TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

THAPATHALI CAMPUS

A Major Project Progress Report

On

Intrusion Detection System for IOT Networks using Machine Learning

Submitted by:

Krishna Rauniyar (073/BEX/318)

Nabin Pakka (073/BEX/320)

Rupan Chaulagain (073/BEX/337)

Sagar Dangal (073/BEX/338)

Submitted to:

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

THAPATHALI CAMPUS

KATHMANDU, NEPAL

February, 2020

ACKNOWLEDGMENT

It gives us immense pleasure to express our deepest sense of gratitude and sincere thanks to the Department of Electronics and Computer Engineering, Thapathali Campus and our highly respected and esteemed teachers, for their valuable guidance and encouragement. Their useful suggestions and co-operative behavior are sincerely acknowledged.

We would also like to express our sincere thanks to our HOD Er. Kiran Chandra Dahal, for giving us this opportunity to undertake this project and also for his whole-hearted support.

We would also like to express our special thanks to Mr. Sandip Pandey, CTO of Sireto technology for providing us with this golden opportunity to do this project.

We would also like to express our special thanks to Mr. Dipendra Rauniyar as our supervisor for all his encouragement, help and timely support as a teacher.

We also express our gratefulness to our friends, who got directly or indirectly engaged in this project.

Krishna Rauniyar (073/BEX/318)
Nabin Pakka (073/BEX/320)
Rupan Chaulagain (073/BEX/337)
Sagar Dangal (073/BEX/338)

ABSTRACT

The task of developing Intrusion Detection System (IDS) crucially depends on the preprocessing along with selecting important data features of it. Another crucial factor is the design of effective learning algorithm that classify normal and anomalous patterns. The objectives of this research work is to propose a new and better version of the Naive Bayes classifiers that improves the accuracy of the intrusion detection in IDS. The proposed classifier is also supposed to take less time as compared to the existing classifier. To gain better accuracy and fast processing of network traffic, this study applied some standard methods of feature selection. This study tested the performance of the new proposed classifier algorithm with existing classifier, namely Naive bayes, J48 and REPTree thereby measuring different performance parameters using 10-fold cross validation. This study evaluates the performance of the new proposed classifier algorithm by using CICIDS-2019 Dataset. All these algorithms have been implemented in WEKA machine learning tool to evaluate performance. Empirical results of our study show that the proposed updated version of the Naive Bayes classifiers gives better result in terms of intrusion detection and false alarm rate. Furthermore, a network of IOT devices is formed. A raspberry pi is used as main server for communication between the devices. The intrusion detection application is used in raspberry pi. Attacks such as port scan, DDoS etc. are performed on the network before and after implementing the intrusion detection system application. The results are graphed and analyzed as final output.

Keywords: Denial-of-service attack (DoS), Multi-Layer Perceptron (MLP), Network Intrusion Detection system (NIDS), User to Root (U2R), Remote to Local (R2L), Open web application security project (OWASP), Core rule set (CRS), Cross site scripting (XSS), Web application firewall (WAF)

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ABC	Association Based Classification
CICIDS	Canadian Institute of Cybersecurity IDS
CFS	Correlation feature selection
CRS	Core Rule Set
DoS	Denial of Service
DDoS	Distributed Denial of Service
HIDS	Host intrusion detection system
HTTP	Hyper Text Transfer Protocol
IDS	Intrusion Detection System
IoT	Internet of Things
MLP	Multi-Layer Perceptron
NIDS	Network Intrusion Detection system
OWASP	Open Web Application Security Project
R2L	Remote to Local
SSH	Secure Shell
SQL	Structured Query Language
UIDs	Unique Identifiers
WEKA	Waikato Environment for Knowledge Analysis
XSS	Cross Site Scripting

1 INTRODUCTION

1.1 Background

The most common risk to a network's security is an intrusion such as brute force, denial of service or even an infiltration from within a network. With the changing patterns in network behavior, it is necessary to switch to a dynamic approach to detect and prevent such intrusions. A lot of research has been devoted to this field, and there is universal acceptance that *static datasets do not capture traffic compositions and interventions*. This is where an intrusion detection system using machine learning techniques comes into play. An intrusion detection system (IDS) is a system that monitors network traffic for suspicious activity and issues alerts if such activities are discovered. Although, its primary function is to detect anomalies and report them, some of these systems are also capable of taking actions on these malicious activities and blocking traffic sent from suspicious IP addresses. Intrusion detection systems can use a different kind of methods to detect suspicious activities. It can be broadly divided into **Signature based intrusion detection** and **Anomaly based intrusion detection**.

The systems using signature-based intrusion detection compares the incoming traffic with a pre-existing database of known attack patterns known as **signatures**. Detecting new attacks is difficult. The vendors supplying the systems actively release new names i.e. similar to anti-virus software. The systems using anomaly-based intrusion detection uses statistics to form a baseline usage of the networks at different time intervals. They were introduced to detect unknown attacks. This system uses machine learning to create a model simulating regular activity and then compares new behavior with the existing model. IDS detect intrusions in different places. Based on where they discover, they can be classified as **Network intrusion detection (NIDS)** and **Host intrusion detection (HIDS)**. NIDS is a strategically placed system to monitor all the network traffic. HIDS runs on all devices in the network which is connected to the internet/ intranet of the organization. They can detect malicious traffic which originates from within. IDS can be classified as **Active** and **Passive** based on their action. Active IDS is also known as an intrusion detection and prevention system. It generates alerts and logs entries along with commands to change the configuration to protect the network.

Passive IDS just detect malicious activity and generates an alert or logs, but it doesn't take any action.

The Internet of Things (IOT) is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. The definition of the internet of things has evolved due to the convergence of multiple technologies, real-time analytics, machine learning, commodity sensors, and embedded systems. Traditional fields of embedded systems, wireless sensor networks, control systems, automations (including home and building automation), and others all contribute to enabling the internet of things.

