-based CIDS: The author in [47] proposed the architecture of CIDS, which is based on blockchain and its distributed nature. In this model, the nodes communicate with each other on two layers called the Alert exchange layer and the Consensus layer. The peers in the CIDS network can collaborate with each other without disclosing confidential data. This model provides data privacy and integrity.

Provchain: The author in [48] designed and implemented a decentralized data provenance using blockchain to gather and confirm cloud information provenance, by inserting the provenance information into blockchain transactions. The Provchain provides reliability, user privacy, and security to the applications stored on the cloud.

Blockchain to protect Personal Data: The author [44] proposed an architecture that uses blockchain to secure the personal data by saving the file access permissions in the blockchain on a centralized cloud.

Block secure P2P cloud storage: Another author proposed a blockchain-based solution for the cloud data leak. This paper [49] demonstrates a new cloud storage architecture to provide more reliable and secure cloud storage. They customized the genetic algorithm and reduced the file loss rate.

3.2.2 Machine Learning Techniques for Intrusion Detection

2 tier Machine Learning Approach: In this paper, the author [50] proposed a 2-tier architecture for network intrusion detection using deep learning and machine learning algorithms. They performed simulations on KDD data set using the weka data mining tool and have proved that when the data goes through two classifiers, they increase system security.

Hybrid machine learning techniques: In this paper, the author [23] has presented a design and implementation to detect the attacks, which are known by supervised learning and unsupervised learning to detect the unknown attacks. The design consists of 7 hybrid models. The first layer consists of a supervised learning algorithm, and the second layer consists of an unsupervised algorithm. The goal of this approach is to increase the intrusion detection in networks.

3.2.3 Deep Learning Techniques for Intrusion Detection Techniques

Deep Learning approach: The author [7] proposed a deep learning approach for creating a network intrusion detection system (NIDS) [51] [52]. They used sparse auto-encoder and soft-max regression on the NSL-KDD dataset [53]. Anomaly detection accuracy is evaluated. Results show that they performed better than the previous anomaly detection [54].

Deep belief network (DBN): In this paper, the author [55] proposed a deep learning approach and built an intrusion detection system that uses DBN. This approach showed higher accuracy than the existing system using other training approaches like SVM,

DBN and SVM-DBN [56]. In this approach, DBN was used for data classification on the NSL-KDD dataset.

3.3 Chapter Conclusions

The latest review of intrusion detection techniques reveals the variety of intrusion detection techniques that have evolved in the past 20 years and its significant contribution to combating network attacks.

We have reviewed 57 papers from ACM Digital Library, IEEE Xplore, ScienceDirect, and SpringerLink databases. This research starts with a brief introduction to IDS, collaborative IDS, machine learning, deep learning, and blockchain technology.

A description of blockchain and its applications and limitations is reviewed, and a detail literature review on current intrusion detection techniques contributions is given. A review of the present state of the blockchain technology in cyber-security in detecting attacks is provided. Corresponding blockchain-based solutions are discussed. A classification of machine learning and deep learning technologies used for detecting malicious user attacks in the network is surveyed, and machine learning-based solutions are presented. A review of a machine and deep learning algorithms used by researchers to identify any malicious activities in the NSL KDD dataset using a variety of tools is provided. In conclusion, a distribution of all the papers by its year of publication and its databases is depicted in the graph.

Chapter 4

Related work on NSL KDD dataset

For the past 30 years, there has been immense progress in the area of intrusion detection systems. The research is growing with significant algorithms and methodologies to overcome the most advanced threats in the network. Previously researches had to create their own dataset for testing purposes. But now there are huge datasets available openly on the internet. One of the most widely used datasets for intrusion detection research is called the NSL KDD 99 cup.

The author [57] identified the inherent issues with the KDD 99 data, such as redundant records, difficulty in accessing the data due to biased results, and provided a new data set to overcome these challenges. The attacks in the data are divided into four categories [58] [59], as shown below in Fig 4.1. [60] conducted an experiment to detect attacks in the National Security Lab Knowledge Discovery and Data mining (NSL KDD) dataset [61] using various machine learning algorithms such as J48, SVM, and Naive Bayes. This was carried out using the Weka tool to predict the accuracy rate of normal and attack categories.

Similar research was conducted by another author [62] to detect anomaly in the dataset. He used Decision trees, neural networks, and nearest neighbor algorithms to propose an efficient hybrid model.

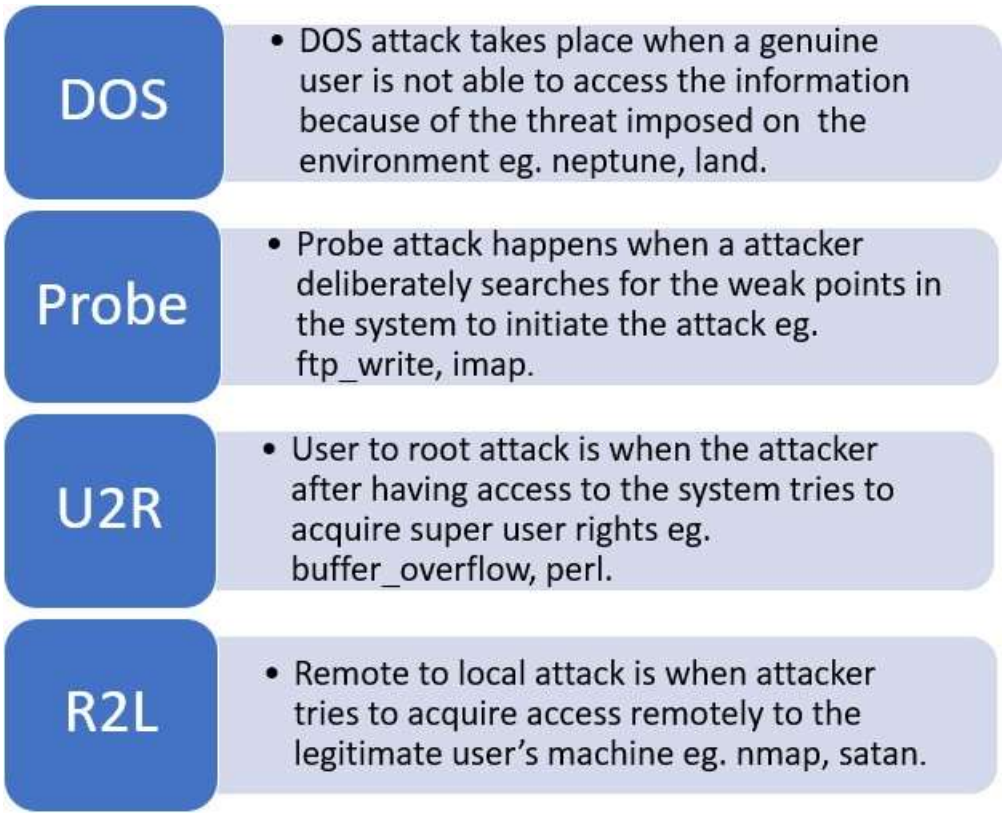

Figure 4.1: Classification of attacks in NSL KDD dataset

In this paper, the author [63] proposed an intelligent hybrid model using a combination of different classifiers to detect the attack on NSL KDD. The algorithms used are SVM, PCA, Random forest, and decision trees. He performed ten-fold cross-validation on 25192 training instances. The result yielded Random forest, when combined with other classifiers, has a precision rate of 99.9%.

[64] has developed an intrusion detection system with the Naive Bayes classifier. This model provides improved accuracy when compared to other models using the Naive Bayes classifier. A similar experiment was undertaken by [65] with a combination of different

classifiers such as decision trees, naive Bayes, RF, SVM to develop a discriminative multinomial naive Bayes model which performed better than the naive Bayes with an accuracy rate of 96.5% over 76.56%.

[66] proposed a detection framework using Random Forest (RF) classifier. He performed data mining to remove features that are not relevant and thus to lead improved accuracy. He trained the data using RF and detected the highest accuracy for the DOS attack by up to 99.67%.

The author [67] has proposed an ensemble classifier of Artificial neural network (ANN) and Bayesian net classifiers to detect the attacks in the latest NSL KDD and old KDD cup 99 datasets. There were numerous other researchers conducted to improve the detection rate of the attacks using machine learning and deep learning algorithms, which will be explained in the next section.

They have tested various classification techniques such as CART, ANN, Bayesian Net along with its ensemble methods. The experiment has resulted in the highest accuracy rate of 97.53% for ANN and Bayes Net for NSL KDD and 99.41% for KDD cup 99 for 70% partition of the data.

Chapter 5

Methodologies

In this section, Research Methodology is mentioned. From the literature review, we incorporate that there is an immense need in the development of an effective machine learning/ deep learning models for detecting the attack in the dataset.
One of the datasets used for intrusion detection related research projects is called the NSL-KDD dataset. An experiment is conducted to analyze the NSL-KDD dataset and train them using three machine learning algorithms such as Random Forest (RF), Naive Bayes(NB), and hybrid decision trees to detect the attacks.

5.1 Sequence of Steps

This section focuses on the steps taken to perform this research. We describe the sequence of actions taken to obtain the results. Fig 5.1 [68], shows the various steps taken to build this hybrid model. Below is the itemized explanation of each step in the flow diagram [66] [63] [69].

- First, the NSL-KDD dataset is loaded and pre-processed using numerical one-Hot-encoding, which is explained in detail in the next section.
- Then, the Feature selection of the data is made to eliminate redundant and irrelevant data.

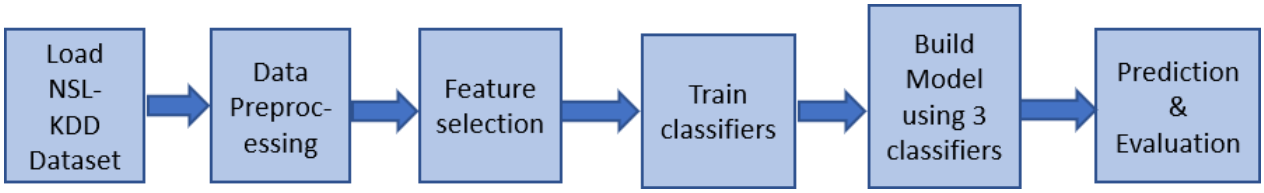

Figure 5.1: Sequence of actions for the Hybrid model

- Next, we train the classifiers for all the features and for the reduced features for later comparison.
- The complete Hybrid model is built using these steps by implementing all the three classifiers, such as Random forest, Hybrid decision trees, and Naive Bayes classifier.
- The model is evaluated by its performance in terms of Accuracy, Precision, F-measure, and Recall.

1. Input Dataset

The latest NSL-KDD dataset used in this research overcomes the limitation o NSL-KDD cup 99 datasets [70] [71] [72]. Pre-processing the data is a very important step in preparing the data to be fed into the algorithm. The dataset contains 42 features with the last column indicating whether it is normal and the name of the attack. It contains 80% train set and the remaining 20% test set. A sample view of the training and testing set is shown in Fig 5.2 and Fig 5.3 for better understanding. The dataset had the following issues which are addressed for our research;

- The test set has six missing categories, and we have updated that missing features from a train set.

	0	tcp	ftp_data	SF	491	0.1	0.2	0.3	0.4	0.5	...	25	0.17.1	0.03	0.17.2	0.24	0.25	0.26	0.05	0.27	normal
0	0	udp	other	SF	146	0	0	0	0	0	...	1	0.00	0.60	0.88	0.00	0.00	0.00	0.0	0.00	normal
1	0	tcp	private	S0	0	0	0	0	0	0	...	26	0.10	0.05	0.00	0.00	1.00	1.00	0.0	0.00	neptune
2	0	tcp	http	SF	232	8153	0	0	0	0	...	255	1.00	0.00	0.03	0.04	0.03	0.01	0.0	0.01	normal
3	0	tcp	http	SF	199	420	0	0	0	0	...	255	1.00	0.00	0.00	0.00	0.00	0.00	0.0	0.00	normal
4	0	tcp	private	REJ	0	0	0	0	0	0	...	19	0.07	0.07	0.00	0.00	0.00	0.00	1.0	1.00	neptune

5 rows × 42 columns

Figure 5.2: Sample view of train dataset

	0	tcp	private	REJ	0.1	0.2	0.3	0.4	0.5	0.6	...	10.1	0.04.1	0.06.1	0.22	0.23	0.24	0.25	1.2	1.3	neptune
0	0	tcp	private	REJ	0	0	0	0	0	0	...	1	0.00	0.06	0.00	0.00	0.00	0.0	1.00	1.00	neptune
1	2	tcp	ftp_data	SF	12983	0	0	0	0	0	...	86	0.61	0.04	0.61	0.02	0.00	0.0	0.00	0.00	normal
2	0	icmp	eco_i	SF	20	0	0	0	0	0	...	57	1.00	0.00	1.00	0.28	0.00	0.0	0.00	0.00	saint
3	1	tcp	telnet	RSTO	0	15	0	0	0	0	...	86	0.31	0.17	0.03	0.02	0.00	0.0	0.83	0.71	mscan
4	0	tcp	http	SF	267	14515	0	0	0	0	...	255	1.00	0.00	0.01	0.03	0.01	0.0	0.00	0.00	normal

5 rows × 42 columns

Figure 5.3: Sample view of the test dataset

- The data column names were missing, and this was fixed by referring to an NSL -KDD pre-processing paper [71] [73] [57]. We created a list of columns names and assigned them to each column, with the 42nd column being the category of the attack.