Feature selection is the process of selecting a subset of relevant features for use in model construction. Feature selection techniques are used for several reasons:

- Simplification of models to make them easier to interpret by researchers
- Shorter training times
- To avoid the curse of dimensionality
- Enhanced generalization by reducing overfitting

The central premise when using a feature selection technique is that the data contains some features that are either redundant or irrelevant, and can thus be removed without incurring much loss of information.

The Correlation feature selection (CFS) measure evaluates subsets of features on the basis of the following hypothesis: "Good feature subsets contain features highly correlated with the classification, yet uncorrelated to each other'. [1]

A decision tree is a simple structure where non-terminal nodes represent tests on one or more attributes and terminal nodes reflect decision outcomes. The information gain measure is used to select the test attribute at each node of the decision tree. The information gain measure prefers to select attributes having a larger number of values.

In machine learning, naive bayes classifiers are a family of simple "probabilistic classifiers" based on applying bayes' theorem with strong independence assumptions

between the features. They are among the simplest bayesian network models. Naive Bayes has been studied extensively since the 1960s. It was introduced into the text retrieval community in the early 1960s, and remains a popular method for text categorization, the problem of judging documents as belonging to one category or the other (such as spam or legitimate, sports or politics, etc.) with word frequencies as the features.

OneR, short for “One Rule”, is a simple, yet accurate, classification algorithm that generates one rule for each predictor in the data, then selects the rule with the smallest total error as its “one rule”. To create a rule for a predictor, we construct a frequency table for each predictor against the target. It has been shown that OneR produces rules only slightly less accurate than state-of-the-art classification algorithms while producing rules that are simple for humans to interpret. [2]

1.2 Motivation

There are many limitations to the traditional intrusion detection system. Several real attacks are far less than the number of false alarms raised. This causes real threats to go often unnoticed. Constant software updates are required for signature-based IDS to keep up with the new threats. Therefore, we need the modifiable, reproducible and extensible dataset to learn and tackle sophisticated attackers who can easily bypass basic intrusion detection systems. This is where machine learning comes into play. The use of machine learning on network security systems can be employed to build robust IDS.

1.3 Problem Definition

Since many years ago, lights are controlled via the manual switching of the switches and it cannot be remotely controlled. Also, when people forget to turn off the switches while going out which leads to constant energy drain. So, in this modern world, to overcome these problems, we moved forward in doing this project.

1.4 Objectives

- To generate a model from dataset to tackle sophisticated attackers.
- To build a robust IDS using machine learning techniques for IOT network.

- To notify concern party if system is attacked.
- To take measures to prevent breach.

1.5 Project Applications

- Anomaly detection
- Industrial applications
- Prevention of attacks on IOT devices with alert system

1.6 Scope of Project

- Generating an intrusion detection model from real world dataset using machine learning techniques.
- Building a simple IOT network
- Building a robust intrusion detection application using the generated dataset.

1.7 Report Organization

The material presented at the report is organized into ten chapters. Chapter 1 is an introduction section which mainly describes the background, motivation to choose this project, problem definition of project, objective of the project, scope and application of the project. Chapter 2 presents the brief summaries of the works that have already been carried out in the past related to this project. Chapter 3 is related to the hardware and software requirement analysis which briefly describes why and where the abovementioned hardware/ software requirements are used. Chapter 4 explains how a particular sequence in which the work has been carried out along with detail procedures, 4 block diagram of data flow diagram which describes the explanation of how the hardware and software are used to accomplish this project. Chapter 5 contains the details of the implementation of the things that have been explained in the methodology. It describes how the system methodology is implemented. Chapter 6 contains the result of our project. Chapter 7 gives the information about the future enhancements that can be implemented in our project. Chapter 8 contains the additional information's such as project, budget, project timeline, circuit diagrams, PCB designs and module's specifications. Chapter 9 contains the references from which we were able to complete the project.

2 LITERATURE REVIEW

2.1 Anomaly Detection Systems

Bayesian Network is a probabilistic graphical model that represents a set of variables and their probabilistic independencies. Bayesian networks, literally, are directed acyclic graphs whose nodes represent variables, and whose edges encode conditional dependencies between the variables [3]. They have been applied in anomaly detection in different ways; for example, Valdes et al. developed an anomaly detection system that employed Naive Bayes, which is a two-layer Bayesian network that assumes complete independency between the nodes. [4]

2.2 CICIDS2017 dataset

The CICIDS2017 dataset spanned over eight different files containing five days normal and attacks traffic data of Canadian Institute of Cybersecurity. This state of art dataset not only contains UpToDate network attacks but also fulfils all the criteria of real-world attacks. [5]

2.3 Network Intrusion Detection

Deep packet inspection is a major component in Network Intrusion Detection system where incoming data streams packets need to be compared with patterns in an attack database, byte-by-byte, using string matching or regular expression matching. Regular expression matching, despite flexibility/efficiency in attack identification, has high computation and storage complexities for NIDS, making line-rate packet processing challenging [6]. Stride Finite Automata (StriFA), a new finite automata family, to accelerate string matching and regular expression matching was presented by Wang, et al., (2013). Axellson proposes implication and base-rate fallacy for intrusion detection system that works on the principle of Bayesian rule of conditional probability [7].

2.4 Feature Selection for IDS

Labeled datasets have a big role in validating and evaluating machine learning techniques in IDS. To obtain evaluation accuracy very large datasets should be considered. Intrusion traffic and normal traffic are dependent on many network characteristics called features. You Chen et al. explore existing feature selection

algorithms in intrusion detection systems group and compare different algorithms in three broad categories: filter, wrapper, and hybrid [8]. An approach which analyzed intrusion datasets, evaluates features for its relevance to a specific attack, determines a feature's contribution level and eliminates it from a dataset automatically was suggested by Suthaharan, & Panchagnula, (2012).