2. Data Pre-processing and Feature Selection

All the data features undergo a data transformation process in order to input it into the algorithm. Data re-modeling is a significant step in this research. Following is the description of how it is being done.

(a) Data Re-modelling

The NSL-KDD dataset consists of 12,5972 data records in the training set and 22,543 data records in the test set. The data has categorical and numerical

values, as shown in Fig 3. It does not have any null or missing values, which made it easy and consistent for detecting the accuracy of the attack. The dataset is treated with various techniques to make it suitable for classification [72] [60]. Following machine learning approaches are taken during the data mining process;

i. One-Hot-encoding(one-of-k): The features are made numerical using one-Hot-encoding [74]. It is used to transform all categorical features into binary features. The input to this must be a matrix of integers will be Sparx matrix with each column holding value of one feature.

ii. Label-encoding: The features are transformed from categorical to numerical using a label encoding approach in machine learning using python [75]. The features are first transformed with Label-encoder from category to a number.

iii. Split Dataset: The dataset is then split into four datasets for every attacking category. Then, we rename every attack label as 0 for normal, 1 for DOS, 2 for Probe, 3 for R2L, and 4 for U2R.

iv. The features are scaled to avoid features with large values that may weigh too much in the results [72].

(b) Feature Selection

Recursive feature elimination (RFE) is performed to bring in the important features out 41 features in the dataset. In this research, we eliminate redundant and irrelevant data by selecting a subset of relevant features that fully represents the given problem. Feature selection with ANOVA F-test. This analyzes each feature individually to determine the strength of the relationship between the feature and labels. Using Second Percentile

method (sklearn. feature selection) to select features based on percentile of the highest scores. When this subset is found: Recursive Feature Elimination (RFE) is applied.

The authors in [73] [59] [60] express that" In the wake of getting the satisfactory number of highlights during the univariate choice procedure, an RFE was worked with the number of highlights went as a parameter to recognize the highlights chose." This either suggests RFE is utilized for getting the highlights recently chose yet additionally acquiring the position. This utilization of RFE is anyway repetitive as the highlights chose can be gotten in another manner. One can likewise not say that the highlights were chosen by RFE, as it was not utilized for this. The statement could anyway additionally suggest that solitary the number 13 from univariate highlight determination was utilized. RFE is then utilized for highlight determination attempting to locate the best 13 highlights. With this utilization of RFE, one can really say that it was utilized for highlight determination.

We will be proceeding with the data mining; the subsequent choice is considered as this uses RFE. Starting now and into the foreseeable future, 13 features for every attacking category will be considered in this research. We have not faced any over-fitting or under-fitting problems while conducting this experiment.

3. Classifiers used for the Model

We have selected three classifiers for building this model. The reason behind selecting this combination of algorithms in the Hybrid model is to incorporate both machine learning, deep learning methodologies supervised and

unsupervised learning techniques on the data. The experiment is conducted to measure and validate the performances of these classifiers on the NSL-KDD dataset [62] [71]. The classifiers chosen to build this hybrid machine learning models are described as follows:

(a) Random Forest Classifier

RF is also called as ensemble classifier because it is a package of developed on sample training data. It provides very good accuracy in finding faults. It is known to be a supervised machine learning algorithm [68]. The applications of RF are, it builds multiple decision trees; the trees generated can be used in the future [66].

(b) Hybrid Decisions Tree

These can be used as supervised or unsupervised learning. One of the reasons for choosing decision trees is that they can handle both categorical and numerical data well. These trees are constructed and operated by the recursive partitioning principle. Issues with over-fitting can be addressed by pruning techniques [63].

(c) Naive Bayes Classifier

Is a supervised learning technique which functions on independence assumption in which probability of attributes are independent of the other [64] [76]. NB classifiers provide correct results almost always, and error could be because of noisy data or other data variance. Researches indicate that NB provides accurate results.

4. Performance Metrics

The hybrid machine learning model is tested similar to other machine learning models by measuring its classification accuracy, precision, recall, f-measure and

by taking into account, it's error rate, true positive, false positive [77]. We have used the following parameters and metrics [73], and the confusion matrix is utilized for showing the classification [62] [77].

A confusion matrix is a visual representation of how an algorithm performs. After the classifier is trained, it is applied to test the data. The predicted probabilities are viewed with the help of a confusion matrix. The accuracy, Precision, Recall, and f-measure are analyzed with the below equations. And is depicted with respect to every attack category.

(a) True Positive

The instances which is positive and are correctly predicted as positive by the classifier.

(b) False Positive

The instances considered are negative but are miss-classified as positive by the classifier.

(c) True Negative

The instances considered are negative and are classified correctly as negative.

(d) False Negative

The instances considered are positive, but the miss-classified as negative by the classifier.

(e) Accuracy

The average accuracy rate is calculated by this formula given in equation 5.1.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{5.1}$$

(f) Precision

Is the measure of of exactness [77] 5.2.

$$\text{Precision} = \frac{TP}{TP+FP} \tag{5.2}$$

(g) Recall

Measures what percent of positive instances are positive [77]5.3.

$$\text{recall} = \frac{TP}{TP+FN} \tag{5.3}$$

(h) F-measure: Harmonic mean of precision and recall in order to measure the accuracy of a [77] test5.4

$$\text{F-measure} = \frac{2*\text{precision}*\text{recall}}{\text{precision}+\text{recall}} \tag{5.4}$$

Chapter 6

Results

6.1 Experiment Details

For this study, 30 papers are thoroughly researched, and the experiment is conducted using a windows HP dual-core personal computer with 8 GB RAM. The environment on which the application is built is called Jupyter, which is a workbench provided by an Anaconda platform. The programming language used for developing this intrusion model is python.

The dataset considered for this research is called NSL-KDD, which has an 80% train set and a 20% test set that is 125,972 records in training set and 22,543 records in the test set. In order to make intelligent decisions with the data, we considered various intelligent decision algorithms such as the Random Forest classifier, Naive Bayes classifier, and Decision trees to build this intrusion detection model. The performance of these classifiers tested on 13 features of the NSL-KDD dataset for DOS, Probe, U2R, and R2L attack types shown in the table below 6.1.

From Table 6.2, 6.3, and 6.4, it is evident that our model, when compared with the performances of other models, provided the highest accuracy for the DOS attack with 99.81% for RF classifier.

Test conducted with 13 features					
Classifiers Used	**Attack-Type**	**Accuracy**	**Precision**	**Recall**	**F-measure**
Random Forest Ensemble Classifier Normal- 99.7	DOS	99.81	99.91	99.65	99.79
	Probe	99.63	99.61	99.16	99.39
	R2L	98.03	97.50	96.26	97.20
	U2R	99.70	94.88	82.82	87.11
Naïve Bayes Model Normal- 71.4	DOS	86.73	98.82	70.30	82.14
	Probe	97.89	97.32	96.05	96.65
	R2L	93.56	89.09	95.08	91.62
	U2R	97.25	60.13	97.91	66.06
Hybrid Decision Trees Normal- 85.7	DOS	99.63	99.50	99.66	99.58
	Probe	99.57	99.39	99.26	99.32
	R2L	97.92	97.15	96.95	97.05
	U2R	99.66	86.48	91.67	88.62

Figure 6.1: Accuracy predicted for all attack types with 13 features for three classifiers.

Similarly, for the Naive Bayes classifier, our model with 13 features provided an increased accuracy for DOS, Probe, U2R, and R2L attacks when compared to other NB model with 15 features. When compared with other decision trees, our model gave increased accuracy for DOS, Probe, and U2R.

6.2 Comparison of Intrusion models

We are comparing the performance of each Rf, NB, and DT classifiers used to build our model with the performance existing intrusion model for the same dataset. Given below is the detail comparision:

1. Comparing RF classifiers

Accuracy of Random Forest Classification Algorithm				
Attack Class	Our Hybrid Model with 13 features	Another Model with 41 features & 15 features		Another model with 41 features
DOS	99.81	98.7	99.1	99.67
Probe	99.63	97.6	98.9	99.67
U2R	99.70	97.5	98.7	99.67
R2L	98.03	96.8	97.9	99.67

Figure 6.2: Comparing the results of our model with the performances of other Random Forest classifiers

In this paper [66] and [59], the RF algorithm yields very close accuracy of 98.7 % for DOS, 97.6% for Probe, 97.5% for U2R, and 96.8% for R2L compared to our model which is as shown below in Fig 6.2.

We can see that our model provided higher accuracy for DOS, U2R, and R2L attack compared to other models.

2. Comparing the results of Naive Bayes classifier

In this paper [76], the accuracy DOS, Probe, U2R, and R2L for the dataset with 41 features and 15 features are obtained as 75.0%, 75.1%, 74.3%, and 71.1% which is less compared to the results obtained in this paper [59].

However, our model outperforms the first model by providing increased accuracy for DOS as 86.73%, Probe at 97.89%, U2R as 97.25%, and R2L as 93.56% as shown in 6.3.

3. Comparing the results with other Hybrid Decision Trees

We compared the accuracy of our model with the performances of the other two papers, [60] and [62]. The results are shown in table 6.4. We can see that

Performance of Naïve Bayes Classification Algorithm									
Attack Class	Our Hybrid Model with 13 features				Another Model Accuracy of		Another model with 41 features		
	Accur	Preci	Recall	F-m	41 & 15 features		Preci	Recall	F-m
DOS	86.73	98.82	70.30	82.14	72.7	75.0	96.3	95.3	95.8
Probe	97.89	97.32	96.05	96.65	70.9	75.1	59.4	89.8	71.5
U2R	97.25	60.13	97.91	66.06	70.7	74.3	0	0.5	0.006
R2L	93.56	89.09	95.08	91.62	69.8	71.1	26.5	82.9	40.1

Figure 6.3: Comparing the results of our model with the performances of other Naive Bayes classifiers

Accuracy of Decision Trees			
Attack Class	Our Hybrid Model with 13 features	Another Model with 6 features	Another model with 41 features
DOS	99.63	99.1	99.1
Probe	99.57	98.9	98.9
U2R	99.66	98.7	98.7
R2L	97.92	97.9	97.9

Figure 6.4: Comparing the results of our model with the performances of other Hybrid decision trees

Our model gave an accuracy of 99.6% for DOS, 99.57% for Probe, 99.66% for U2R, and 97.92% for R2L. Moreover, it shows that our model performed better for DOS, Probe, and U2R with almost the same accuracy for R2L.

4. Recursive Feature Elimination Cross-validation (RFECV)

Recursive feature elimination (RFE) is a feature determination technique that fits a model and expels the weakest feature until the predefined number of highlights is reached. Features are positioned, and by recursively killing few features for every recursion, RFE endeavors to dispense with conditions and collinearity that may exist in the model. The experiment reveals that RF has the highest accuracy compared to other algorithms, followed by hybrid decision trees. We have plotted the number of features versus the number of cross-validation scores for illustration. Following is the RF RFECV plot for all four attack types shown in Fig. 6.5 6.6 6.7 and 6.8.

We can see that in Fig 6.5, the curve provides excellent accuracy as the number features increases. As we are selecting 13 features each of 122 and then getting 13 best features from 122 from RFE, the curve captures all the informative features are added in the model and remains with almost the same accuracy of 99.81% and F-measure of 99.7%. It indicates that the model performs best for all 13 to 122 features even there are slight variations in the curve for the DOS attack.

The same follows for Fig 6.6 and Fig 6.8 with slight variation in accuracy of Probe as 99.63%, R2L as 98.03%. However, the accuracy of a number of features captured for U2R gradually increases in 8 features are added into the model and then decreases in accuracy as non-informative features are added. This process takes place recursively until all the features have been added to the model.

RF has the highest accuracy as compared to other classifiers.