2.5 Extrusion Detection Systems (EDS)

Extrusion Detection Systems (EDS) deal with huge data amounts with irrelevant and/or redundant features. These result in a slow training and testing process, heavy computational resources and low detection accuracy. Feature selection was an important EDS issue. A new and simple method Enhanced Support Vector Decision Function (ESVDF) for features selection was proposed by Zaman, & Karray, (2009). The novel method chose features based on 2 factors: feature's rank (weight), calculated using Support Vector Decision Function 41 (SVDF), and correlation between features, determined by Forward Selection Ranking (FSR) or Backward Elimination Ranking (BER) algorithm.

2.6 Hybrid Optimization based IDS

Network intrusions which cannot be analyzed, detected and prevented may paralyze the entire system while abnormal detection could prevent it by detecting data's known/unknown character. A mixed fuzzy clustering algorithm using Quantum-behaved Particle Swarm Optimization (QPSO) algorithm combined with Fuzzy CMeans (FCM) was adopted and used in abnormal detection by Hao Wang, et al., (2010). The iteration algorithm was replaced by new hybrid algorithm based on FCM gradient descent making the algorithm a strong global searching entity and avoiding FCM's local minimum problems. Number of hybrid techniques has been used in machine learning field to overcome the problem of feature selection in intrusion detection. Hybrid approach based upon neural fuzzy or fuzzy genetic combine classification and clustering to enhance the performance of IDS [9].

2.7 IDS in IoT

By 2009, Cho, et al. [10] present a centralized IDS for IoT where packets that pass through the border router, between the physical and the network domain, are analyzed aiming to detect botnet attacks. They propose a detection scheme based on anomaly-based method and assume that botnets cause unexpected changes in the traffic of 6LoWAPN sensors. The proposed solution computes the average for three metrics to compose the normal behavior profile. When metrics from any node violate the computed averages, the system raises an alert.

In their 2011 work, Le et al. [11] followed the approach of organizing the network in regions. With this approach, they use a hybrid placement strategy to build a backbone of monitor nodes, one per region. The function of monitor nodes is to sniff the communication from its neighbors and define whether a node is compromised. One of the advantages of this solution is that there is no communication overhead.

Also in 2017, Shreenivas et al. [12] propose a solution on IDS for IoT. Their work is an extension of SVELTE, the work presented by Raza et al. With the objective of improving the security within 6LoWPAN networks, the authors extend SVELTE with an intrusion detection module that uses the ETX (Expected Transmissions) metric. In RPL, ETX is a link reliability metric and monitoring the ETX value can prevent an intruder from actively engaging 6LoWPAN nodes in malicious activities. They also propose geographic hints to identify malicious nodes that conduct attacks against ETX-based networks. Their experimental results show that compared with rank-only mechanisms the overall true positive rate increases when they combine the EXT and rank based detection mechanisms.

3 REQUIREMENT ANALYSIS

3.1 Software Requirements

- WEKA for feature selection and classifier analysis

  Waikato Environment for Knowledge Analysis (WEKA) is a free software widely used for data analysis and predictive modeling. It was developed at the University of Waikato, New Zealand. Feature selection and comparison of algorithms are performed using this tool. The datasets can be visualized, decision tree can be created by submitting from the visualized graph.

- Intellij and Pycharm for server setup

  Ideal platform for java development, Intellij is an IDE of JetBrains. While for python, JetBrains have another IDE: Pycharm. Both IDE provides easy and suitable access to many features which makes development easy and reliable.

- Metasploit

  The Metasploit Framework is a Ruby-based, modular penetration testing platform that enables you to write, test, and execute exploit code. The Metasploit Framework contains a suite of tools that you can use to test security vulnerabilities, enumerate networks, execute attacks, and evade detection. At its core, the Metasploit Framework is a collection of commonly used tools that provide a complete environment for penetration testing and exploit development.

- NGINX

  Nginx is a linux-based webserver. It can be a reverse proxy server, a load balancer, a mail proxy and HTTP cache. It is an open-source web server used by majority of the 100,000 busiest websites such as Redit, Wikipedia. The nginx will serve as IoT server. All rule set and IDS will be set in nginx environment.

- Mod Security

  Mod Security is an open-source web application firewall (WAF). It provides an array of Hypertext Transfer Protocol request and response filtering capabilities along with other security features. Mod security is used as firewall for nginx. Based on OSWAP rule sets, nginx rules are set. It serve as filter in between IDS and internet.

- Arduino IDE

  It is an open-source application especially designed to program microcontrollers. Nodemcu is programmed using this IDE. It accepts c as its main programming language.

3.2 Hardware Requirements

- Dc fan

  It is a fan that runs on dc power. It is controlled according to the temperature reading of the temperature sensor.

- DHT 11

  It is a temperature and humidity sensor. It operates in 3.5-5.5 v dc supply with operating current 0.3mA. Temperature range of 0-50 C can be measured through this sensor. 16 bits are used to represent the data. Temperature data provided by DHT 11 is analysed and used to control speed of dc fan.

- Node MCU

  NodeMCU is an open-source IoT platform. It is a Single-board microcontroller having XTOS operating system. It can store upto 4MB of data and memory is 128Kb for processing. USB is used to power the device. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module. It acts as mediator between IoT devices and server.

4 SYSTEM ARCHITECTURE AND METHODOLOGY

4.1 System Architecture

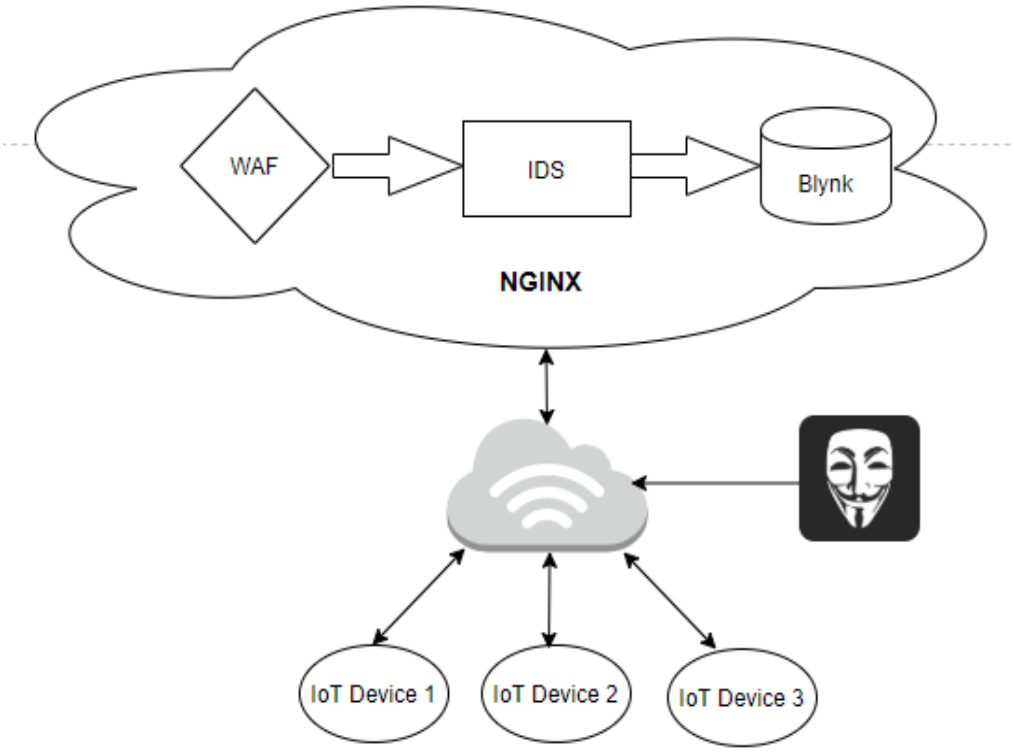

Figure 4.1.1: System Block Diagram

- IoT Devices

  They are interconnected devices such as digital machines, objects, people, animal, mechanical machines, computing devices having ability to transfer data over a network automatically. Each device have their unique identifier. Dc fan, DHT 11 are IoT devices of this project.

- Attacker

  An individual, group of people or computing devices that fling to gain unauthorized access are known to be attackers. There are many attacks such as SQL injection, port scan, DDOS, DOS etc.

- Nginx

  Nginx is a server capable of load balancing and reverse proxy. It is server for IoT.

- WAF

  Web application firewall contains set of OWASP rules to filter known and most obvious attacks.

- IDS

  The intrusion detection system is heart of the block diagram. Attacks that cannot be filtered by firewall are detected here. Packets passed from firewall is subjected to a machine learning model. Result provided by model decides whether given request is an attack or not.

- Blynk

  Blynk is a open source server for IoT devices. This server is main data controller of IoT devices. Blynk is protected by double layer- WAF and IDS.

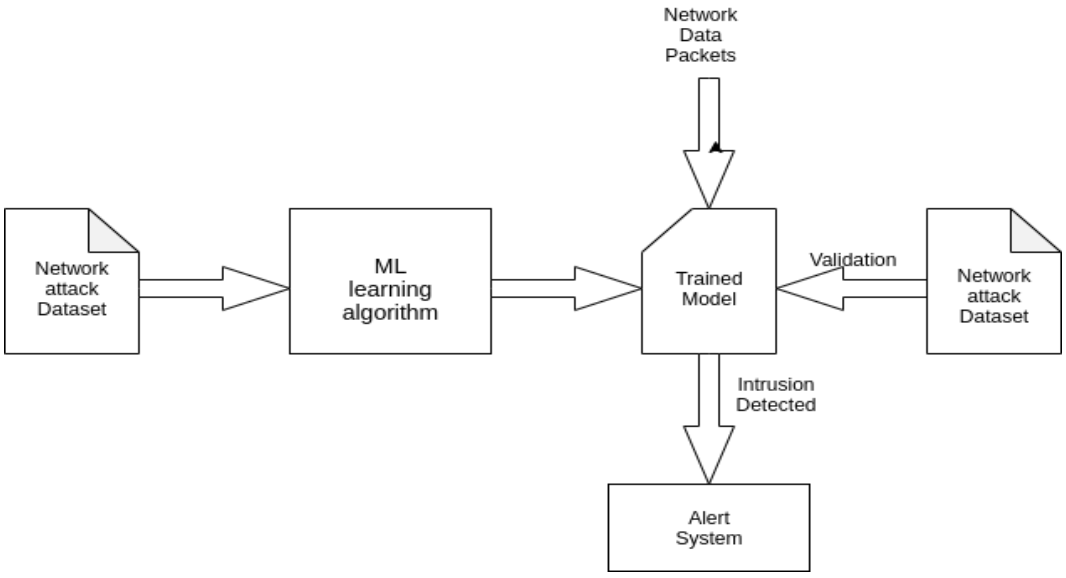

Figure 4.1.2: IDS Block Diagram

4.2 Methodology

CICIDS-2019 dataset is considered to train the supervised model. The collected data is cleaned by reducing its dimensions for fast and accurate detection. Machine learning algorithms are used to formulate more accurate model for detection of intrusion from real world attack. Result of algorithms are compared using confusion matrix. Validation of the model is checked by using testset as input. The validity check is graphed to obtain the accuracy of the model. The model is reanalysed to obtain accurate model. Final

model is subjected to actual network attack. The network packets are analysed in application layer rather than network layer. Network packets are subjected to test using model obtained. Packets crossing threshold value of similarity to attack as per set by the model is termed as attack. The alert system informs admin or security engineers about the attempt of breach.