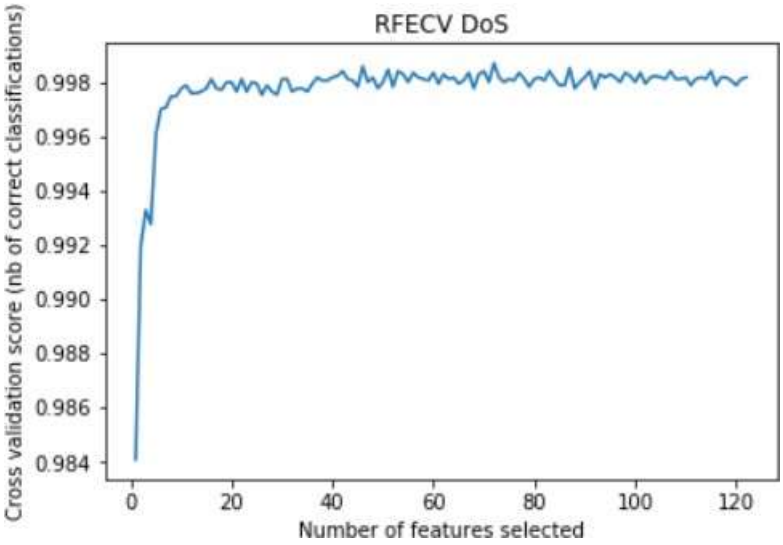

Figure 6.5: Random Forest RFECV DOS

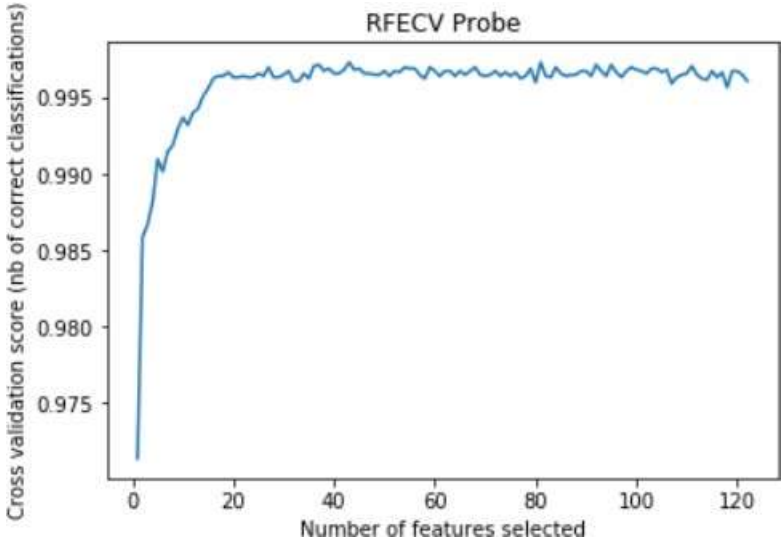

Figure 6.6: Random Forest RFECV Probe

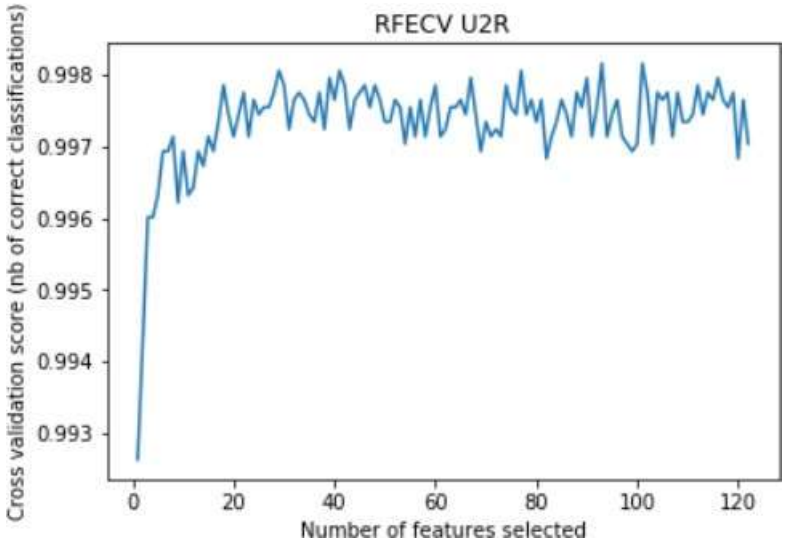

Figure 6.7: Random Forest RFECV U2R

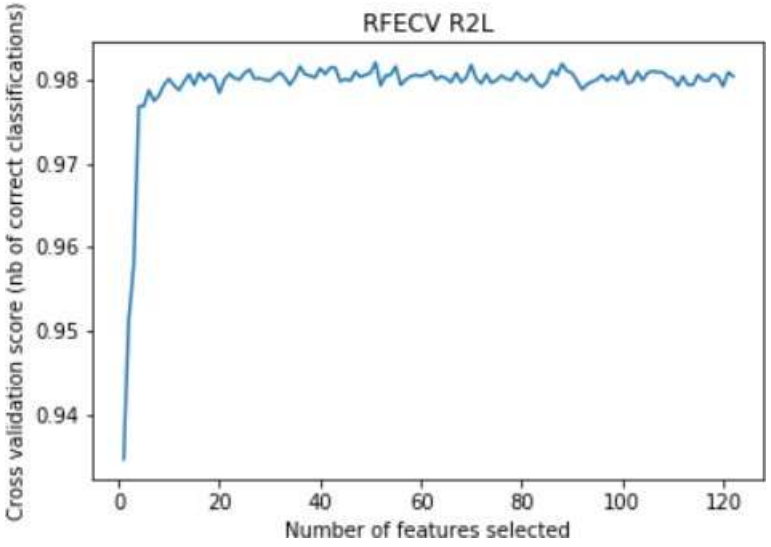

Figure 6.8: Random Forest RFECV R2L

Chapter 7

Future Work and Conclusion

Due to the enormous growth in cyber-attacks, there is a requirement of an effective intrusion detection system to protect the data and the network.

This thesis is an effort towards the development of a significant intrusion model. Firstly, we have reviewed over 56 papers on intrusion detection techniques from 2009 to 2019. We found that some of the detection techniques, such as machine learning, deep learning, and blockchain technology, play a vital role in constructing these life-saving systems. A literature review provides background on these techniques with their applications and limitations in the area of intrusion detection.

Our next step is that we researched over 30 papers and proposed an intrusion detection model using three classifiers and compared their performances with existing models with the NSL-KDD dataset.

In this work, we have considered three classifiers, namely Random Forest, Naive Bayes, and decision trees. In future work, we suggest using algorithms like K-Nearest neighbor and other deep learning algorithms to achieve good classification accuracy for a greater number of features in the dataset. Furthermore, popular data mining techniques such as deep neural networks and dbscan can be used to improve the results in the future.