A network of IOT devices is formed. A raspberry pi is used as main server for communication between the devices. The intrusion detection application is used in raspberry pi. Attacks such as port scan, DDoS etc are performed on the network before and after implementing the intrusion detection system application. The results are graphed and analysed as final output.

4.2.1 Features selection Algorithms

- Correlation-based feature selection

  Feature selection is a process of choosing a subset of the relevant attribute selected in a large number of basic attributes of a particular dataset by applying unique assessment standards to enhance the pleasant of classifier, while the dimension of the data reduces. It is used to evaluate the subset of features based on the well-suited subsets, which have highly correlated facilities with classification, are still unrelated to each other.

$$\frac{R = \sum_i^n (a_i - \mathrm{A}')(b_i - \mathrm{B}')}{N\sigma_A\sigma_B}$$

- Gain ratio feature evaluator

  Gain ratio is a ratio of information gain to the intrinsic information. It was proposed by Ross Quinlan, to reduce a bias towards multi-valued attributes by taking the number and size of branches into account when choosing an attribute. The information gain ratio is calculated as

  Gain ratio = Gain (A)/ Split info(A)

- Classifier Subset Evaluator
  - It evaluates the specialty subset on training data or a separate hold-out test set.
  - It uses a classification to evaluate the eligibility of a set of features.
  - With whom the classification algorithms perform well, it considers subsets of those tasks.

4.2.2 Machine learning Algorithms

- Gaussian Naïve Bayes (NB)

  Naïve Bayes classifier is a simple probabilistic classifier implementing Bayes theorem with strong independence assumptions between the features. While dealing with continuous data, it is assumed that the continuous values associated with each class are distributed according to a Gaussian distribution. If the training data contains a continuous attribute x, We first segment the data by class, and then compute the mean and variance of x in each class.

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)}$$

- One R

  OneR, short for "One Rule", is a simple, yet accurate, classification algorithm that generates one rule for each predictor in the data, then selects the rule with the smallest total error as its "one rule". To create a rule for a predictor, we construct a frequency table for each predictor against the target. It has been shown that OneR produces rules only slightly less accurate than state-of-the-art classification algorithms while producing rules that are simple for humans to interpret.

- Reduced Error Pruning Tree

  REPTree is considered to be a quick decision-making tree, which uses the benefit of information as the criterion of division to make the decision

regression tree, and prunes it using a low error pruning method. In cases of large volumes of training and test data, the result of reducing the error is a more accurate and simple classification tree.

4.2.3 Dataset

It is hardly possible to amass enough data of network attacks and anomalies in just a year. Also, companies do not disclose such data for security reasons. Hence dataset provided by some universities are being used for this system to train the model. Dataset KDD'99 (University of California, Irvine 1998, 99), CDX (United States Military Academy 2009), CAIDA (Center of Applied Internet Data Analysis – 2002/2016), CICIDS datasets etc. are to be used for training. The KDD'99 is widely used and reliable data set but it is outdated hence results in high false alarms and breaches. Hence CICIDS-2019 data set will be used to train the model. It consists of 58 features. Only 21 features are selected for the training. WEKA tool is used for feature selection.

5 IMPLEMENTATION DETAILS

5.1 Component Function

- DHT 11

  VDD of DHT 11 is connected to 3v of nodemcu, data is received from pin D4, ground to the ground of nodemcu. Temperature data received is used as controlling parameter for the speed of dc fan.

- Dc fan

  It is connected to D2 of nodemcu. Its speed is controlled according to the temperature reading from DHT 11.

- NodeMCU

  NodeMCU is the mediator between IoT devices and server. It also acts as controlling unit for the IoT network. It is connected to the server via internet. An Ip address or domain name is assigned.

- NGINX

  It is Linux-based web server. It can act as load balancer, reverse proxy. A script or nginx.conf file is modified as required to monitor the network traffic. The http requests are redirected to https for secure requests. ModSecurity, a third party WAF is integrated with it.

- Blynk

  The IoT devices are connected to the blynk server through nginx-WAF-IDS. It is a open-source server especially designed for IoT devices. The IoT devices are connected to blynk through nodemcu.

- ModSecurity

  It is an open-source WAF. It is integrated with nginx. Based on OWASP CRS, the firewall is setup. This acts as an extra layer of protection between the internet and IDS. It filter signed attacks. SQL injection, DDos, XSS etc are top attacks according to 2019 OWASP.

- LetsEncrypt

  The https requests are not recognized by pcs or devices. It is made secure by letsencrypt. It is used in nginx. It modifies the nginx.conf file to redirect all the requests to https request.

- Certbot

  It is an open-source application that helps to generate and automatically re-new the certificate for letsencrypt.
- Arduino IDE

  It is an open-source software to program micro-controlling unit. Nodemcu is programmed by writing codes in Arduino IDE. The code is uploaded to nodemcu using a USB cable.

5.2 Interfacing protocols

5.2.1 SSH

Secure Shell is a cryptographic network protocol for operating network services securely over an unsecured network. Typical applications include remote command-line, login, and remote command execution, but any network service can be secured with SSH. The protocol is used to access the remote nginx server in personal computer. All the nginx configuration, modsecurity configuration and other server related configurations are performed through ssh.

5.2.2 HTTPS

Hypertext Transfer Protocol Secure (HTTPS) is an extension of the Hypertext Transfer Protocol (HTTP). It is used for secure communication over a computer network, and is widely used on the Internet. In HTTPS, the communication protocol is encrypted using Transport Layer Security (TLS) or, formerly, its predecessor, Secure Sockets Layer (SSL). IoT devices are connected to blynk server using https protocol.

6 RESULTS AND ANALYSIS

Figure 6.1: Access log of nginx

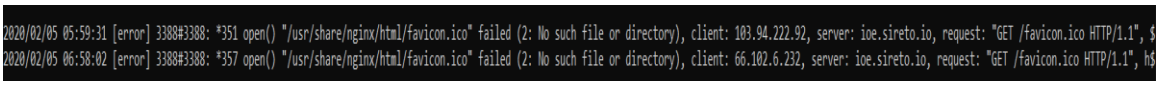

Figure 6.2: Error log of nginx

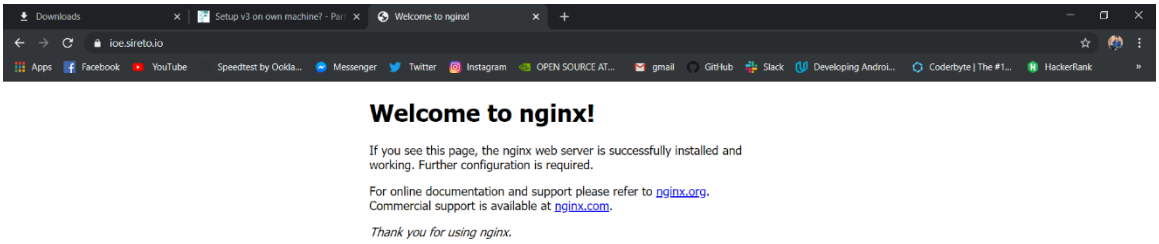

Figure 6.3: Welcome page through LetsEncrypt

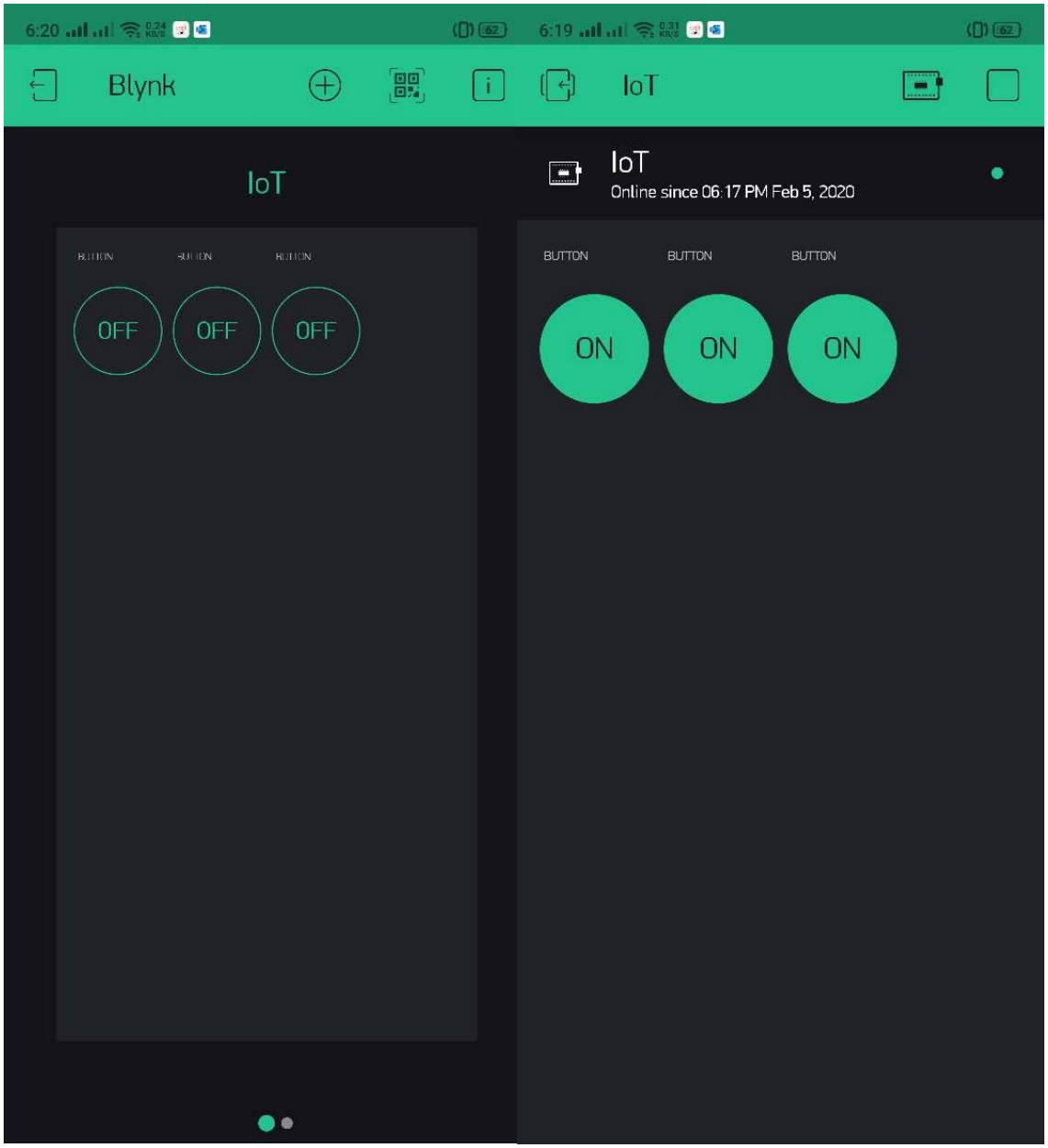

Figure 6.4: APP Illustration controlling IoT devices

Ngnix was setup in Contabo server, a Germany based server, with domain name “ioe.sireto.io”. The network traffic on the domain was stored in access log of nginx in /var/log/nginx/access.log and the error log was stored in /var/log/nginx/error.log. The access log file contains the ip of client, date of request, type of request, response code and browser info. The error log contains date of request, ip of client, error type, cause

of error and request type. There was insecure error while calling the domain ioe.sireto.io until letsencrypt was implemented.