BIBLIOGRAPHY

[1] Weizhi Meng, Elmar Wolfgang Tischhauser, Qingju Wang, Yu Wang, and Jinguang Han. When intrusion detection meets blockchain technology: a review. *Ieee Access*, 6:10179–10188, 2018.

[2] Nikolaos Alexopoulos, Emmanouil Vasilomanolakis, Natalia Reka Ivanko, and Max Muhlhauser. Towards blockchain-based collaborative intrusion detection systems. In Gregorio D'Agostino and Antonio Scala, editors, *Critical Information Infrastructures Security*, pages 107–118, Cham, 2018. Springer International Publishing.

[3] Yang Xin, Lingshuang Kong, Zhi Liu, Yuling Chen, Yanmiao Li, Hongliang Zhu, Mingcheng Gao, Haixia Hou, and Chunhua Wang. Machine learning and deep learning methods for cybersecurity. *IEEE Access*, 6:35365–35381, 2018.

[4] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang. An overview of blockchain technology: Architecture, consensus, and future trends. In *Big Data (BigData Congress), 2017 IEEE International Congress on*, pages 557–564. IEEE, 2017.

[5] Chih-Fong Tsai, Yu-Feng Hsu, Chia-Ying Lin, and Wei-Yang Lin. Intrusion detection by machine learning: A review. *Expert Systems with Applications*, 36(10):11994–12000, 2009.

[6] Bo Dong and Xue Wang. Comparison deep learning method to traditional methods using for network intrusion detection. In *2016 8th IEEE International Conference on Communication Software and Networks (ICCSN)*, pages 581–585. IEEE, 2016.

[7] Ahmad Javaid, Quamar Niyaz, Weiqing Sun, and Mansoor Alam. A deep learning approach for network intrusion detection system. In *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (Formerly BIONETICS)*, BICT'15, pages 21–26, ICST, Brussels, Belgium, Belgium, 2016. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[8] Zheng Wang. Deep learning-based intrusion detection with adversaries. *IEEE Access*, 6:38367–38384, 2018.

[9] Uzair Bashir and Manzoor Chachoo. Intrusion detection and prevention system: Challenges & opportunities. In *Computing for Sustainable Global Development (INDIACom), 2014 International Conference on*, pages 806–809. IEEE, 2014.

[10] D.H. Lakshminarayana J. P. Philips, N. Tabrizi. A survey of intrusion detection techniques. *IEEE International Conference on Machine Learning Application*, 175(542):7–9, 2019.

[11] Karen Scarfone and Peter Mell. Guide to intrusion detection and prevention systems (idps). *NIST special publication*, 800(2007):94, 2007.

[12] Steven R Snapp, James Brentano, Gihan V Dias, Terrance L Goan, L Todd Heberlein, Che-Lin Ho, Karl N Levitt, Biswanath Mukherjee, Stephen E Smaha, Tim Grance, et al. Dids (distributed intrusion detection system)-motivation, architecture, and an early prototype. In *Proceedings of the 14th national computer security conference*, volume 1, pages 167–176. Washington, DC, 1991.

[13] Vinod Yegneswaran, Paul Barford, and Somesh Jha. Global intrusion detection in the domino overlay system. In *NDSS*, 2004.

[14] Ryan Huebsch, Brent Chun, Joseph M Hellerstein, Boon Thau Loo, Petros Maniatis, Timothy Roscoe, Scott Shenker, Ion Stoica, and Aydan R Yumerefendi. The architecture of pier: an internet-scale query processor. 2005.

[15] Paul Brutch and Calvin Ko. Challenges in intrusion detection for wireless ad-hoc networks. In *null*, page 368. IEEE, 2003.

[16] Shiyong Yin, Jinsong Bao, Yiming Zhang, and Xiaodi Huang. M2m security technology of cps based on blockchains. *Symmetry*, 9(9):193, 2017.

[17] Marko Vukolić. Rethinking permissioned blockchains. In *Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts*, pages 3–7. ACM, 2017.

[18] Hadi Sarvari and Mohammad Mehdi Keikha. Improving the accuracy of intrusion detection systems by using the combination of machine learning approaches. In *2010 international conference of soft computing and pattern recognition*, pages 334–337. IEEE, 2010.

[19] Phurivit Sangkatsanee, Naruemon Wattanapongsakorn, and Chalermpol Charnsripinyo. Practical real-time intrusion detection using machine learning approaches. *Computer Communications*, 34(18):2227–2235, 2011.

[20]T Poongothai and K Duraiswamy. Intrusion detection in mobile adhoc networks using machine learning approach. In *International Conference on Information Communication and Embedded Systems (ICICES2014)*, pages 1–5. IEEE, 2014.

[21] David Endler. Intrusion detection. applying machine learning to solaris audit data. In *Proceedings 14th Annual Computer Security Applications Conference (Cat. No. 98EX217)*, pages 268–279. IEEE, 1998.

[22] Arkadiusz Warzyński and Grzegorz Kołaczek. Intrusion detection systems vulnerability on adversarial examples. In *2018 Innovations in Intelligent Systems and Applications (INISTA)*, pages 1–4. IEEE, 2018.

[23] Deyban Perez, Miguel A Astor, David Perez Abreu, and Eugenio Scalise. Intrusion detection in computer networks using hybrid machine learning techniques. In *2017 XLIII Latin American Computer Conference (CLEI)*, pages 1–10. IEEE, 2017.

[24] Bisyron Wahyudi, Kalamullah Ramli, and Hendri Murfi. Implementation and analysis of combined machine learning method for intrusion detection system. *International Journal of Communication Networks and Information Security*, 10(2):295–304, 2018.

[25] MA Jabbar, Rajanikanth Aluvalu, and S Sai Satyanarayana Reddy. Intrusion detection system using bayesian network and feature subset selection. In *2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pages 1–5. IEEE, 2017.

[26] Tahir Mehmood and Helmi B Md Rais. Machine learning algorithms in context of intrusion detection. In *2016 3rd International Conference on Computer and Information Sciences (ICCOINS)*, pages 369–373. IEEE, 2016.

[27] Kwangjo Kim and Muhamad Erza Aminanto. Deep learning in intrusion detection perspective: Overview and further challenges. In *2017 International Workshop on Big Data and Information Security (IWBIS)*, pages 5–10. IEEE, 2017.