The blynk server was setup in blynk server and accessed through blynk app. The data from DHT 11, led and fan was controlled using the app. A pcb was designed and etched for the circuit.

There were many errors during setup of server due to change in version of the software. Some files were mis-configured which caused 404 error. Also due to not secure environment of https request pc refused connection with the domain ioe.sireto.io. During etching, some connections were lost.

7 REMAINING TASKS

7.1 Completed Tasks

- Nginx setup

  The open-source server, ngnix, is setup with lets encrypt enabled.
- PCB design and etching

  Proteus, a circuit simulating and pcb designing tool, is used to create required circuit. Using $FeCl_3$, the printed circuit was etched.
- Blynk connection with nodemcu

  Blynk server was setup and nodemcu was connected using blynk authentication token
- IoT devices connected to server through nodemcu

  Devices were connected in the pcb. Devices were connected to the server through the setup nodemcu.

7.2 Remaining Tasks

- Training of the model for attack detection
- Setting up modsecurity in nginx
- Setting up the blynk server in custom server
- Attacking the server

8 APPENDICES

8.1 Project Timeline

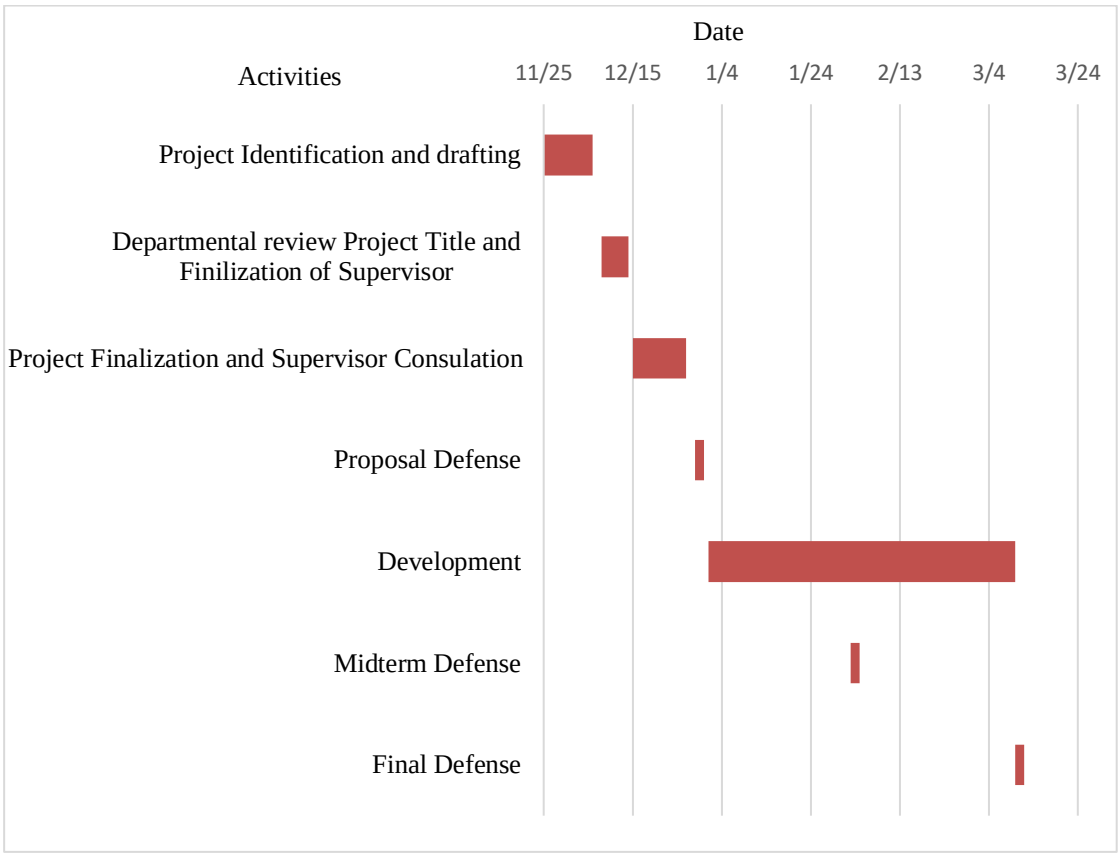

Figure 8.1: Gantt Chart

8.2 Project Budget

Table 8.2: Cost Estimation

Material	**Price (in Rs)**
DHT 11	350
DC fan	300
Node MCU	1000
Transistor	10
Relay	50
Bulb	180
Miscellaneous	5000
Total	**6890**