[28] Sidharth Behera, Ayush Pradhan, and Ratnakar Dash. Deep neural network architecture for anomaly based intrusion detection system. In *2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)*, pages 270–274. IEEE, 2018.

[29] Chuanlong Yin, Yuefei Zhu, Jinlong Fei, and Xinzheng He. A deep learning approach for intrusion detection using recurrent neural networks. *Ieee Access*, 5:21954–21961, 2017.

[30] Li Deng. A tutorial survey of architectures, algorithms, and applications for deep learning. *APSIPA Transactions on Signal and Information Processing*, 3, 2014.

[31] Tuan A Tang, Lotfi Mhamdi, Des McLernon, Syed Ali Raza Zaidi, and Mounir Ghogho. Deep learning approach for network intrusion detection in software defined networking. In *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, pages 258–263. IEEE, 2016.

[32] Nikola Bozic, Guy Pujolle, and Stefano Secci. A tutorial on blockchain and applications to secure network control-planes. In *Smart Cloud Networks & Systems (SCNS)*, pages 1–8. IEEE, 2016.

[33] Marc Pilkington. 11 blockchain technology: principles and applications. *Research handbook on digital transformations*, 225, 2016.

[34] Zibin Zheng, Shaoan Xie, Hong-Ning Dai, and Huaimin Wang. Blockchain challenges and opportunities: A survey. *Work Pap.–2016*, 2016.

[35] Karl Wust and Arthur Gervais. Do you need a blockchain? In *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*, pages 45–54. IEEE, 2018.

[36] Suveen Angraal, Harlan M Krumholz, and Wade L Schulz. Blockchain technology: applications in health care. *Circulation: Cardiovascular Quality and Outcomes*, 10(9):e003800, 2017.

[37] Janusz J Sikorski, Joy Haughton, and Markus Kraft. Blockchain technology in the chemical industry: Machine-to-machine electricity market. *Applied Energy*, 195:234–246, 2017.

[38] Steve Huckle, Rituparna Bhattacharya, Martin White, and Natalia Beloff. Internet of things, blockchain and shared economy applications. *Procedia computer science*, 98:461–466, 2016.

[39] Fangfang Dai, Yue Shi, Nan Meng, Liang Wei, and Zhiguo Ye. From bitcoin to cybersecurity: A comparative study of blockchain application and security issues. In *2017 4th International Conference on Systems and Informatics (ICSAI)*, pages 975–979. IEEE, 2017.

[40] Krystsina Sadouskaya et al. Adoption of blockchain technologyin supply chain and logistics. 2017.

[41] BF Franca. Homomorphic mini-blockchain scheme, 2015.

[42] Jelle van den Hooff, M Frans Kaashoek, and Nickolai Zeldovich. Versum: Verifiable computations over large public logs. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 1304–1316. ACM, 2014.

[43] Ittay Eyal, Adem Efe Gencer, Emin Gun Sirer, and Robbert Van Renesse. Bitcoin-ng: A scalable blockchain protocol. In *13th Symposium on Networked Systems Design and Implementation (16)*, pages 45–59, 2016.

[44] Guy Zyskind, Oz Nathan, et al. Decentralizing privacy: Using blockchain to protect personal data. In *2015 IEEE Security and Privacy Workshops*, pages 180–184. IEEE, 2015.

[45] Ethan Heilman. One weird trick to stop selfish miners: Fresh bitcoins, a solution for the honest miner. In *International Conference on Financial Cryptography and Data Security*, pages 161–162. Springer, 2014.

[46] Chris Sinclair, Lyn Pierce, and Sara Matzner. An application of machine learning to network intrusion detection. In *Proceedings 15th Annual Computer Security Applications Conference (ACSAC'99)*, pages 371–377. IEEE, 1999.

[47] Nikolaos Alexopoulos, Emmanouil Vasilomanolakis, Natalia Reka Ivanko, and Max Muhlhauser. Towards blockchain-based collaborative intrusion detection systems. In *International Conference on Critical Information Infrastructures Security*, pages 107–118. Springer, 2017.

[48] Xueping Liang, Sachin Shetty, Deepak Tosh, Charles Kamhoua, Kevin Kwiat, and Laurent Njilla. Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability. In *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, CCGrid '17, pages 468–477, Piscataway, NJ, USA, 2017. IEEE Press.

[49] Jiaxing Li, Jigang Wu, and Long Chen. Block-secure: Blockchain based scheme for secure p2p cloud storage. *Information Sciences*, 465:219–231, 2018.

[50] Manasa Sreekesh et al. A two-tier network based intrusion detection system architecture using machine learning approach. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pages 42–47. IEEE, 2016.

[51] Majjed Al-Qatf, Yu Lasheng, Mohammed Al-Habib, and Kamal Al-Sabahi. Deep learning approach combining sparse autoencoder with svm for network intrusion detection. *IEEE Access*, 6:52843–52856, 2018.

[52] Kaichen Yang, Jianqing Liu, Chi Zhang, and Yuguang Fang. Adversarial examples against the deep learning based network intrusion detection systems. In *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*, pages 559–564. IEEE, 2018.

[53] Md Zahangir Alom and Tarek M Taha. Network intrusion detection for cyber security using unsupervised deep learning approaches. In *2017 IEEE National Aerospace and Electronics Conference (NAECON)*, pages 63–69. IEEE, 2017.

[54] Robert Mitchell and Ray Chen. A survey of intrusion detection in wireless network applications. *Computer Communications*, 42:1–23, 2014.

[55] Md Zahangir Alom, VenkataRamesh Bontupalli, and Tarek M Taha. Intrusion detection using deep belief networks. In *2015 National Aerospace and Electronics Conference (NAECON)*, pages 339–344. IEEE, 2015.

[56] Khaled Alrawashdeh and Carla Purdy. Toward an online anomaly intrusion detection system based on deep learning. In *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 195–200. IEEE, 2016.

[57] Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A Ghorbani. A detailed analysis of the kdd cup 99 data set. In *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pages 1–6. IEEE, 2009.

[58]P Gifty Jeya, M Ravichandran, and CS Ravichandran. Efficient classifier for r2l and u2r attacks. *International Journal of Computer Applications*, 45(21):28–32, 2012.

[59]S Revathi and A Malathi. A detailed analysis on nsl-kdd dataset using various machine learning techniques for intrusion detection. *International Journal of Engineering Research & Technology (IJERT)*, 2(12):1848–1853, 2013.