8.3 Circuit Diagrams

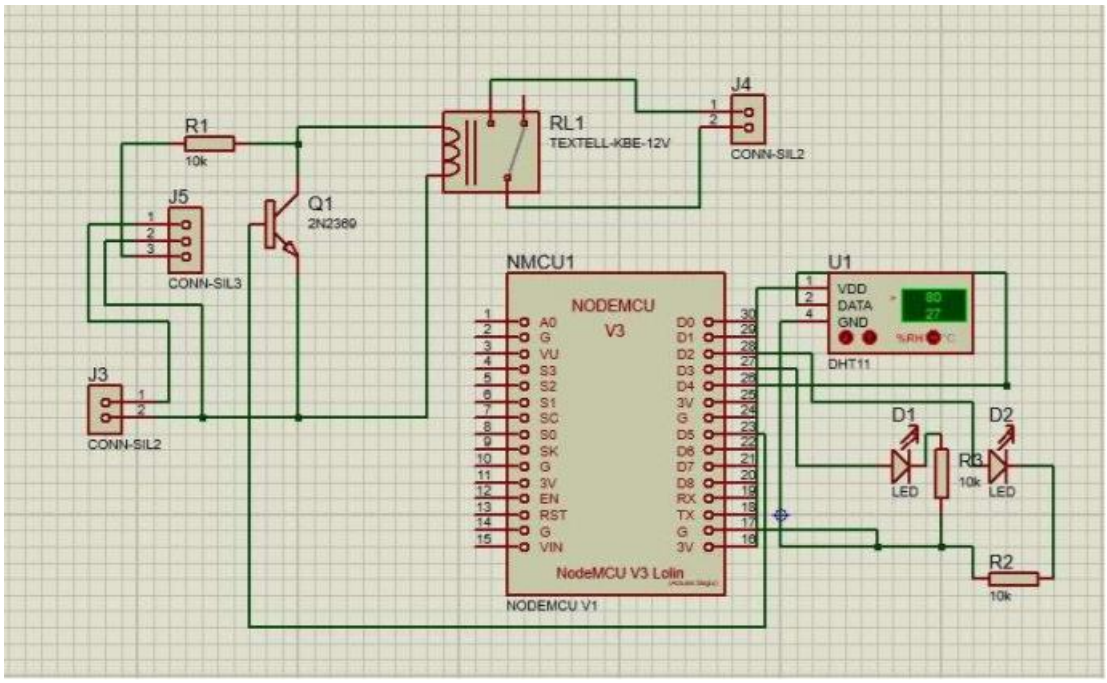

Figure 8.3: Circuit Design

8.4 PCB Designs

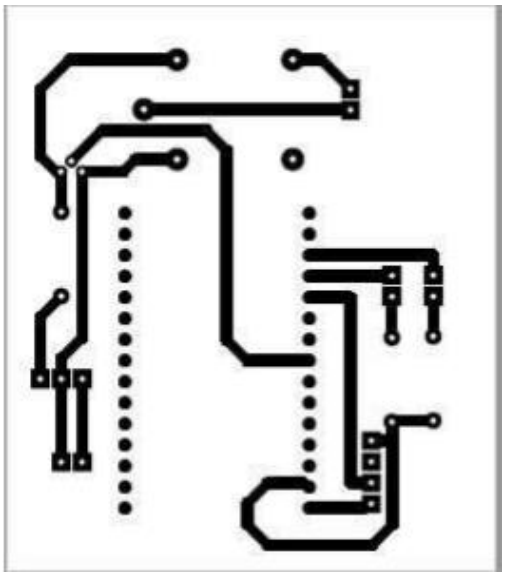

Figure 8.4: PCB design

8.5 Module Specifications

8.5.1 NODEMCU MODULAR SPECIFICATIONS

- Microcontroller: ESP-8266 32-bit
- Clock Speed: 80 MHz
- USB Converter: CP2102
- USB Connector: Micro USB
- Operating Voltage: 3.3V
- Flash Memory: 4MB
- Digital I/O: 11
- Analog Inputs: 1
- Communications: Serial, SPI I2C and 1-Wire via software libraries
- Wi-Fi: Built-in 802.11 b/g/n

8.5.2 DHT-11 MODULAR SPECIFICATION

- Operating Voltage: 3.5V to 5.5V
- Operating current: 0.3mA (measuring) 60uA(standby)
- Output: Serial Data
- Temperature Range: 0-50ºC
- Humidity Range: 20% to 90%
- Resolution: Temperature and Humidity both are 16-bit
- Accuracy: ±1ºC and ±1%

8.6 Relevant Datasets

CICIDS- 2019 dataset is used to generate machine learning model. The dataset consists of more than 50 features. Using feature selection algorithms only 15-16 features will be used for training. Data contains destination port, duration of request, source ip address, destination ip address, flow ip address, total duration, length of header, total length of packet etc. The data are divided into different files according to the attack performed. DDoS, port scan, web attacks etc are performed for dataset.

References

[1] J. S. B. Koushal Kumar, "Network Intusion Detection with Feature Selection Techniues using Machine Learning Algorithms," *Internation Journal of Computer Applications,* vol. 150, no. 12, p. 13, 2016.

[2] Y. R. Shailesh Singh Panwar, "Evaluation of Network Intusion Detection with Feature Selection and Machine Learning algorithms on CICIDS-2017 Dataset," *ICAESMT,* p. 10, 2018.

[3] D. Heckerman, "A tutorial on learning with bayesian networks," 1995.

[4] K. S. A. Valdes, "Adaptive Model-Based Monitoring for Cyber Attack Detection," *Third International Symposium on Recent Advances in Intrusion Detection,* pp. 80-92, 2000.

[5] A. H. L. a. A. A. G.-n. Iman Sharafaldin, "Towards Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," *Fourth Internation Conference on Information System Security and Privacy,* 2018.

[6] K. K. Y. a. J. L. T. S. Chou, "Network Intrusion Detection Design Using Feature Selection of Soft Computing Paradigms," *Internation Journal of Computational Intelligence,* 2008.

[7] S. Axelsson, "The base rate fallacy and its implications for the difficulty of Intrusion detection," in *ACM Conference on Computer and Communication Security,* 1999.

[8] Y. L. X.-Q. C. a. L. G. You Chen, "Survey and Taxonomy of Feature Selection Algorithms in Intrusion Detection System," Springer, Berlin, 2006.

[9] A. A. M. R. P. Mrutyunjaya Panta, "A Hybrid Intelligent Approach for Network Intrusion Detection," in *International Conference on Communication Technology and System Design*, 2011.

[10] E. Cho, J. Kim, and C. Hong, "Attack model and detection scheme for botnet on 6LoWPAN," In Management Enabling the Future Internet for Changing Business and New Computing Services, Lecture Notes in Computer Science 5787. Springer, Berlin, Heidelberg, 515-518, 2009.

[11] A. Le, J. Loo, Y. Luo, and A. Lasebae, "Specification-based IDS for securing RPL from topology attacks," In: Wireless Days (WD), 2011 IFIP, pp. 1-3, 2011.