[60]L Dhanabal and SP Shantharajah. A study on nsl-kdd dataset for intrusion detection system based on classification algorithms. *International Journal of Advanced Research in Computer and Communication Engineering*, 4(6):446–452, 2015.

[61] Zaib Hassan M. Nsl-kdd. https://kaggle.com/hassan06/nslkdd. accessed 5 nov. 2019. In *Network Security, Information Security, Cyber Security*, 2019.

[62] Shadi Aljawarneh, Monther Aldwairi, and Muneer Bani Yassein. Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. *Journal of Computational Science*, 25:152–160, 2018.

[63] Mrutyunjaya Panda, Ajith Abraham, and Manas Ranjan Patra. A hybrid intelligent approach for network intrusion detection. *Procedia Engineering*, 30:1–9, 2012.

[64] Saurabh Mukherjee and Neelam Sharma. Intrusion detection using naive bayes classifier with feature reduction. *Procedia Technology*, 4:119–128, 2012.

[65] Mrutyunjaya Panda, Ajith Abraham, and Manas Ranjan Patra. Discriminative multinomial naive bayes for network intrusion detection. In *2010 Sixth International Conference on Information Assurance and Security*, pages 5–10. IEEE, 2010.

[66] Nabila Farnaaz and MA Jabbar. Random forest modeling for network intrusion detection system. *Procedia Computer Science*, 89:213–217, 2016.

[67] Akhilesh Kumar Shrivas and Amit Kumar Dewangan. An ensemble model for classification of attacks with feature selection based on kdd99 and nsl-kdd data set. *International Journal of Computer Applications*, 99(15):8–13, 2014.

[68] Abebe Tesfahun and D Lalitha Bhaskari. Intrusion detection using random forests classifier with smote and feature reduction. In *2013 International Conference on Cloud & Ubiquitous Computing & Emerging Technologies*, pages 127–132. IEEE, 2013.

[69] Preeti Aggarwal and Sudhir Kumar Sharma. Analysis of kdd dataset attributes-class wise for intrusion detection. *Procedia Computer Science*, 57:842–851, 2015.

[70] Hee-su Chae, Byung-oh Jo, Sang-Hyun Choi, and Twae-kyung Park. Feature selection for intrusion detection using nsl-kdd. *Recent advances in computer science*, pages 184–187, 2013.

[71] Mohammadreza Ektefa, Sara Memar, Fatimah Sidi, and Lilly Suriani Affendey. Intrusion detection using data mining techniques. In *2010 International Conference on Information Retrieval & Knowledge Management (CAMP)*, pages 200–203. IEEE, 2010.

[72] Dikshant Gupta, Suhani Singhal, Shamita Malik, and Archana Singh. Network intrusion detection system using various data mining techniques. In *2016 International Conference on Research Advances in Integrated Navigation Systems (RAINS)*, pages 1–6. IEEE, 2016.

[73] Himadri Chauhan, Vipin Kumar, Sumit Pundir, and Emmanuel S Pilli. A comparative study of classification techniques for intrusion detection. In *2013 International Symposium on Computational and Business Intelligence*, pages 40–43. IEEE, 2013.

[74] Kedar Potdar, Taher S Pardawala, and Chinmay D Pai. A comparative study of categorical variable encoding techniques for neural network classifiers. *International Journal of Computer Applications*, 175(4):7–9, 2017.

[75] Tingkai Sun and Songcan Chen. Class label versus sample label-based cca. *Applied Mathematics and computation*, 185(1):272–283, 2007.

[76]R Najafi and Mohsen Afsharchi. Network intrusion detection using tree augmented naive-bayes. In *The Third International Conference on Contemporary Issues in Computer and Information Sciences (CICIS'12)*, 2012.

[77] Muhammad Shakil Pervez and Dewan Md Farid. Feature selection and intrusion classification in nsl-kdd cup 99 dataset employing svms. In *The 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014)*, pages 1–6. IEEE, 2014.

[78] Yuanfeng Cai and Dan Zhu. Fraud detections for online businesses: a perspective from blockchain technology. *Financial Innovation*, 2(1):20, 2016.

[79] Jennifer J Xu. Are blockchains immune to all malicious attacks? *Financial Innovation*, 2(1):25, 2016.

[80] Phillip A Porras and Peter G Neumann. Emerald: Event monitoring enabling response to anomalous live disturbances. In *Proceedings of the 20th national information systems security conference*, pages 353–365, 1997.

[81] Pradip Kumar Sharma, Seo Yeon Moon, and Jong Hyuk Park. Block-vn: A distributed blockchain based vehicular network architecture in smart city. *JIPS*, 13(1):184–195, 2017.

[82] Robin Sommer and Vern Paxson. Outside the closed world: On using machine learning for network intrusion detection. In *2010 IEEE symposium on security and privacy*, pages 305–316. IEEE, 2010.

[83] Nutan Farah Haq, Abdur Rahman Onik, Md Avishek Khan Hridoy, Musharrat Rafni, Faisal Muhammad Shah, and Dewan Md Farid. Application of machine learning approaches in intrusion detection system: a survey. *IJARAI-International Journal of Advanced Research in Artificial Intelligence*, 4(3):9–18, 2015.

[84] Donghwoon Kwon, Hyunjoo Kim, Jinoh Kim, Sang C Suh, Ikkyun Kim, and Kuinam J Kim. A survey of deep learning-based network anomaly detection. *Cluster Computing*, pages 1–13, 2017.

[85] Ahmad Javaid, Quamar Niyaz, Weiqing Sun, and Mansoor Alam. A deep learning approach for network intrusion detection system. In *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*, pages 21–26. ICST (Institute for Computer Sciences, Social-Informatics and â¦, 2016.

[86] Hatim Mohamad Tahir, Wail Hasan, Abas Md Said, Nur Haryani Zakaria, Norliza Katuk, Nur Farzana Kabir, Mohd Hasbullah Omar, Osman Ghazali, and

Noor Izzah Yahya. Hybrid machine learning technique for intrusion detection system. 2015.

[87] Hamed Haddad Pajouh, GholamHossein Dastghaibyfard, and Sattar Hashemi. Two-tier network anomaly detection model: a machine learning approach. *Journal of Intelligent Information Systems*, 48(1):61–74, 2017.

[88] Mostafa A Salama, Heba F Eid, Rabie A Ramadan, Ashraf Darwish, and Aboul Ella Hassanien. Hybrid intelligent intrusion detection scheme. In *Soft computing in industrial applications*, pages 293–303. Springer, 2011.